

# Proyecto Final

## 2020

*Universidad Tecnológica Nacional - Facultad Regional Mendoza*  
*Ingeniería en Sistemas de Información*



**Omics**

**Sistema Líder de Gestión Inteligente Digital de Cómicos**

**Integrantes del Grupo N°8:**

- ❖ Daniela Amin 37256
- ❖ Alan Boglioli 38507
- ❖ Mariel Haarth 23797
- ❖ Gustavo Lemos 32069
- ❖ Julián Muñoz Velázquez 33145

**Plantel Docente:**

- ❖ Alejandro Vazquez
- ❖ Raúl Moralejo
- ❖ Gustavo Manino
- ❖ Marina Carrizo

## Resumen

### 1. Motivación

Existen diversos servicios y aplicaciones enfocados en la distribución de cómics y la lectura (visualización) de los mismos. Sin embargo, estos servicios tienen falencias en alguna de estas dos funcionalidades.

Existen grandes editoriales que distribuyen obras de este estilo, aunque para nuevos autores es muy difícil entrar en el campo de los cómics y hacer conocer sus obras. Además, no existen estrategias claras para estos autores a la hora de obtener ganancias con sus obras.

A su vez, muchas de estas distribuidoras carecen de lectores digitales adecuados (cómodos y sencillos) para el usuario final, en este caso el lector.

También podemos encontrar lectores digitales avanzados, pero aparece la falta de la disponibilidad de un catálogo listo para consumirse en la misma plataforma de lectura.

### 2. Objetivos

El objetivo de *Omic*s es acercar a los usuarios finales una plataforma con un amplio catálogo de obras (cómics) de diversa índole, donde puedan acceder directamente al contenido. También servir como plataforma de publicación y distribución para nuevos autores donde puedan dar a conocer sus obras y recibir donaciones por las mismas.

Por esto, buscamos la excelencia en tres funcionalidades principales:

- ❖ Catálogo de obras, con posibilidad de búsqueda según preferencias.
- ❖ Lector digital de cómics (visor).
- ❖ Gestor de publicaciones de obras para los autores.

## Palabras Clave

Cómics, Entretenimiento, Historias, Creación, Imaginación.

## ÍNDICE

Resumen	2
1. Motivación	2
2. Objetivos	2
Palabras Clave	2
Marco Teórico	17
1. ¿Qué es un Cómic?	17
2. Historia del Cómic	18
3. Diferencias Claves	21
3.1. Cómic	21
3.2. Historieta	22
3.3. Novela Gráfica	22
3.3. Manga	23
4. Problemática Actual de la Industria	25
Trabajo Práctico N°1:	26
Definición de Requerimientos	27
A. Mangatoon	28
1. Relevamiento General	28
1.1. De la Organización	28
1.2. Funciones Detectadas e Interfaces	29
1.3. Tecnología de Información	31
2. Relevamiento Detallado y Análisis del Sistema	34
2.1. Detalle, explicación y documentación detallada de todas las funciones seleccionadas	34
2.2. Modelo Lógico del Sistema Actual	41
2.3. Problemas y necesidades detectadas en las funciones relevadas en detalle y en su entorno organizacional	43
B. Tapas	45
1. Relevamiento General	45
1.1. De la Organización	45

---

1.2. Funciones Detectadas e Interfaces	46
1.3. Tecnología de Información	48
2. Relevamiento Detallado y Análisis del Sistema	49
2.1. Detalle, explicación y documentación detallada de todas las funciones seleccionadas	49
2.2. Modelo Lógico del Sistema Actual	57
2.3. Problemas y necesidades detectadas en las funciones relevadas en detalle y en su entorno organizacional	59
C. Webtoon	60
1. Relevamiento General	60
1.1. De la Organización	60
1.2. Funciones Detectadas e Interfaces	60
1.3. Tecnología de Información	61
2. Relevamiento Detallado y Análisis del Sistema	62
2.1. Detalle, explicación y documentación detallada de todas las funciones seleccionadas	62
2.2. Modelo Lógico del Sistema Actual	66
2.3. Problemas y necesidades detectadas en las funciones relevadas en detalle y en su entorno organizacional	66
D. Crunchyroll	67
1. Relevamiento General	67
1.1. De la Organización	67
1.2. Funciones Detectadas e Interfaces	68
1.3. Tecnología de Información	70
2. Relevamiento Detallado y Análisis del Sistema	71
2.1. Detalle, explicación y documentación detallada de todas las funciones seleccionadas	71
2.2. Modelo Lógico del Sistema Actual	76
2.3. Problemas y necesidades detectadas en las funciones relevadas en detalle y en su entorno organizacional	78
E. Rootinks	81

1. Relevamiento General	81
1.1. De la Organización	81
1.2. Funciones Detectadas e Interfaces	82
1.3. Tecnología de Información	83
2. Relevamiento Detallado y Análisis del Sistema	84
2.1. Detalle, explicación y documentación detallada de todas las funciones seleccionadas	84
2.2. Modelo Lógico del Sistema Actual	88
2.3. Problemas y necesidades detectadas en las funciones relevadas en detalle y en su entorno organizacional	88
Comparación y Conclusiones:	90
2.4. Nuevo Sistema	92
1. Objetivos y alcances preliminares del nuevo Sistema	92
1.1. Objetivo general	92
1.2. Funcionalidades y Requisitos	92
1.3. Alcance	93
1.4. Módulos	95
Diseño	99
1. Objetivos y alcances definitivos del nuevo sistema	99
1.1. Objetivos	99
1.2. Alcances	99
2. Salidas del Sistema	102
2.1. Módulo de Registro y Actividad de Usuario	103
2.2. Módulo de Catálogo y Búsqueda	106
2.3. Módulo de Visualización y Seguimiento	107
2.4. Módulo de Reseñas	108
2.5. Módulo de Suscripción y Donación	109
2.6. Módulo de Publicación de Obras	111
2.7. Módulo de Notificaciones	113
2.8. Módulo de Contrato	114
2.9. Módulo de Reportes	116

2.10. Módulo de Configuración	118
3. Modelo Funcional	119
3.1. Diagrama de Casos de Uso	119
3.2. Historias de Usuario	120
3.2.1. Módulo de Registro y Actividad de Usuario	121
3.2.2. Módulo de Catálogo y Búsqueda	128
3.2.3. Módulo de Visualización y Seguimiento	131
3.2.4. Módulo de Reseñas	134
3.2.5. Módulo de Suscripción y Donación	135
3.2.6. Módulo de Publicación de Obras	139
3.2.7. Módulo de Notificaciones	144
3.2.8. Módulo de Contrato	145
3.2.9. Módulo de Reportes	151
3.2.10. Módulo de Configuración	154
4. Pantallas	159
4.1 Mockups	159
4.2 Módulos	159
4.2.1. Módulo de Registro y Actividad de Usuario	159
4.2.2. Módulo de Catálogo y Búsqueda	166
4.2.3. Módulo de Visualización y Seguimiento	168
4.2.4. Módulo de Reseñas	169
4.2.5. Módulo de Suscripción y Donación	169
4.2.6. Módulo de Publicación de Obras	171
4.2.7. Módulo de Notificaciones	176
4.2.8. Módulo de Contrato	176
4.2.9. Módulo de Reportes	180
4.2.10. Módulo de Configuración	181
4.3. Correspondencia entre las Salidas del Sistema, las Historias de Usuario, y las Pantallas:	186
5. Modelo de Datos	197

5.1. Diagrama de Clases	197
Propiedades Importantes	200
5.2. Diagrama de Estados	203
Desarrollo e Implementación	205
1. Programación y Documentación	205
1.1. Tecnologías Utilizadas	205
1.2. Plantilla de Documentación del código	209
1.3. Documentación del Código	210
1.4. Arquitectura	261
1.4.1. Domain-Driven Design	261
1.4.2. Clean Architecture	262
1.4.3. Interacción	263
1.4.4. Consistencia Eventual	265
2. Planificación de Capacitación	267
2.1. Introducción	267
2.2. Alcance	267
2.3. Destinatarios	267
2.4. Objetivos de la Planificación de Capacitación	267
2.5. Temas de Capacitación	268
2.6. Método o modalidad de Capacitación	269
2.7. Recursos Necesarios	270
2.8. Tiempos de Capacitación	271
2.9. Evaluación de Capacitación	272
3. Planificación, Ejecución y Documentación de Pruebas	273
3.1. Alcance	273
3.2. Planificación de pruebas	274
3.2.1. Pruebas de Validación de Ingreso de Datos	274
3.2.2. Pruebas de Lógica de los Módulos Principales	275
3.2.3. Pruebas de Integración entre los Módulos del Sistema	275
3.2.4. Pruebas de Seguridad por Niveles de Usuario	276

3.2.5. Pruebas de Carga	277
3.2.6. Estimación de tiempos	278
3.3. Casos de prueba	278
3.3.1. Pruebas de Validación de Ingreso de Datos	282
3.3.2. Pruebas de Lógica de los Módulos Principales	289
3.3.3. Pruebas de Integración entre los Módulos del Sistema	296
3.3.4. Pruebas de Seguridad por Niveles de Usuario	306
3.3.5. Pruebas de Carga	313
4. Manual de Usuario del Sistema Completo	319
5. Planificación de Implementación del Sistema	320
5.1. Objetivos de la Implementación de la Plataforma	320
5.2. Recursos Involucrados	321
5.3. Actividades	322
5.3.1. Especificaciones de Servidores	322
5.3.2. Equipos de Conectividad	322
5.3.3. Comunicaciones	323
5.3.4. Puestos de Trabajo	323
5.3.5. Infraestructura	323
5.3.6. Carga Inicial de Datos	324
5.3.7. Migración	325
5.3.8. Métodos de Replicación	325
5.3.9. Backup	327
5.3.9.1. Requisitos	327
5.3.9.2. Carga inicial	327
5.3.9.2.1. Creación de la base de datos	328
5.3.9.2.2. Correr las migraciones	332
5.3.9.2.3. Datos de prueba	334
5.3.9.3. Realización de backup	335
5.3.9.3.1. Manual	335
5.3.9.3.2. Automático	336
5.3.9.3.3. Volcado de SQL	336



5.3.9.3.4. Migraciones	338
5.3.9.3.5. Eventos	339
5.3.9.3.6. Métodos de Replicación	339
5.3.9.4. Recuperación	340
5.3.9.4.1. Las tareas para realizar la Recuperación son las siguientes:	341
5.3.9.5. Backup por Sistema	342
5.3.10. Etapas del Método de Conversión	343
5.3.11. Instalación del Sistema	344
5.3.11.1. Instalación del Back-end	344
5.3.11.2. Instalación del Front end	345
5.4. Estimación de tiempos	346
Trabajo Práctico N°2:	347
CAPÍTULO I: Actividades	348
1. Definición y descripción de actividades	348
2. Diagrama de Tiempos	362
3. Definición de Sprint	371
CAPÍTULO II: Organización para la Ejecución del Proyecto	377
1. Metodología Ágil	377
1.1. Técnicas para aplicar la Metodología Ágil:	378
1.1.1. Tablero Kanban	378
1.1.2. SCRUM	379
1.1.3. Diagrama Burndown	382
2. Equipo de Trabajo (estructura, puestos, perfiles, cantidades)	384
3. Funciones Principales de los Miembros del Equipo de Trabajo	385
4. Métodos de Comunicación Formal, Control de Avances, Retroalimentación y Decisiones	388
5. Gestión de Configuración del Software: Método de gestión del versionado durante todo el proyecto	391
CAPÍTULO III: Factibilidad	393
1. Definición y Descripción de los Recursos para cada una de las Actividades	393
2. Diagrama de Recursos	406
2.1. Gráficos de barras que muestran la carga horaria por mes:	416

3. Análisis de Factibilidad	418
3.1. Factibilidad Técnica	418
3.1.1. Infraestructura general	418
3.1.2. Volumen, tipos de datos y captura de los mismos	420
3.1.3. Frecuencia y volumen de ingreso de datos	421
3.1.4. Tecnologías a utilizar	421
3.1.5. Comunicación entre partes del sistema	422
3.1.6. Nivel de automatización de las funciones	422
3.1.7. Escalabilidad y seguridad	423
3.1.8. Funcionamiento ininterrumpido del sistema	423
3.1.9. Métodos de desarrollo, testing, implementación y mantenimiento	425
3.1.10. Backups, recuperación y migraciones	427
3.1.11. Integración con otros sistemas	428
3.1.12. Crecimiento estimado de la cantidad de usuarios: Comunidad	428
Conclusión:	430
3.2. Factibilidad Legal	431
3.2.1. Privacidad y tratamiento de datos	431
3.2.2. Propiedad intelectual de la obra	431
3.2.3. Duración y terminación de continuidad en la plataforma	431
3.2.4. Distribución del contenido y derecho de uso	432
3.2.5. Obligaciones y responsabilidades del usuario en la plataforma	432
3.2.6. Cookies	432
Conclusión:	433
3.3. Factibilidad Económica	433
3.3.1. Gestión de costos	433
3.3.2. Ingresos	434
3.3.3. Recuperación de la inversión	434
Conclusión:	439
3.4. Factibilidad Operativa	440
3.4.1. Insumos	440

3.4.2. Capacitación del equipo	442
3.4.3. Resolución de conflictos	443
3.4.4. Capacitación usuarios	444
Conclusión:	444
Conclusión general:	444
4. Costos desagregados por recursos (personal, tecnología) con periodicidad mensual	446
5. Análisis de Riesgos	448
5.1. Tabla de Riesgos:	451
5.1.1. Tratamiento de Riesgos:	451
Conclusión:	454
6. Análisis de Impacto Ambiental	455
Conclusión:	457
TRABAJO PRÁCTICO INTEGRADOR N°1:	458
1. Riesgos del proyecto	459
2. Conversión e implementación del sistema	464
3. Funciones del jefe de proyecto por importancia	471
4. Principales funciones del Jefe de Proyecto en fase de ejecución	474
5. Incorporación de dos personas	475
6. Liderazgo	478
7. Resolución de conflictos	479
8. Técnicas de motivación	481
9. Aspectos a tener en cuenta para negociar	486
TRABAJO PRÁCTICO INTEGRADOR N°2:	493
1. Estructuras Organizativas	494
2. Servicios del Área Nuevos Proyectos de T.I.	497
3. Recomendaciones Técnicas y de Seguridad Física	497
4. Retroalimentación a 360° del Área Nuevos Proyectos de T.I.	500
5. Coaching Eficaz en el Área de Nuevos Proyectos de T.I.	502
6. Equipo de Trabajo Efectivo y Equilibrado	503

7. Funciones del Tablero de Comandos para el Área de Nuevos Proyectos de T.I.	509
8. Estrategia de Mejora: Área Nuevos Proyectos de T.I.	514
Anexo N°1	519
Diagrama de Gantt	519
Anexo N°2	521
Diagrama de Recursos	521
1. Diagrama de Recursos de Arquitecto de Software	522
2. Diagrama de Recursos de Dev Back-end	523
3. Diagrama de Recursos de Dev Front-end	524
4. Diagrama de Recursos de Product Manager	525
5. Diagrama de Recursos de Scrum Master	526
6. Diagrama de Recursos de Tester	527
Anexo N°3	528
Diagrama de Clases	528
Anexo N°4	530
Mockups	530
Anexo N°5	532
Manual de Usuarios	532
1. Introducción	533
1.1. ¿Qué es Omics?	533
1.2. Objetivos	533
2. Instalación del Sistema	534
2.1. Instalación del Back-end	534
2.2. Instalación del Front-end	535
3. Funciones del Administrador	536
3.1. Generales	536
3.1.1. Configuraciones iniciales	536
3.2. Administrador General	539
3.2.1. Gestionar las Reglas de Negocio	539
3.2.2. Generar Reportes	541

3.3. Administrador Content Manager	542
3.3.1. Gestión de Categorías de Publicaciones	542
3.3.2. Gestión de Publicaciones	545
3.3.2.1. Aceptar Publicación	546
Rechazar Publicación	547
3.3.3. Gestión de Contratos	548
3.3.3.1. Aceptar Contrato	549
3.3.3.2. Rechazar Contrato	550
4. Mensajes de Error	551
4.1. Errores de Pantalla de Registro	551
4.2. Errores de Pantalla de Login	552
4.3. Errores de Pantalla Recuperar Contraseña	552
4.4. Errores de Pantalla Editar Perfil	553
4.5. Errores de Pantalla Eliminar Cuenta	554
4.6. Errores de Pantalla Introducir Pago	555
4.7. Errores de Pantalla Publicar Obra	557
4.8. Errores de Pantalla Crear Colecciones	558
4.9. Errores de Pantalla Editar Colección	559
4.10. Errores de Gestionar Colecciones	559
4.11. Errores de Gestionar Categorías	559
5. Guía de creadores de contenido	561
6. Políticas de Privacidad	562
A) Información que se recolecta	562
B) Cómo usamos la información que recolectamos.	563
C) Con quién compartimos su información.	564
D) Retención de información personal.	565
E) Sus opciones.	566
F) Sus derechos.	566
G) Cómo protegemos la información personal.	566
H) Transferencia Internacional de información.	566
I) Actualizaciones a Nuestra Política de Privacidad.	567

J) Contáctenos.	567
K) Información personal de menores.	567
7. Términos de Uso de la plataforma	568
A) Aceptación de las Condiciones de Uso	568
B) Cambios a las condiciones de Uso	568
C) Definiciones	568
D) Contenido Digital y Plazo de Licencia Limitada	569
E) Sitio Web OMICS	570
F) Condiciones de Uso	571
G) Privacidad	572
H) Propiedad Intelectual y Otros Derechos	572
I) Publicidad	573
J) Suspensión de Acceso	573
K) Terminación	573
L) Ausencia de Garantías	573
M) Limitación de Responsabilidad	574
N) Indemnización	575
O) Restricciones Geográficas y Ley Aplicable; Controversias	575
P) Integridad del Contrato y Divisibilidad	575
Q) Modificación	575
R) Información de Contacto	576
8. Backup	577
8.1. Requisitos	577
8.2. Carga inicial	577
8.2.1. Creación de la base de datos	578
8.2.2. Correr las migraciones	582
8.2.3. Datos de prueba	584
8.3. Realización de backup	585
8.3.1. Manual	585
8.3.2. Automático	586

8.3.3. Volcado de SQL	586
8.3.4. Migraciones	588
8.3.5. Eventos	589
8.3.6. Métodos de Replicación	589
8.4. Recuperación	591
9. Preguntas Frecuentes	593
9.1. OMICS	593
9.2. Cuentas e inicio de sesión	593
9.3. Uso	594
9.4. ¿Cómo publicar en OMICS?	596
9.5. Otros	597
10.1. Agregar a favoritos y like	599
10.2. Aprobar publicación	599
10.3. Cambiar contraseña	599
10.4. Corregir publicación rechazada	599
10.5. Crear comentario, ver perfil y seguir autor	599
10.6. Crear nueva publicación	599
10.7. Editar perfil	599
10.8. Eliminar colección	600
10.9. Eliminar publicación	600
10.10. Rechazar publicación	600
10.11. Recuperar contraseña	600
10.12. Registrarse	600
10.13. Solicitar y aprobar contrato	600
10.14. Suscribir y desuscribir	600
10.15. Ver notificaciones y crear colección	600
10.16. Visualizar publicación aprobada	600
Bibliografía	601
Índice de Figuras y Diagramas	603
Figuras	603

Diagramas	610
Tablas	610



## Marco Teórico

La industria de los cómics es una de las más lucrativas en el mundo del entretenimiento, no sólo en su formato físico, sino también por cómo luego son transferidos a otros medios, tales como los videojuegos y el cine. El impacto actual en la cultura, en cada país y a nivel mundial, es enorme e influye hasta a quienes no son asiduos al medio. En el presente marco teórico no se dispone a dar demasiado detalle de cómo funciona esta industria, tanto como medio de arte y/o entretenimiento, ya que sería un tema demasiado extenso, que para el fin de la plataforma que queremos plantear como proyecto, daremos solamente conceptos claves y sus respectivas fuentes.

### 1. ¿Qué es un Cómic?

Lo primero es dar una definición base, aunque pueden citarse dos definiciones claves, ya que según la Real Academia Española define a los cómics como: <<Serie o secuencia de viñetas que cuentan una historia>> (Española, 2019a); aunque esta definición engloba bastante lo que es el medio que estamos analizando, vemos más certera la definición escrita en un libro que realiza un estudio en profundidad el cual dice que un cómic es: <<Un conjunto de imágenes yuxtaposiciones (Española, 2019b) deliberadamente en secuencia con la intención de proveer información y/o producir una reacción estética>> (McCloud, 1993)

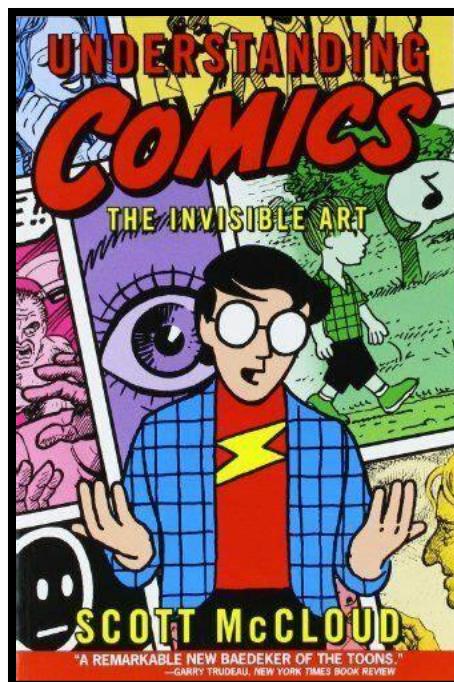


Figura 1. Portada del influyente libro de estudios  
"Understanding Comics: The invisible Art" by Scott McCloud

## 2. Historia del Cómic

El ser humano desde épocas primitivas ha tenido la necesidad de plasmar en imágenes una secuencia de figuras para representar un evento, historia o idea, desde que tiene la capacidad de hacerlo como puede observarse en las pinturas rupestres de las cavernas, en diversas partes del mundo. También se puede mencionar que el uso de imágenes en la antigüedad y el orden de las mismas son las que dieron las bases de cómo se comunica el humano en la actualidad, sin ir más lejos, usamos emoticonos que, según el orden, dan una definición diferente (History of Stuff, 2017).

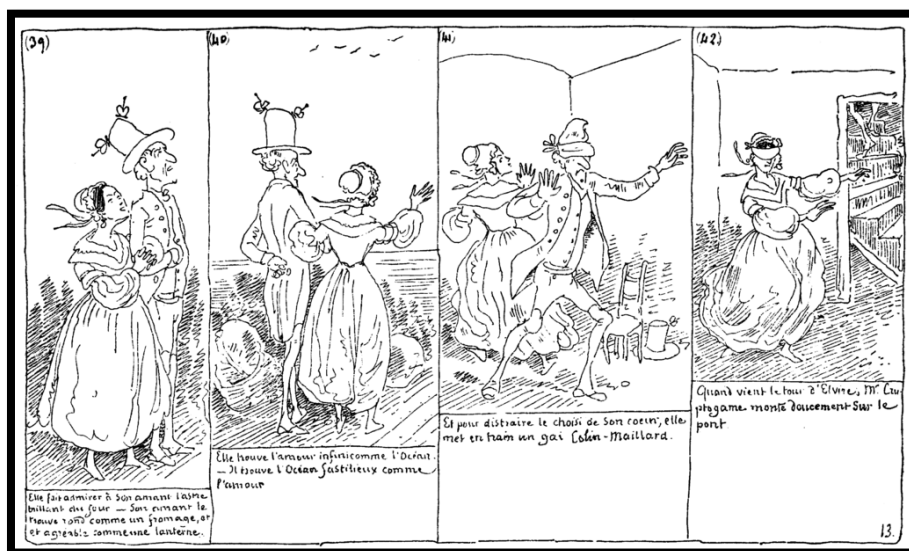


Figura II. Una de las primeras ilustraciones realizadas por Rodolphe Töpffer

Las imágenes impresas para contar historias o transmitir ideas comenzaron a volverse común en Europa en 1827, principalmente por los trabajos ilustrados del suizo Rodolphe Töpffer (Fundación Wikipedia, 2020h), mientras que en Estados Unidos y Canadá comenzó a en los trabajos de Richard Outcault (Fundación Wikipedia, 2020g) con la tira "The Yellow Kid" (Fundación Wikipedia, 2020i) (realizada desde 1895 a 1898) que incluía críticas sociales y políticas para atraer lectores a New York World (diario creado por Joseph Pulitzer (Topping, 2016)). Por otro lado, la evolución en algunos países como Francia y Bélgica se usó la parte ilustrativa más como un agregado a lo narrativo, pero al tiempo el formato comenzó a ser muy parecido al del resto de Europa.



Figura III. *The Yellow Kid's R-R-Revenge; Or, How the Painter's Son Got Fresh* (09/01/1898) por Richard F. Outcault

Además de lo mencionado anteriormente, un caso particular de cómo fue evolucionando este medio, es Japón. Aunque podemos remontarnos a principios del siglo XII, donde se mostraban leyendas y relatos pasados por generaciones en pergaminos, esto se mantuvo casi igual hasta un libro del período Edo (1603-1867) donde se recolectaron en un libro llamado Toba Ehon (Fundación Wikipedia, 2017), esto continuó evolucionando hasta lo que hoy se conoce como el equivalente de cómic en Japón llamado manga (Fundación Wikipedia, 2020c) y que se volvió popular internacionalmente este estilo de ilustraciones en 1951 con las obras Astro Boy (Screen, 2015; Swidlicka, s/f).

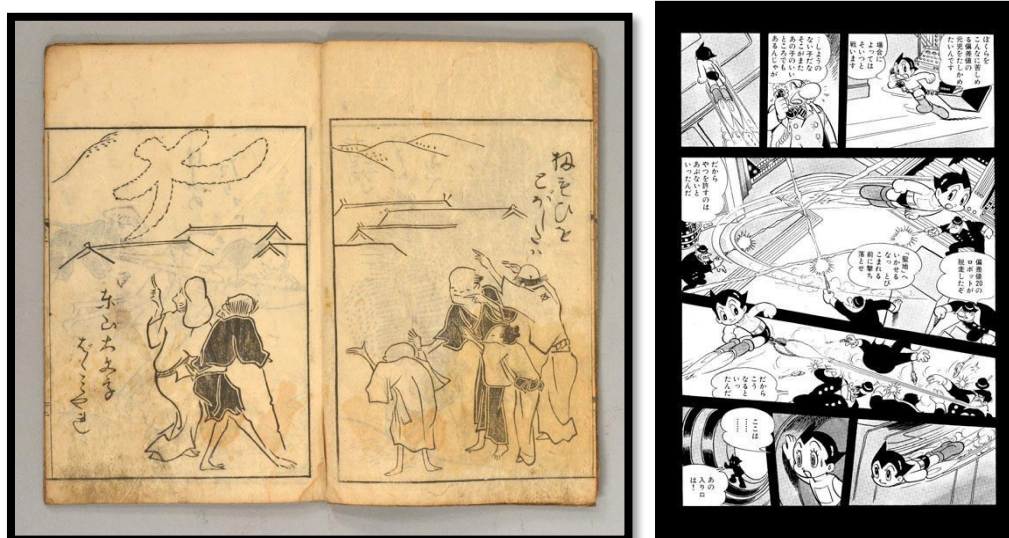
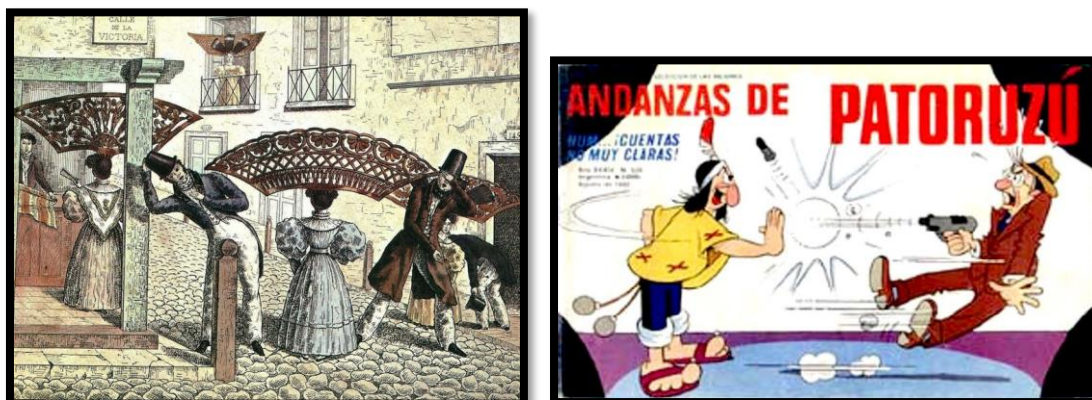


Figura IV. (izquierda) Una de las ilustraciones en el libro Toba Ehon (derecha) Una página ilustrativa del primer número de Astroboy.

Por último, a pesar de que cada región del mundo tuvo su evolución, puede mencionarse como fue evolucionando el concepto de cómic en Argentina. En 1824 aparece la primera viñeta humorística en el país llamado "Viva el Rey", luego en 1834 se realizó una serie humorística sobre las modas de vestimenta de aquel entonces realizado por César Bacle. Recién en 1835 aparece el primer periódico ilustrado de Argentina, pero no empezaría a realizarse en formato de cómic hasta la llegada de "El Mosquito" la cual era considerada como la primera revista con humor político nacional. Comenzaron los diarios a tener viñetas y pequeñas historietas para atraer al público, uno de ellos fue el diario Crítica que adaptó varias obras del extranjero con vocabulario y usos locales. No sería hasta 1928 que en las "Aventuras de Don Gil Contento" aparece un ícono nacional de los cómics: Patoruzú. Los años dorados para este medio en el país fueron entre los '40 y '60 donde personajes como Isidoro Cañones, El Eternauta, y Mafalda. La diferencia más notable que puede notarse en la región de Latinoamérica con el resto del mundo es que el tinte político y religioso está mucho más presente, y puede verse más información al respecto en la página web "Todo Historietas" (Giuanta, s/f).



*Figura V. (izquierda) Ilustración de César Bacle (derecha) Portada de una de las historietas de Las Andanzas de Patoruzú.*

Actualmente los cómics se siguen adquiriendo en formato físico y pueden conseguirse ediciones especiales de aficionados, así como también relanzamientos. Un método muy popular, gracias a las tecnologías actuales que nos dan los dispositivos móviles, son los Webcomic (Fundación Wikipedia, 2020j) que permiten no sólo dejar de depender de un editor y presupuesto más elevado, por el material físico requerido, sino también llegar a mayores audiencias al costo de mayor competencia.

### 3. Diferencias Claves

En la industria de las historias narrativas ilustradas hay unas divisiones que pueden en algunas no ser muy claras pero que marcan un estilo específico para diferenciarlas. Usaremos indistintamente el término cómic, historieta, y novela gráfica para nuestra plataforma para mayor entendimiento del uso del lenguaje de los usuarios, pero cabe aclarar que depende mucho de la región y profundidad de conocimiento que tienen los lectores para realmente distinguirlas (el estilo español es conocido como "Tebeo" mientras que hay una clara diferencia en la forma de realizarse en Francia que se conoce como "bandes dessinées" (*Sutori: Diferencia entre "Cómic" y "Manga", 2012*)).

Muchas de las diferencias vienen tanto del formato físico en que se distribuyen, qué tanto ocupa el texto sobre la imagen y viceversa, el tamaño del equipo necesario para realizarse y hasta la frecuencia en que son lanzados al público.

A continuación, marcaremos cuatro tipos principales y sus características:

#### 3.1. Cómic

Es el uso más generalizado y que la mayoría de las personas usan al referirse a este medio indistintamente del estilo. El término proviene de Estados Unidos donde se hizo muy popular en la década de los años '30 y es de la palabra "comedy" (comedia) ya que usualmente era para un público joven y para hacer reír inicialmente (Fundación Wikipedia, 2020a). Suele presentarse con cubiertas de papel y son lanzamientos continuos (numerados). Generalmente su lanzamiento es mensual, pero depende de la época y popularidad, pueden llegar a ser trimestral o semanal su lanzamiento.

Este estilo requiere pocos cuadros porque sus imágenes son bastante explícitas sobre lo que quieren comunicar y los elementos narrativos son más escasos y cortos. El estilo artístico suele ser caricaturesco y humorístico.

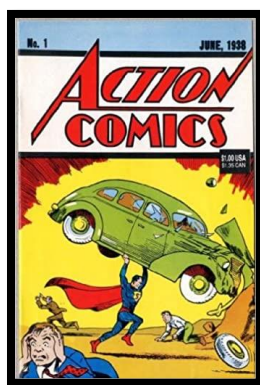


Figura VI. Portada famosa del primer cómic de "Action Comics" donde aparece Superman.

### 3.2. Historieta

Este término suele usarse de forma indistinta en países hispanohablantes como México, Argentina, España y Chile; su origen fue más en forma de viñetas continuas que se lanzaban en diarios y podían leerse sin necesidad de contemplar una cronología (ejemplos de ellos son "Yo, Matías", Diógenes y el linyera, Clemente, Garfield y Dilbert). En Norte América se suelen referir a los mismos como "comic strip".

Las historietas suelen ser de lanzamientos diarios o semanales, con una secuencia cronológica entre sus viñetas, pero no necesariamente entre sus diferentes entregas. Requiere más viñetas para representar lo que se quiere comunicar y sus imágenes generalmente son más artísticas. Su temática es libre y predominan no sólo el humor político, sino drama, terror y suspenso (Diferencias.eu, 2019).



Figura VII. Dilbert, historieta conocida en el mundo de la informática y sistemas.

### 3.3. Novela Gráfica

La mayoría de los lectores experimentados en cómics, suelen leer en verdad lo que es llamado "novelas gráficas" (ejemplos son Watchmen, Maus, El regreso del caballero oscuro, V de Vendetta y Sandman). Estos se diferencian al ser distribuidos físicamente con cubiertas más rústicas o tapa dura, tienen una historia completa que puede o no tener secuelas y su trama consiste en un principio, un nudo y un final; además suelen apuntar a un público maduro y con contenido complejo. Se destinan mayormente a adolescentes y adultos; con temática de todo tipo (ciencia ficción, cotidianos, terror, etcétera), además de contar con un número mucho mayor de páginas que el concepto dado de cómics (Valvasuata, 2014).

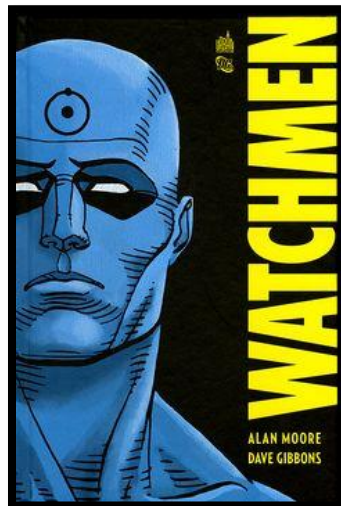


Figura VIII. Watchmen, novela gráfica famosa que revolucionó la industria del cómic.

### 3.3. Manga

Esta categoría es la más clara de diferenciar por su origen oriental, suelen ser de Japón y Corea del Sur, aunque algunos autores occidentales suelen usar su estilo. El término "manga" proviene de la formación de dos kanjis (Fundación Wikipedia, 2019) que literalmente a nuestro idioma sería "imágenes caprichosas o improvisadas" ya que en sus inicios se los consideraba una pérdida de tiempo realizarlos, pero actualmente en Japón y otros países orientales se usa para todo tipo de historietas y comics (en occidente se suele referir a los mismos como "Cómics Japoneses", aunque no provengan de dicho país).

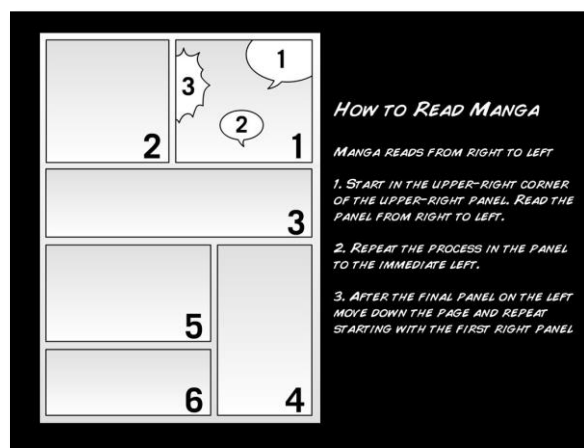


Figura IX. Es común encontrar guías de cómo se lee un manga cuando son traducidos y exportados a Occidente.

Debido a su origen, se leen de derecha hacia izquierda y mayormente están escritos en japonés. Suele tener lanzamientos semanales o mensuales como si fueran

revistas, y si llega a ser muy popular una obra; se realiza un lanzamiento en un tomo único independientes llamados Tankōbon (Fundación Wikipedia, 2018). Otra diferencia clave es que aun siendo populares algunas, se realizan en blanco y negro debido a costes y están tan inculcados en la cultura japonesa que lo realizan equipos llamados "mangakas" y pueden tocar desde temas cotidianos, dramas, terror, acción e incluso se usa para entrenamiento de personal o para informar sobre usos de algún instrumento o tecnología como hizo Microsoft con varias de sus versiones de Windows usando OS-tan (personajes basados en los sistemas operativos usados como mascotas para darse a conocer) (Fundación Wikipedia, 2020f).

Los mangas se orientan a todos los públicos y tienen subcategorías según el contenido que presentan: para hombres, mujeres, ancianos, violencia, eróticos, LGBTIQ+, etcétera; por lo que hasta podría considerarse una plataforma aparte solo para lo que es este tipo de medio (Fundación Wikipedia, 2020e).



Figura X. Es tan fuerte la cultura manga en oriente que incluso los utilizan para presentar productos, guías para aprender a programar y lógica de las computadoras.



## 4. Problemática Actual de la Industria

La industria del cómic ha crecido a punto que existen nichos específicos de lectores que buscan contenidos orientados a sus gustos, la competencia se ha vuelto feroz, ejemplos de ello son Marvel, DC, Jump y Dark Horse, y aún más en estos últimos años debido a la masificación de Internet y las diferentes facilidades que existen para leer y producir contenido.

Los autores sólo pueden llegar a su público si tienen un apoyo muy grande, y se le dificulta a quien recién empieza lograr tener contacto con sus lectores. En Argentina hay muchos que han abandonado la industria de los cómics, al no poder competir debido al costo de papel y el poco acceso que tienen a plataformas online para poder realizar sus publicaciones, además que suelen ser opacados por las grandes obras como Batman, Dragon Ball, Deadpool, etcétera.

Por otra parte, los lectores se encuentran abrumados por la gran cantidad de contenido existente, y los nuevos suelen entrar por las modas que pueden introducirse por otros medios como el cine, evitando explorar otras obras y autores menos conocidos; perdiendo así la posibilidad de encontrar nuevos cómics favoritos y de su interés que compartan con otros lectores.

Muchas de las plataformas actuales no ayudan demasiado debido a que suelen ser desorganizadas y sin filtro alguno, prevaleciendo a veces el erotismo para llamar la atención y no la calidad de los cómics. Las aplicaciones que tienen algún tipo de filtro suelen ser muy específicas con editoriales como DC y Marvel, dejando a autores independientes a la deriva.

# Trabajo Práctico N°1:

## *Desarrollo de un Sistema de Información*

## Definición de Requerimientos

A continuación, se evalúan algunas aplicaciones similares para en primer lugar, tener un punto de comparación y análisis y, en segundo lugar, poder así conformar un criterio para establecer todo lo que puede mejorarse o extenderse en el proyecto Omics y su idea inicial. Se pone énfasis en cada alternativa, en aquellas partes que resulten de mayor interés para el proyecto.

Cabe aclarar que todos los casos de estudio que se analizarán a continuación cuentan con un **módulo de usuario** que permite a los usuarios registrarse y autenticarse para leer las obras (lectores), así como también publicar obras propias (autores). El módulo de usuario y su planteamiento, mediante roles, categorías, y privilegios, es necesario para cualquier plataforma compuesta por múltiples usuarios con distintos niveles de acceso al contenido.

Algo de gran relevancia para el proyecto que se analiza, es la posibilidad de que los autores independientes puedan **publicar** sus obras. Así, como la forma de **suscripción/pago** que tanto la plataforma como los autores puedan recibir como medio de ganancia económica.

Finalmente, se identifica a los **lectores digitales** que provee cada aplicación, sea web o móvil, para la lectura y navegación de cada obra. Por lector digital nos referimos a la tecnología, embebida en cada aplicación (una librería, por ejemplo), que permite visualizar el cómic como si de un libro físico se tratase. Esto es importante debido a que la percepción del usuario, en cuanto a la comodidad, familiaridad y simpleza de la plataforma es una de las razones que lo hacen al usuario continuar utilizando *Omics* o no.

## A. Mangatoon

mangatoon.mobi

### 1. Relevamiento General

Se realiza el análisis de la aplicación web de *Mangatoon (Manga Toon - Lee cómics en español, s/f)*.

#### 1.1. De la Organización

Es una aplicación web, la cual a su vez cuenta con aplicación móvil. Se analiza únicamente la aplicación web, ya que la versión móvil no es más que una extensión de esta misma: es una aplicación híbrida, cuya base funciona tanto en web como en móvil.

*Mangatoon* basa su modelo de negocio en el establecimiento de contratos con autores para dividir las ganancias de las visualizaciones.

Así, tanto la plataforma como los autores se benefician al generar comunidad y ganancias.

**MangaToon es una plataforma de contenido de entretenimiento con más de 10 millones de usuarios en todo el mundo**

**La publicación de historias en MangaToon no solo tienen una gran cantidad de lectores y seguidores, sino que también cuenta con los siguientes mecanismos para garantizar su creación**

 <p><b>Rica remuneración y recompensa</b></p> <p>Soporte completo para creadores Elimina las preocupaciones y crea tranquilidad Vea a continuación para más detalles</p>	 <p><b>Soporte de plataforma profesional</b></p> <p>Equipo de operaciones profesionales promueve trabajos a través de varios canales El equipo editorial profesional ayuda a mejorar la calidad También hay oportunidades para que tu historia se convierta en libro, película o incluso videojuego</p>	 <p><b>Entrar en el mercado internacional</b></p> <p>Emitimos a múltiples países y regiones El equipo de traducción profesional ayuda a que tu historia pueda ser leída en todo el mundo</p>
---	--	---

Figura 1. 1 Publicación y remuneración publicando en Mangatoon



Figura 1. 2 Contrato de obras populares.

De esta forma, a través de las obras populares, aquellas que despierten el interés de los lectores o hayan ganado concursos, el autor puede solicitar un contrato para obtener ganancias económicas.

## 1.2. Funciones Detectadas e Interfaces

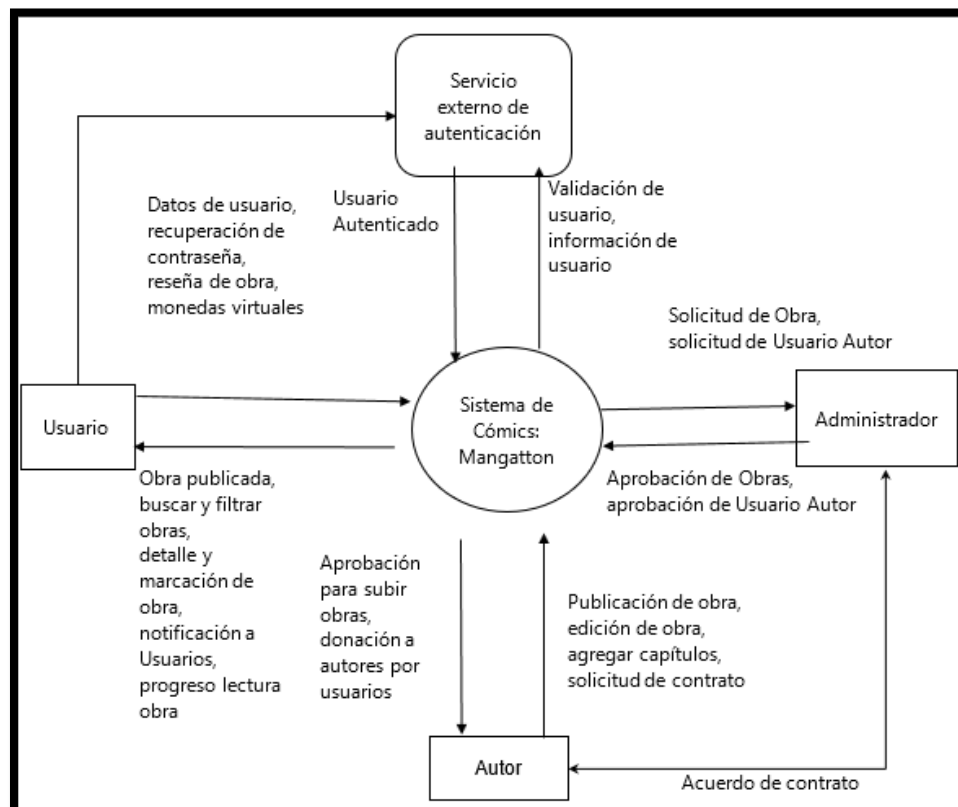


Diagrama 1. 1 Diagrama de contexto de interfaces detectadas en Mangatoon.

Se observa que existen dos niveles de usuario: el lector y el autor. El lector es un usuario común que puede visualizar y valorar los cómics, y el usuario autor es el que sube algún trabajo propio, y puede establecer contrato o no con la plataforma. Existe, además, un usuario administrador, el cual tiene un rol de mayor privilegio que los usuarios mencionados anteriormente, es el encargado de aceptar las nuevas obras publicadas, que están pendientes de aprobación.

Del análisis se extraen las siguientes funcionalidades de la plataforma:

**Funcionalidad 1: Función de login**

- ❖ Registro: Creación de cuenta.
- ❖ Login: Autenticación de usuarios.
- ❖ Configuración de cuenta.

**Funcionalidad 2: Función de catálogo**

- ❖ Catálogo: Presentación de obras filtradas por categorías (temática).
- ❖ Detalle: Información relevante acerca de un cómic seleccionado: título, descripción, valoraciones por otros usuarios, capítulos.

**Funcionalidad 3: Función de lectura**

- ❖ Lectura de cómic: Utiliza una aplicación independiente (visor o lector) para mostrar el cómic en forma de imágenes secuenciales.
- ❖ Dar reseña de cómic: Sólo funciona en aplicación móvil.

**Funcionalidad 4: Función de publicación de obras**

- ❖ Función realizada por el usuario autor de los cómics, historietas, otros.
- ❖ Publicación en panel de gestión web independiente a catálogo.
- ❖ Información del cómic: Título, descripción, derechos de autor y portada. Cada capítulo consta de múltiples imágenes secuenciales, por lo que deben ser subidas al sitio tal cual.

**Funcionalidad 5: Solicitud de contrato**

- ❖ Solicitud de Contrato: Se postula una obra para contrato, el contrato se establece de forma directa, fuera del contexto de la plataforma.
- ❖ Cuando alguna obra alcanza 20 capítulos puede solicitarse un contrato para que se dividan las ganancias de visualización y publicidad, entre otras. Este contrato se pacta directamente entre la plataforma y los autores.
- ❖ El autor, a través del panel de gestión de obras, puede solicitar contrato en cualquier momento, y un administrador atenderá la solicitud. Sin embargo, los acuerdos de contrato se realizan por fuera de la plataforma.

**Funcionalidad 6: Función de aprobación de obras**

- ❖ Función realizada por el administrador de la aplicación.
- ❖ Visualiza obras pendientes de aprobación.
- ❖ Aprueba obras.

**Funcionalidad 7: Función donación a los autores**

- ❖ La función "Donación a autores" por medio de la utilización de monedas virtuales compradas en la misma plataforma (que tiene una equivalencia en dinero transferido del usuario por tarjeta de crédito, débito o lo que acepte la plataforma).
- ❖ El usuario al ver una obra puede seleccionar el "recompensar al autor" y elegir un monto de monedas virtuales, además de ver una lista de "rankings" de los que más aportaron, de esa manera se tiene una lista de competencia entre los mismos usuarios de quien apoya más al autor.

**1.3. Tecnología de Información**

En cuanto a front-end, como aplicación web, podemos destacar que el renderizado completo, tanto del catálogo como del detalle de un cómic, se realiza del lado del servidor. Para el desenvolvimiento dinámico del sitio se utiliza jQuery (v1.11) como librería, aunque para cosas puntuales, como reacción a eventos o mostrar un carousel con cómics destacados utiliza JavaScript junto a CSS.

Tanto el catálogo como el panel para publicar obras son renderizados en el lado del servidor, gracias a PHP, y utiliza jQuery para funcionalidades dinámicas, sobre todo visuales y validación de campos para entrada de datos.

El back-end está desarrollado en PHP 7.3.7, utilizando OpenResty (openresty.org) como servidor de aplicaciones, basado en NGINX.

Se puede observar en **FIGURA 1.3** la respuesta a una consulta HTTP requiriendo un documento HTML. Los encabezados nos indican, también, la tecnología utilizada por el sistema.

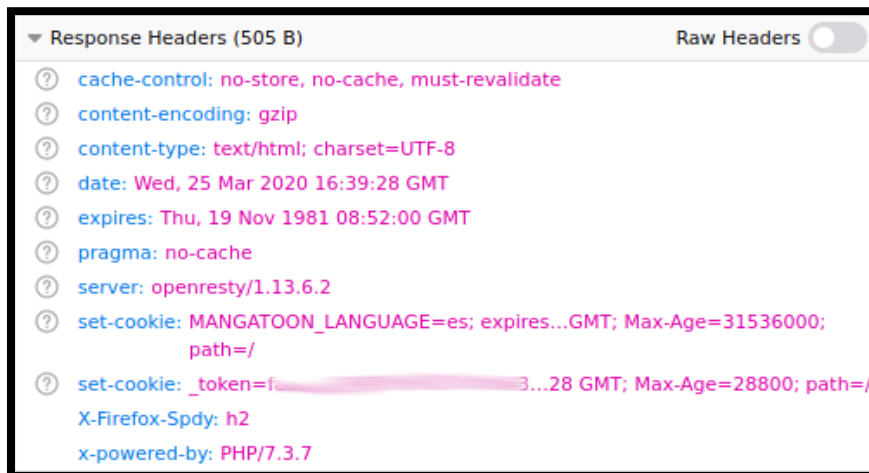


Figura 1. 3 Respuesta del servidor de Mangatoon.

Por último, se puede pensar que el lector digital (VER FIGURA 1. 4) es en sí una aplicación web haciendo uso de librerías complejas para el tratamiento y visualización de imágenes, proveyendo de una navegación fluida (guiada por gestos) para el usuario. Pero en un análisis detallado del código fuente podemos ver que funciona de igual forma a lo descrito para el front-end (catálogo y publicador de obras): renderizado en servidor con PHP, uso de jQuery para dotar de dinamismo a elementos visuales.

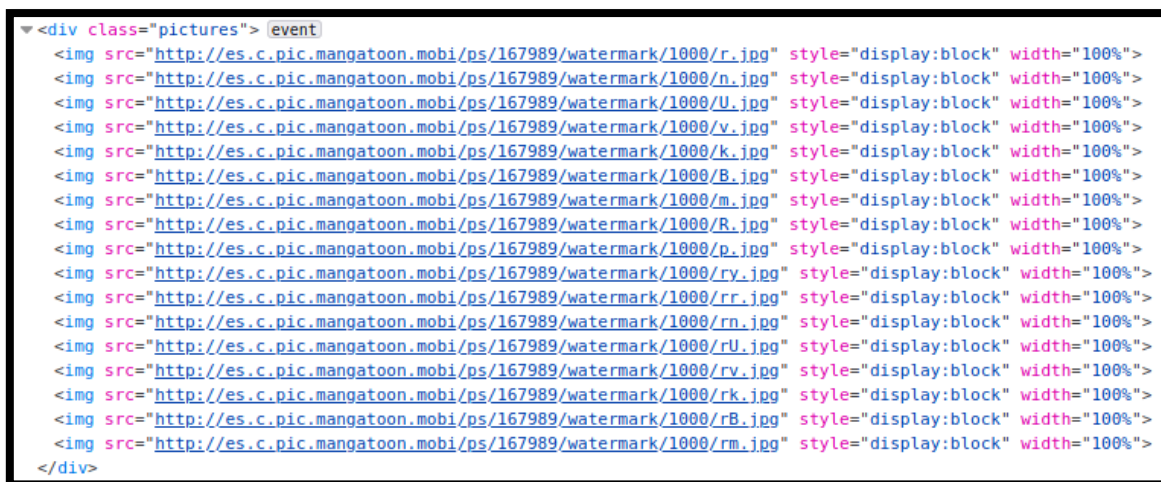


Figura 1. 4 Carga de imágenes secuenciales en lector de Mangatoon.

Las imágenes se obtienen del back-end directamente, colocando una a continuación de la otra.

En resumen, el catálogo, detalle de cómic, lector y panel de gestión para publicar obras son renderizados en el lado del servidor con PHP y utiliza jQuery. El back-end está desarrollado en PHP 7 y utiliza un gestor de base de datos relacional, tal como MySQL.



Cada componente front-end del sitio es responsive, esto significa que es adaptable a múltiples resoluciones de pantalla. Esto hace posible la empaquetación del sitio como aplicación híbrida y su facilidad de publicación en tiendas de aplicaciones móviles.

El registro y autenticación de usuario es realizado mediante OAuth, haciendo uso de múltiples servicios de usuario.

## 2. Relevamiento Detallado y Análisis del Sistema

### 2.1. Detalle, explicación y documentación detallada de todas las funciones seleccionadas

Del sitio se seleccionan todas sus funcionalidades analizadas en el punto 1.2, debido a que resultan de interés para el proyecto Omics.

#### Funcionalidad 1: Función de login

Destaca la gran variedad de opciones para el registro de usuario y su posterior login. En el punto 1.3 anterior se destaca la tecnología utilizada.

Las primeras 6 opciones de registro hacen uso de proveedores de autenticación externos, utilizando OAuth e información del usuario almacenada en dichos servicios. Por esto, si el usuario hace uso de cualquiera de estos servicios, la plataforma accede directamente a la información ya existente del mismo, para poblar la base de datos.

Al realizar la autenticación externamente, la plataforma no necesita almacenar la contraseña del usuario. Exceptuando la última opción de *login*: usuario y contraseña. En este sistema se ha mantenido esta última opción únicamente para administradores del sitio, y por esto mismo no existe una opción para recuperar la contraseña.

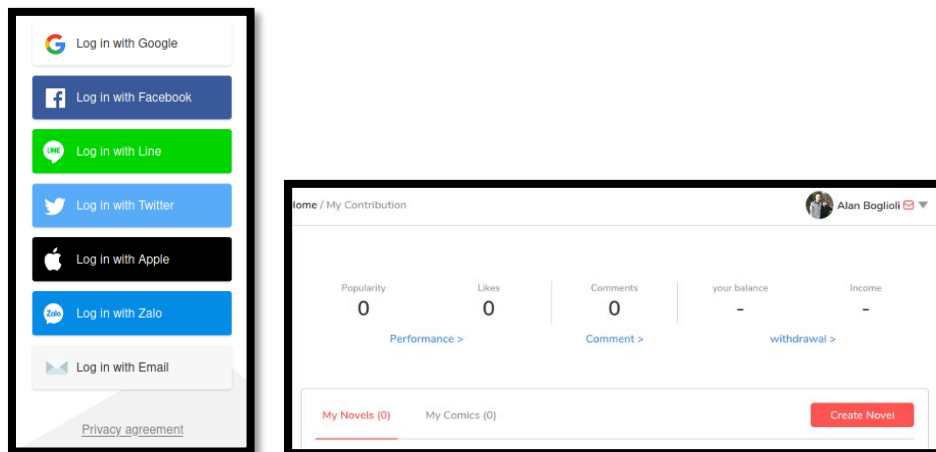


Figura 1. 5 Opciones de registro y login en Mangatoon. Panel de gestión de cuenta.

#### Funcionalidad 2: Función de catálogo

El primer acercamiento al sitio web muestra un catálogo seleccionado de acuerdo a las obras más visitadas y cómics más buscados.

Este catálogo es responsivo, adaptable a cualquier resolución de pantalla. [VER SECCIÓN 1.3 TECNOLOGÍA DE INFORMACIÓN](#) para un análisis detallado.

En pantallas de grandes resoluciones muestra una cabecera con los títulos destacados del momento, la cual es actualizada diariamente. Bajo esta se encuentran, en forma de grilla, las obras que son tendencia entre los lectores, actualizada automáticamente.

A continuación de estas dos secciones, se encuentran las obras nuevas publicadas recientemente.

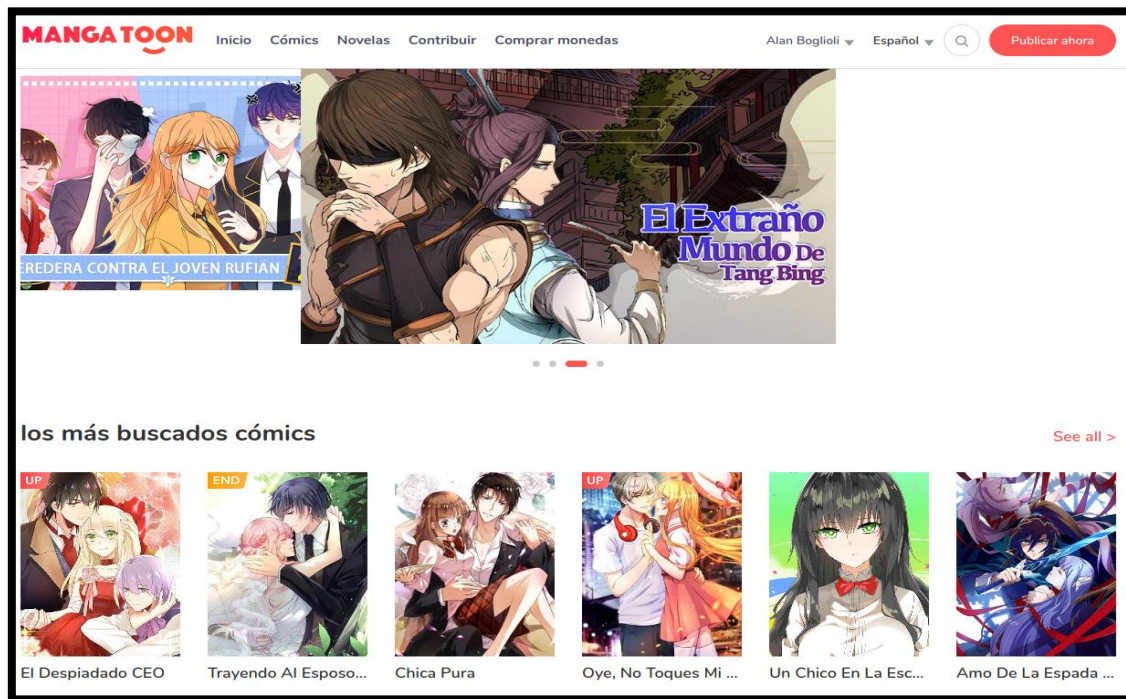


Figura 1. 6 Pantalla de inicio de Mangatoon.

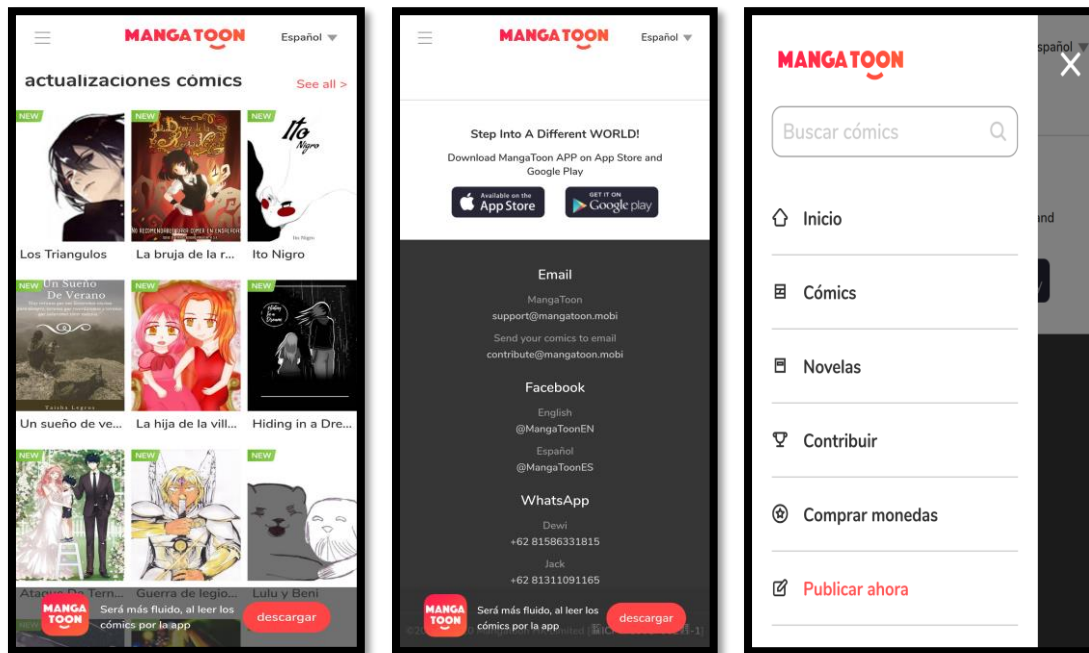


Figura 1. 7 Catálogo en dispositivos móviles (izquierda) y menú de navegación (derecha).

Podemos concluir que *Mangatoon* cuenta con un amplio catálogo categorizado por temática, pero lo más interesante es que propone dos soluciones favorables para los autores de cómics. Cualquier usuario puede comprar monedas con dinero real, éstas conforman el dinero ficticio que se maneja en el sitio para la donación a autores.

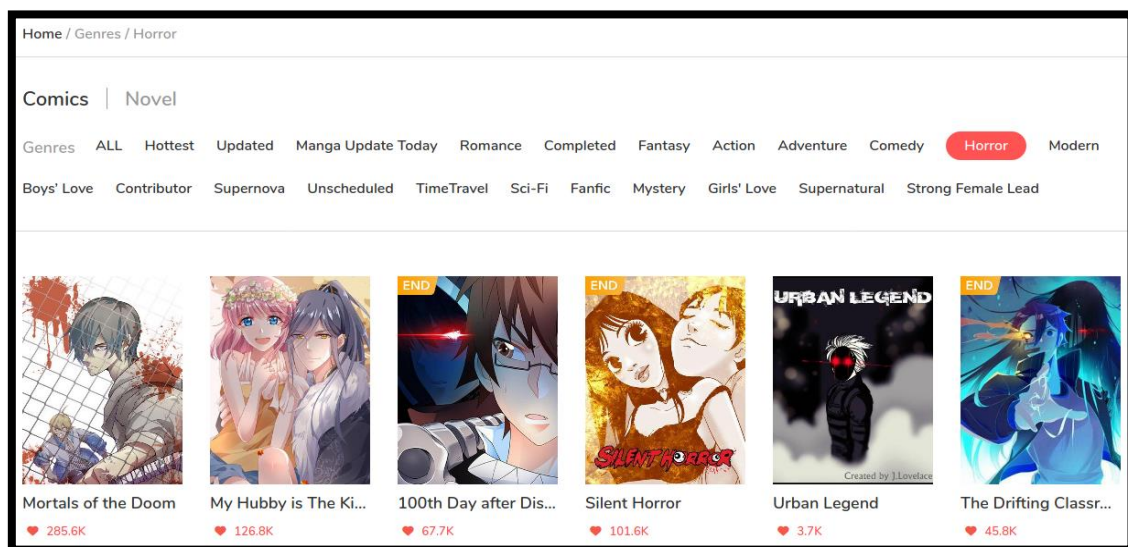


Figura 1. 8 Catálogo de Mangatoon.

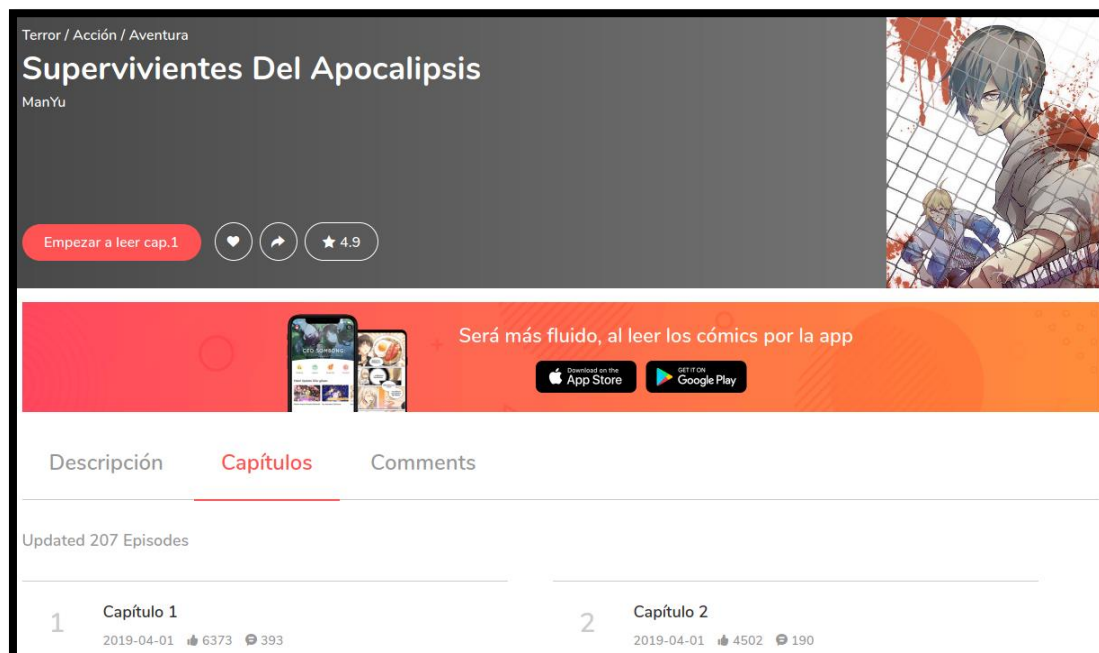


Figura 1. 9 Detalle de un cómic.

El detalle de cada cómic cuenta con la descripción completa, la cantidad de capítulos, y la posibilidad de valorarlo (añadir un *like* o puntuación).

Como ya se mencionó, la aplicación cuenta con versión para móvil también, pero tiene la misma funcionalidad que la aplicación web. Esto es porque utiliza tecnología para generar aplicaciones híbridas. Sin embargo, agrega algunas funcionalidades interesantes como la posibilidad de destacar las obras para conformar una librería personal, y así tener rápido acceso para una lectura posterior.

### **Funcionalidad 3: Función de lectura**

El lector de cómics es básico y adaptado únicamente a pantallas de celulares, como se aprecia en la **FIGURA 1. 10**.

Provee la funcionalidad de avanzar entre capítulos, cada capítulo compuesto por imágenes contiguas, por lo que el único método de navegación entre viñetas es desplazándose (scroll) hacia abajo.

La **FIGURA 1. 10** se obtuvo accediendo de navegador, pero como se menciona anteriormente, el lector está destinado a ser utilizado en dispositivos móviles con pantallas de menores resoluciones.

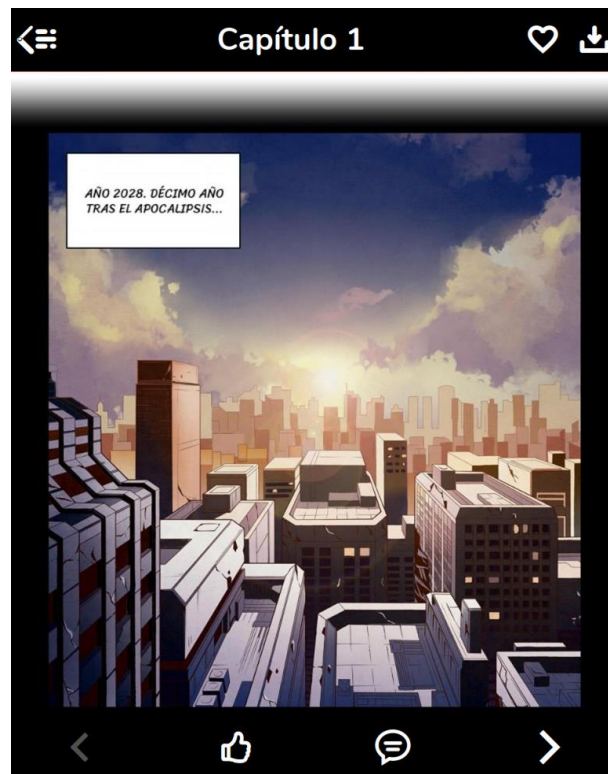


Figura 1. 10 Lector de cómics en Mangatoon.

#### **Funcionalidad 4: Función de publicación de obras**

La publicación de obras se realiza únicamente por web. Posee una aplicación híbrida para móviles que cuenta con el catálogo de obras y un lector integrado.

“Publicar ahora” (**VER FIGURA 1. 11**), del menú principal, permite a cualquier usuario publicar sus cómics, ya sea para obtener rédito o simplemente dar a conocer sus obras. La publicación y gestión de obras se lleva a cabo en una aplicación web independiente, aunque integrada en el sitio.

Se resalta la posibilidad de elegir el idioma del sitio web, y, por tanto, de cómics que soporten múltiples idiomas.



Figura 1. 11 Destaca botón para publicar obra en Mangatoon.

La publicación (**VER FIGURA 1. 12**) es gestionada por una aplicación web separada del catálogo principal, esto es para hacer más cómodo el proceso partiendo de una interfaz simple en lugar de llamativa.

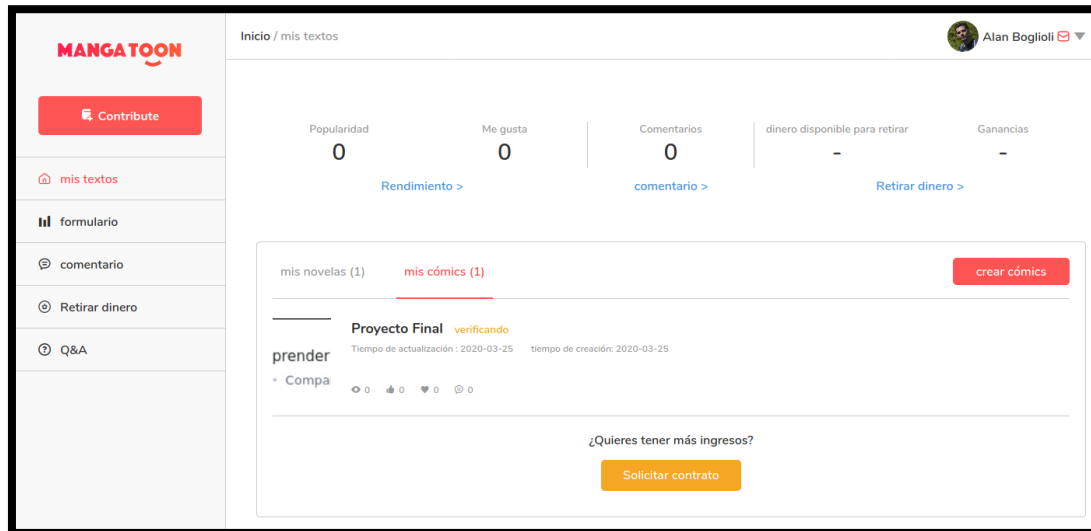


Figura 1. 12 Publicación de obras en Mangatoon.

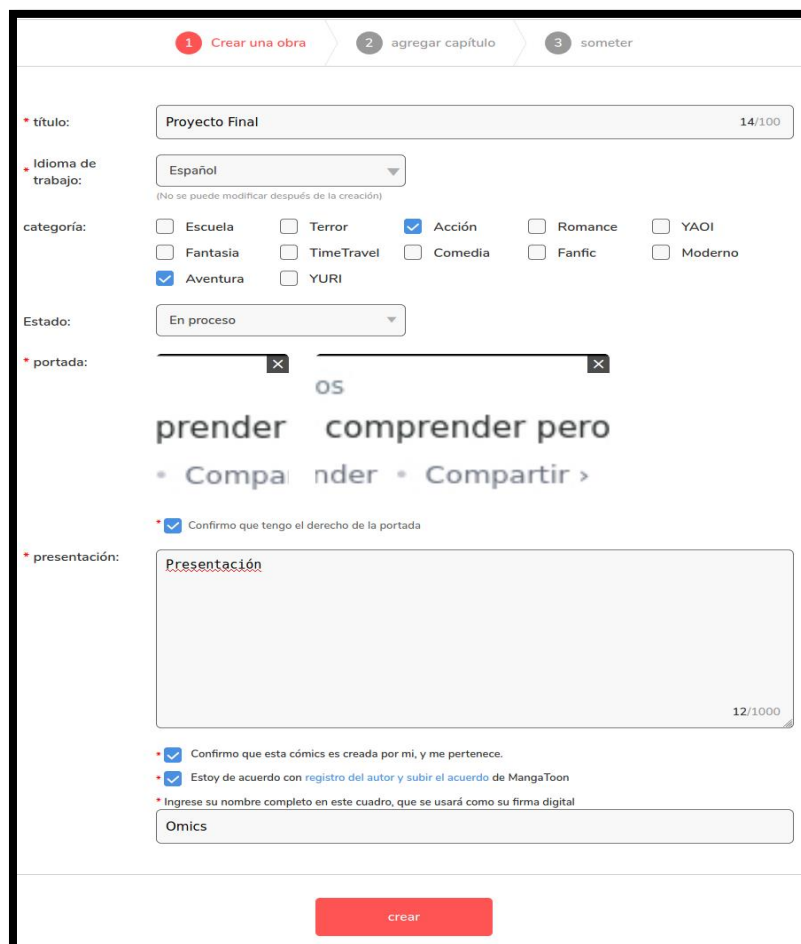


Figura 1. 13 Datos de obra a publicar en Mangatoon.

La creación o carga de un nuevo cómic está guiada a través de simples pasos. En el primer paso se requiere la información básica del cómic: título, categoría, estado (ya que se permite almacenar borradores) y la aceptación de términos referentes a derechos de autor.

Una vez establecida la información necesaria para diferenciar al cómic como obra dentro de la aplicación, se deben agregar capítulos al mismo cargando imágenes en cualquier formato de uso común: JPG, PNG, GIF.

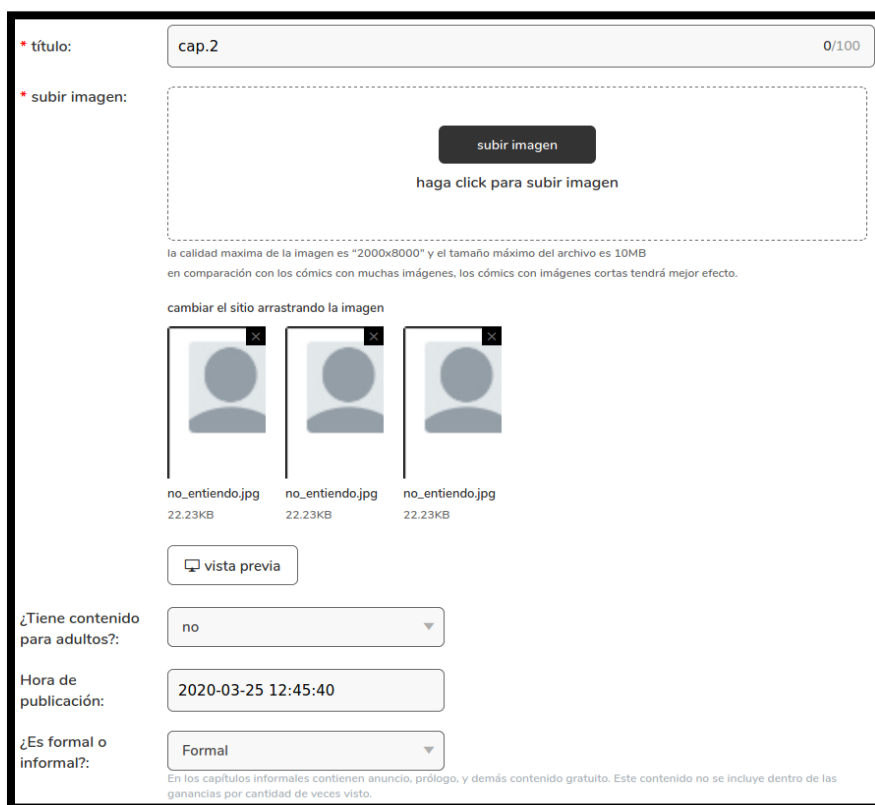


Figura 1. 14 Subida de imágenes que representan las viñetas en Mangatoon.

Each picture is up to 2000 x 8000 pixels and the file size is no more than 10MB.  
 Upload comics in shorter segments will load faster and have better results

Figura 1. 15 Tamaño máximo de imagen a subir en Mangatoon.

### Funcionalidad 5: Solicitud de contrato

Cuando alguna obra alcanza los 20 capítulos el autor, a través del panel de gestión de obras, tiene la posibilidad de solicitar el contrato en cualquier momento, y un administrador atenderá dicha solicitud. El autor solicita un contrato para que se dividan las ganancias de visualización y publicidad, entre otras. este contrato se pacta



directamente entre la plataforma y los autores. Sin embargo, los acuerdos de contrato se realizan por fuera de la plataforma.

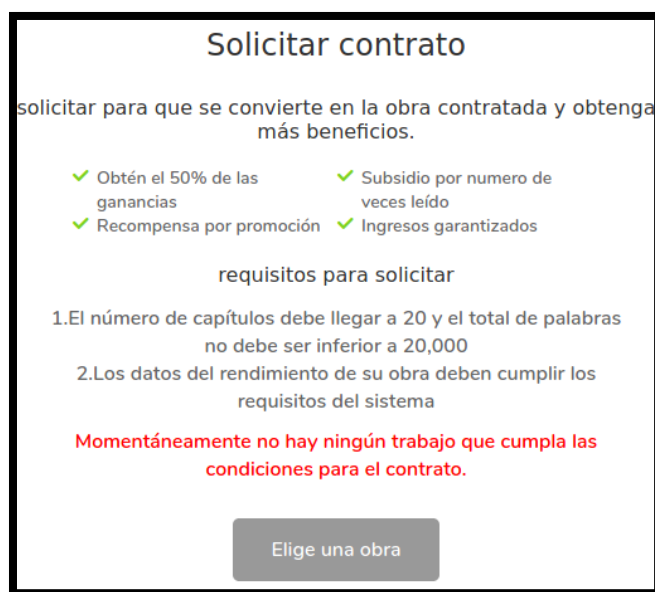


Figura 1. 16 Beneficios del contrato en Mangatoon.

La solicitud de contrato puede automatizarse, pero llegar a un acuerdo entre la plataforma y los autores no sería sencillo, de todas formas, requiere comunicación directa.

## 2.2. Modelo Lógico del Sistema Actual

Un cómic se compone, además de un título, una descripción, y del autor, de una portada y múltiples capítulos. Cada capítulo puede estar conformado por una o más imágenes, las cuales se muestran de forma consecutiva, verticalmente, en el lector digital.

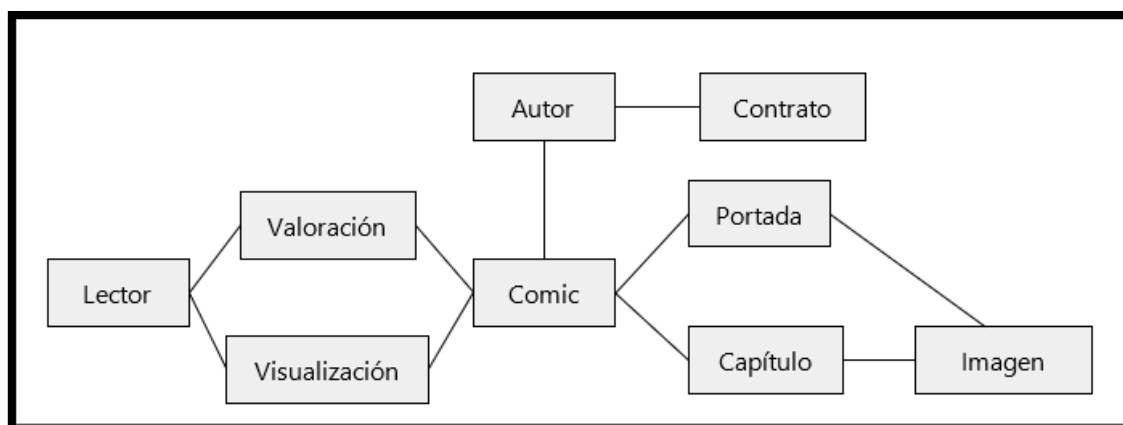


Diagrama 1. 2 Modelo de entidades de Mangatoon.

El **DIAGRAMA 1.2** es una aproximación genérica de la relación de entidades que posee la plataforma de Mangatoon. Se muestra los tipos de usuarios que se puede encontrar y los datos básicos de un cómic.

En el **DIAGRAMA 1.3** se visualiza cuáles son las principales funcionalidades detectadas, como casos de uso. Se ha agregado notas para indicar la posible agrupación de estos, para representar los módulos de la plataforma.

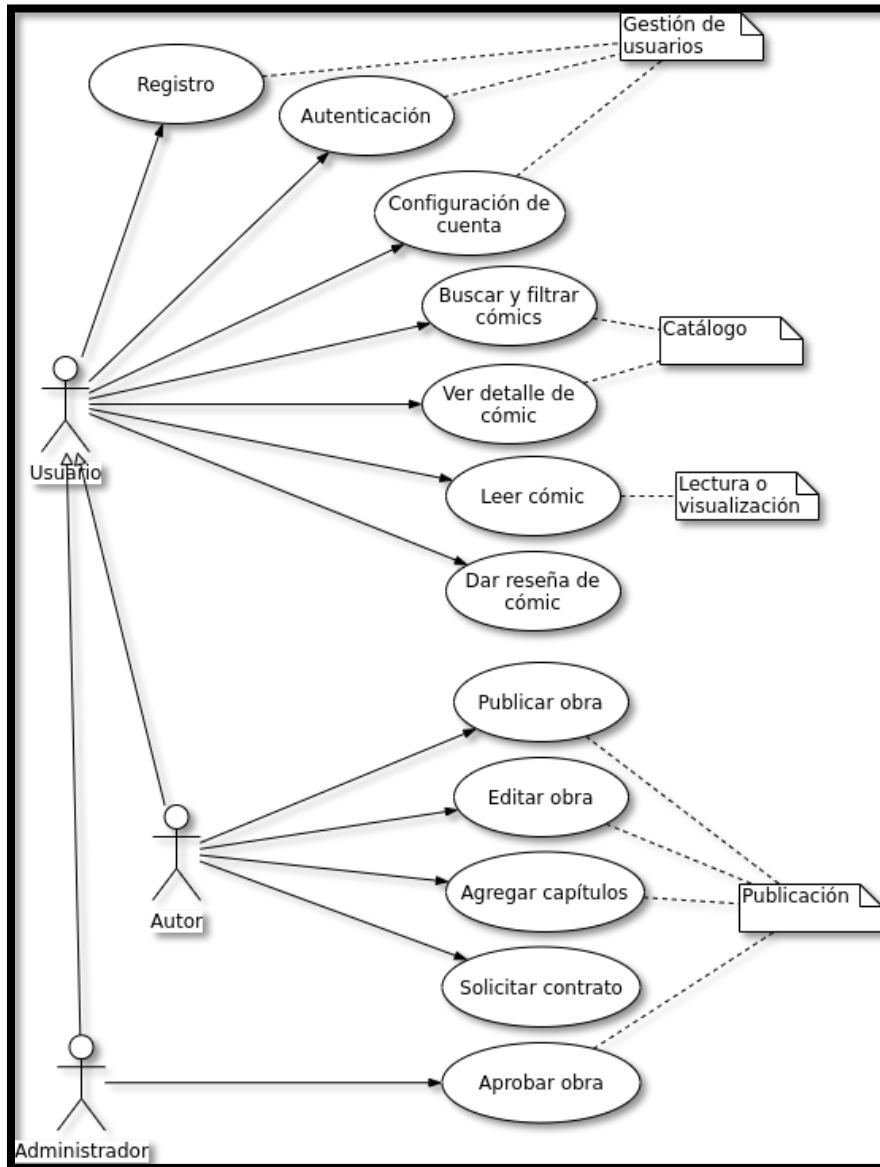


Diagrama 1.3 Diagrama de Casos de Uso general de Mangatoon.

## 2.3. Problemas y necesidades detectadas en las funciones relevadas en detalle y en su entorno organizacional

### **Funcionalidad 1: Gestión de cuenta de usuario**

#### Necesidad de opciones para gestión de cuenta:

Si bien se ofrece la posibilidad de ingresar a la plataforma, utilizando diferentes proveedores de autenticación, la aplicación carece de panel para gestionar la cuenta de usuario. No se permite cambiar el nombre de usuario, el email o la contraseña, y peor aún para autores, subir una imagen de perfil. El sistema depende completamente de los proveedores de autenticación OAuth para obtener la foto de perfil.

### **Funcionalidad 2: Función de Catálogo**

#### Problema de categorías:

El abanico de categorías para filtrar dentro del catálogo es limitado, y no tiene una correlación entre las mismas.

#### Necesidad de variabilidad de contenido:

No es un problema de funcionalidad en sí, sino derivado de gestión y apertura a nuevos contenidos, el catálogo ofrece demasiados títulos referidos a "Romance".

### **Funcionalidad 3: Función de lectura**

#### Problema de carga de imágenes:

El principal problema con la visualización de los cómics radica en la forma en que las imágenes son cargadas, es decir de forma secuencial, todas al mismo tiempo, lo genera que se desaprovecha el ancho de banda del usuario.

Un capítulo está compuesto por múltiples imágenes, el tamaño máximo de imagen es de 10 MB. Si el capítulo de una publicación se compone de 20 imágenes de 5 MB, el lector debe descargar 100 MB para poder visualizarlo, esto produce la necesidad detallada a continuación.

#### Necesidad de procesamiento de imágenes:

El tratamiento de imágenes al momento de la subida es imperativo para optimizar el funcionamiento del lector digital, se requiere procesamiento de estas para adaptarlas correctamente.

#### Problema de valoración de obras:

Si bien el lector digital muestra botones para valorar la aplicación (*like*) o dejar un comentario, estas funcionalidades están disponibles únicamente en la versión móvil.

La versión web presenta la misma interfaz, sin embargo, posee muchas funcionalidades desactivadas, esto es desalentador para los usuarios que acceden mediante navegador en computadoras de escritorio.

Necesidad de funcionalidad para realizar donaciones y valoraciones en web:

Como se describe en el problema anterior, a pesar de estar presentes los botones para donar y valorar un cómic, su funcionalidad no está habilitada, en su lugar, se induce al usuario a descargar la aplicación. Esto genera desconfianza ya que se pueden comprar créditos para donar a autores en el mismo sitio web.

**Funcionalidad 4: Función de publicación de obras**

Problema de categorías:

Las categorías en donde se publican obras son limitadas.

Necesidad de etiquetas:

Las categorías deben ser estáticas, gestionadas únicamente por administradores. Pero un sistema de etiquetas, donde el usuario pueda definir manualmente o añadir otras ya existentes, facilita la búsqueda y el filtrado de contenido en el catálogo.

## B. Tapas

tapas.io

### 1. Relevamiento General

#### 1.1. De la Organización

Es una aplicación web, la cual a su vez cuenta con aplicación móvil, se analiza únicamente la aplicación web, ya que la aplicación móvil no es más que una extensión de esta misma: es una aplicación híbrida, cuya base funciona tanto en web como en móvil.

*Tapas (Tapas: Bite-sized stories you can't find anywhere else, s/f)* le permite a cualquier usuario publicar sus propias historias de manera simple y gratuita, sin recibir asesoramiento de los encargados de la página. En el caso, de que quieran tener soporte técnico y mayor visualización de sus historias, la página ofrece monetizar su contenido. Esta opción ayuda a que los creadores de las historias reciban publicidad cuando los lectores consumen sus cómics o novelas.

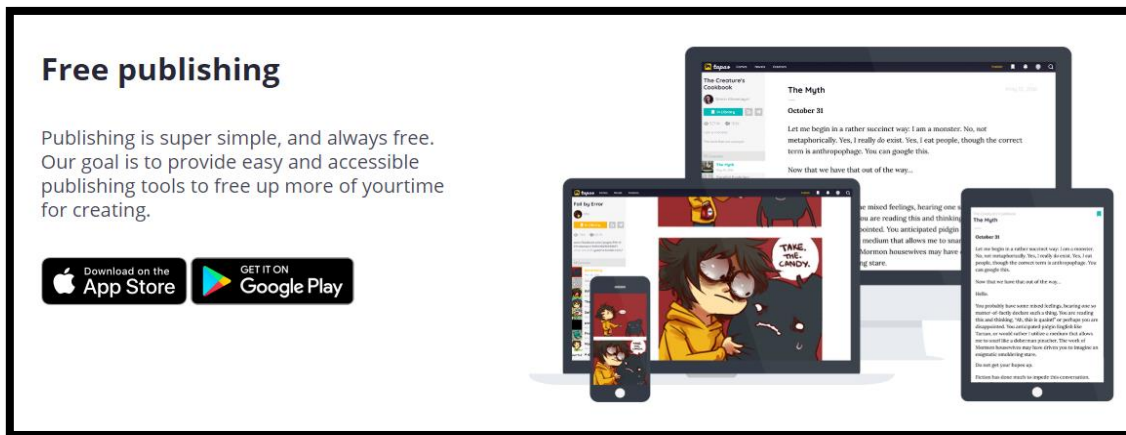


Figura 1. 17 Publicar en Tapas.

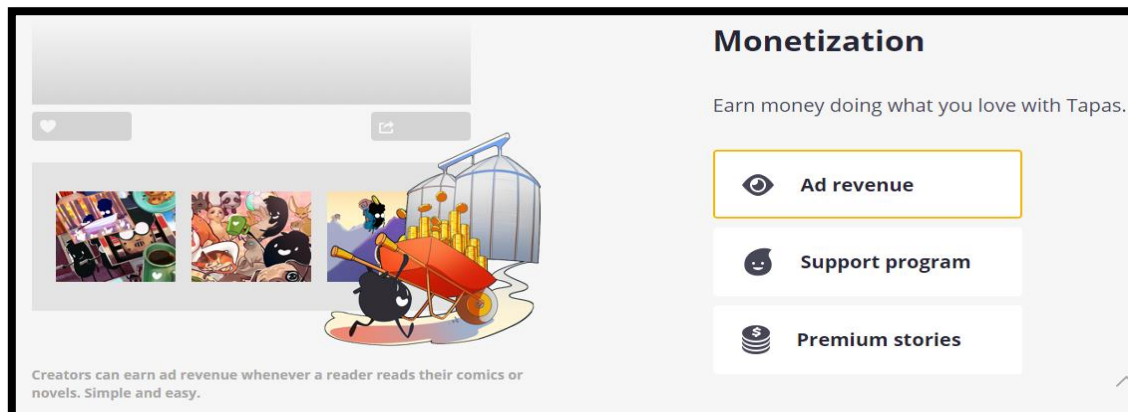


Figura 1. 18 Monetizar publicaciones en Tapas.

## 1.2. Funciones Detectadas e Interfaces

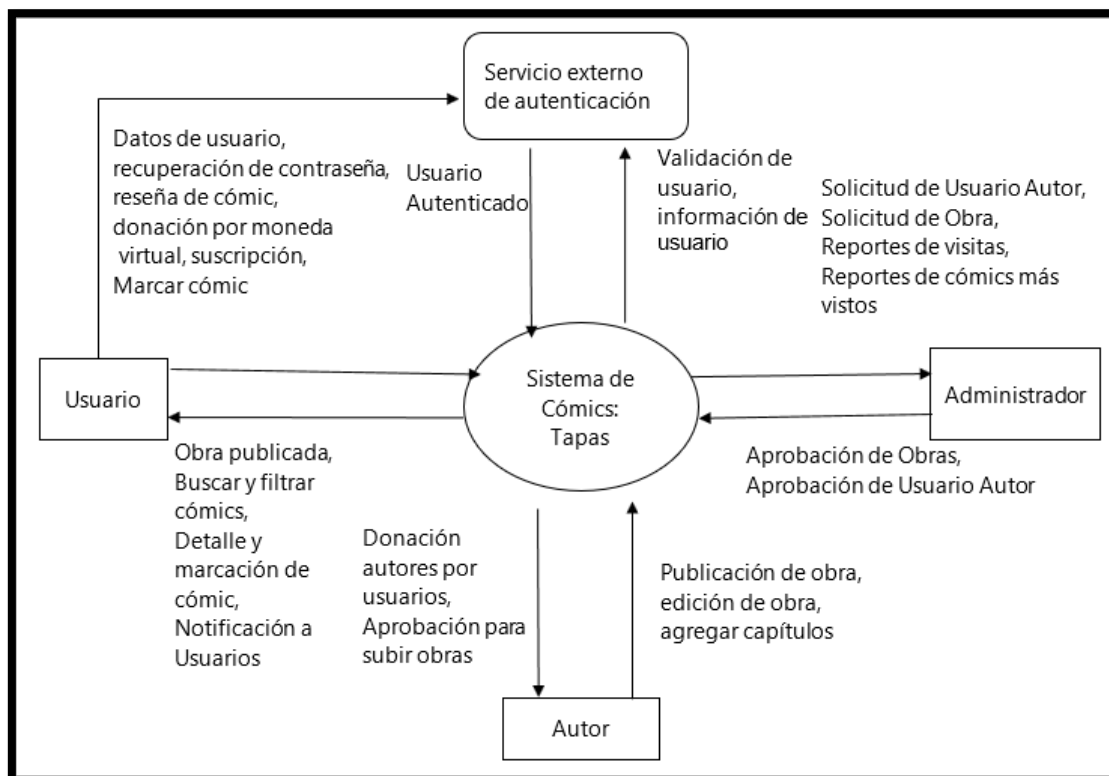


Diagrama 1. 4 Diagrama de contexto de interfaces detectadas en Tapas.

Se detectan dos únicos usuarios: usuario lector quien también tiene la posibilidad de publicar, y el usuario administrador, encargado de moderar los perfiles de usuario, y aceptar/rechazar comics pendientes de revisión.

Del análisis se extraen las siguientes funcionalidades fundamentales que tiene Tapas:

### **Funcionalidad 1: Gestión de usuario**

- ❖ Autenticación y gestión del usuario mediante email y contraseña, o mediante OAuth utilizando los servicios de Facebook y Google.
- ❖ Gestión de cuenta de usuario y perfil de él.
- ❖ Posee una funcionalidad de *microblogging* (al estilo de Twitter) para que tanto lectores como autores puedan publicar estados. A partir del perfil de un usuario se puede seguir su actividad, como la publicación de obras.

### **Funcionalidad 2: Autenticación y recuperación de contraseña**

- ❖ Registro de usuario con email y contraseña únicamente.
- ❖ Ingreso (autenticación) de usuario con email y contraseña.
- ❖ Confirmación de email al registrarse nuevo usuario: Se envía un email automáticamente al nuevo usuario con un enlace único para que verifique la propiedad de la dirección de correo electrónico que ingresó.
- ❖ Recuperación de contraseña: Se le envía un email al usuario con un enlace único autogenerado para que ingrese una nueva contraseña, cambiando la anterior.

### **Funcionalidad 3: Función de lectura**

La lectura de cómics o novelas por parte del usuario lector se centra en un lector digital con capacidades para navegar las viñetas y episodios.

- ❖ Catálogo: Presentación de obras filtradas por categorías (temática).
- ❖ Detalle: Información relevante acerca de un cómics o novela seleccionada: el autor, las visitas, los likes, la cantidad de usuarios suscriptos y cuáles otras creaciones pertenecen al autor.
- ❖ Suscripción a cómics: Para seguir la actividad y futuras actualizaciones, o agregados de episodios.
  - Recepción de notificaciones.
- ❖ Lectura de Cómics o novela: Utiliza una aplicación independiente para mostrar el cómic en forma de imágenes secuenciales, da la opción de dar "me gusta" o comentar la página que se está leyendo.

### **Funcionalidad 4: Función de publicación de obras**

- ❖ Publicación de historias.
- ❖ Publicación en panel independiente a catálogo.
- ❖ Información del cómics o novela: Título, descripción, portada, categoría, entre otros, cada capítulo consta de múltiples imágenes secuenciales, por lo que deben ser subidas al sitio tal cual.

### **Funcionalidad 5: Función donación a los autores**

- ❖ Recibir donaciones: De forma clara, utilizando una moneda virtual propia de la plataforma, es la funcionalidad que más destaca en comparación con el caso anterior.

### **Funcionalidad 6: Función de marcadores**

- ❖ Agregar obras a "Favoritos" para posteriores lecturas. Permite a un usuario armar su biblioteca personal de cómics.

### **Funcionalidad 7: Notificaciones**

- ❖ En este apartado el usuario puede seleccionar diferentes tipos de notificaciones de nuevos episodios, estrenos de obras esperadas, mensajes de otros usuarios o de eventos que desarrolla la plataforma.

## **1.3. Tecnología de Información**

El sistema se divide en aplicación web (front-end) y servidor (back-end).

El back-end utiliza NGINX como su servidor de aplicaciones. Utiliza Cloudfront como CDN para almacenamiento de imágenes. De esta forma, la carga del servidor principal que soporta al back-end se utiliza únicamente para procesamiento, y los archivos estáticos son cargados desde un CDN aparte.

En la funcionalidad de autenticación y recuperación de contraseña se detalla la utilización de un hash criptográfico para el almacenamiento de la contraseña de usuario. Se utiliza en la actualidad, bcrypt (192 bits) para el *hashing* mediante un número aleatorio (*salt*).

Uso de tecnología OAuth para el ingreso de usuarios mediante proveedores externos.



## 2. Relevamiento Detallado y Análisis del Sistema

### 2.1. Detalle, explicación y documentación detallada de todas las funciones seleccionadas

#### Funcionalidad 1: Gestión de usuario

Se selecciona esta funcionalidad como de interés ya que la plataforma provee un perfil de usuario como plataforma de *microblogging* para publicar estados, avances y actividades. De esta manera, un autor puede compartir pensamientos con sus lectores, además de avances en sus trabajos. La gestión de usuario se realiza desde esta misma pantalla.

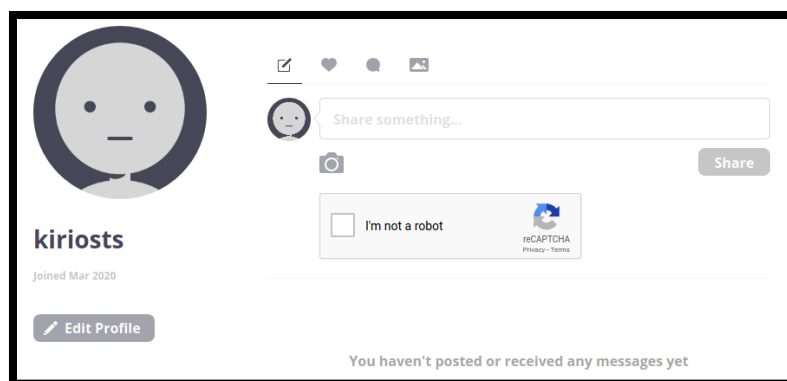


Figura 1. 19 Perfil de usuario en Tapas.

#### Funcionalidad 2: Autenticación y recuperación de contraseña

La aplicación permite la autenticación mediante múltiples plataformas, donde el usuario se acredita con credenciales externas, la última opción requiere que el usuario se registre en la plataforma ingresando sus datos: dirección de correo electrónico, y contraseña. Tanto para el ingreso o el registro por primera vez se requieren los mismos dos datos fundamentales.

Una vez autenticado, el usuario podrá gestionar su cuenta modificando todos sus datos no especificados al momento del registro, como el nombre.

Algo importante a destacar es que al registrarse por primera vez (*Sign up*) se especifica que, al realizar esta acción, el usuario confirma que está de acuerdo con los términos y condiciones de uso.

**Log in to Tapas**  
Read bite-sized episodes of comics and books, or share your own!

Continue with Facebook

Continue with Google

Log in or sign up with email

By continuing, you agree to our [Terms of Use](#)

**Log in**  
Don't have a Tapas account? [Sign up.](#)

Email  
name@email.com

Password [Forgot password?](#)  
password

Log in

Or log in with Facebook or Google

**Sign up**  
Already have a Tapas account? [Log in.](#)

Email  
name@email.com

Password  
password

Sign up

By signing up, you agree to our [Terms of Use](#)

Figura 1. 20 Autenticación (ingreso y registro) en Tapas.

Cuando el usuario se registra por primera vez, debe confirmar su correo electrónico, para evitar así la creación indiscriminada de cuentas. A continuación, se muestra el email de confirmación con un enlace autogenerado para confirmar la dirección de correo electrónico.

Hi [redacted],

**Confirm your Tapas account**

Please click below to confirm your Tapas account.

Confirm Email

Or, you can click this link:  
<https://tapas.io/account/confirm-email/55d0949ed453347ce3e89c3ffa2d430d>

Figura 1. 21 Confirmación de email luego de registro en Tapas.

De esta confirmación se destaca el enlace autogenerado. Tanto el botón en amarillo como el enlace que aparece debajo de este dirigen a la misma URL.

Este enlace se compone de un identificador único como último parámetro, componiéndose como: `tapas.io/account/confirm-email/:id-unico`. Donde *email* es la parte que representa el nombre de usuario dentro en el dominio de la dirección de correo electrónico, e *id-único* es una colección de caracteres alfanuméricos de longitud extensa generados de forma automáticamente utilizando alguna función particular.

Lo importante de este ID es que sea único, irrepetible y extenso, para evitar fraude, principalmente, debe ser imposible de predecir.

En **Login**, se requiere el ingreso de dirección de email y contraseña para validar al usuario.

En la pantalla de ingreso se observa un enlace para **recuperar la contraseña** (*¿Forgot password?, ¿Olvidó la contraseña?*). En esta sección sólo se requiere al usuario que ingrese la dirección de correo electrónico con la que se registró en la aplicación.

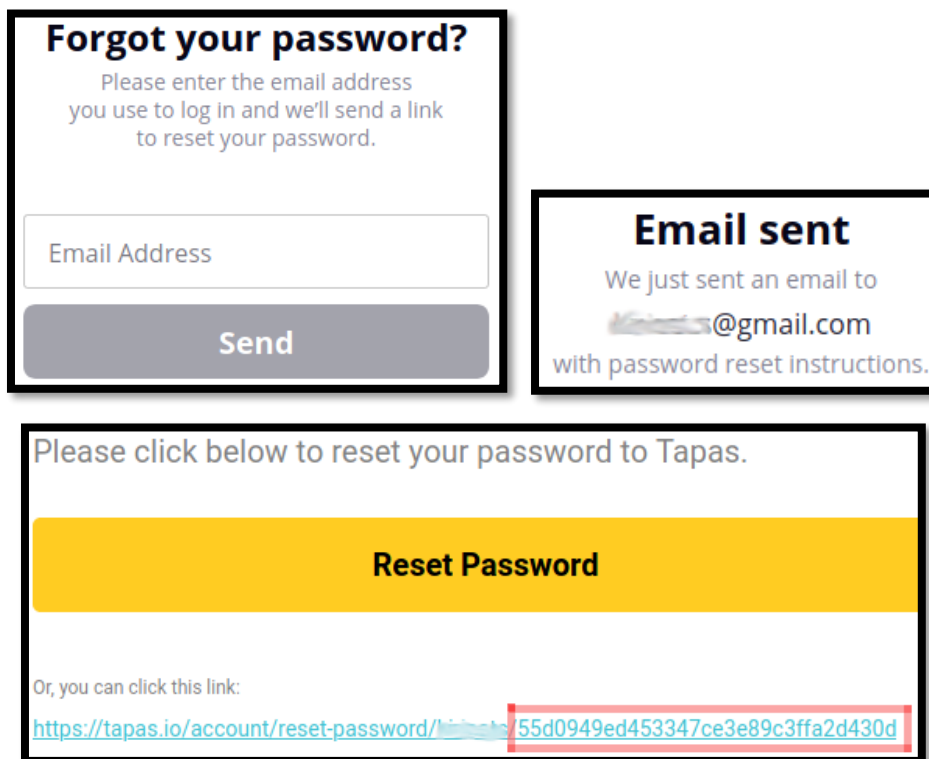


Figura 1. 22 Recuperación de contraseña mediante dirección de correo electrónico en Tapas.

Se observa nuevamente el enlace generado, que se envía junto al mail de recuperación de contraseña, la URL es similar a la utilizada para confirmar la dirección de

correo electrónico en el primer registro del usuario. Al final de esta URL se coloca como parámetros un ID autogenerated, único e impredecible.

El ID mencionado, en este punto de recuperación de un dato tan importante para la autenticación como la contraseña, es un indicador crítico a tener en cuenta en términos de seguridad. Este enlace conduce a una sección particular de la página en donde se le da al usuario la posibilidad de ingresar una nueva contraseña, con las validaciones pertinentes.

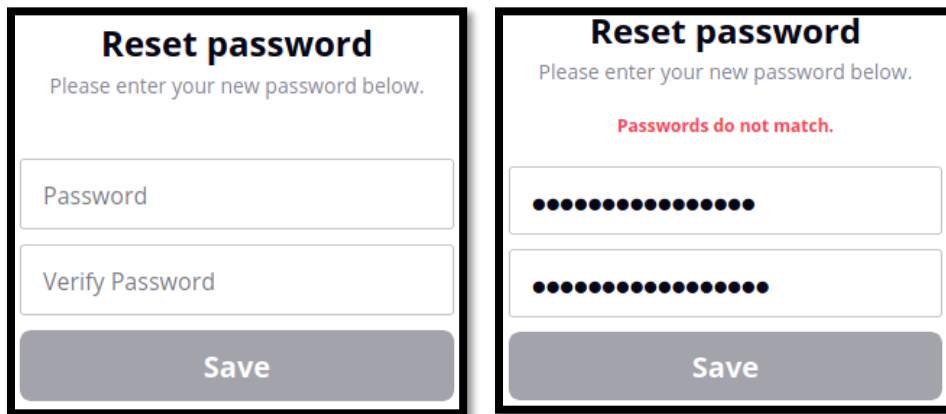


Figura 1. 23 Cambio de contraseña.

Cabe resaltar que, como norma general y principal tarea de seguridad a la hora de definir un módulo de usuarios junto a la forma de registro e ingreso de estos, ningún sistema contempla almacenar la contraseña de este en texto plano. Por el contrario, lo único que se almacena en la base de datos del sistema es un *hash* criptográfico de la contraseña. A partir de este hash no puede recuperarse la contraseña ya que es simplemente una firma única que la representa. Así, si el usuario la olvida, el único camino posible es cambiarla.

### Funcionalidad 3: Función de lectura

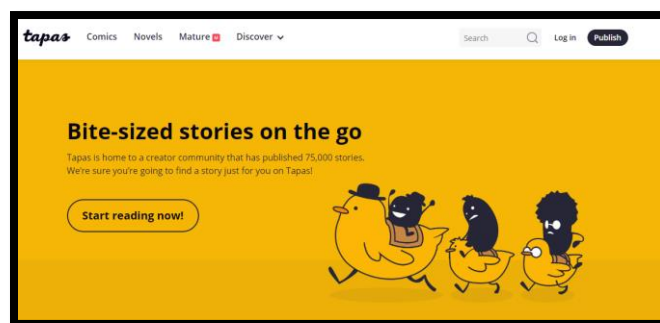


Figura 1. 24 Pantalla inicial de Tapas.

Se ofrece un amplio catálogo y, sobre todo, variado. A pesar de mantener categorías limitadas y fijas, también mantiene un criterio de filtro de acuerdo con tendencias y nuevas publicaciones.

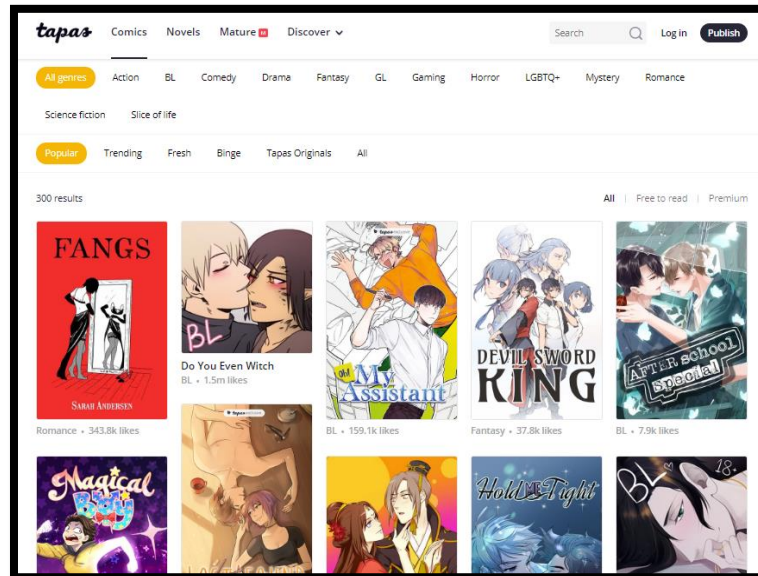


Figura 1. 25 Catálogo de Tapas.

Tapas también cuenta con una sección de obras destacadas por los mismos administradores de la aplicación.

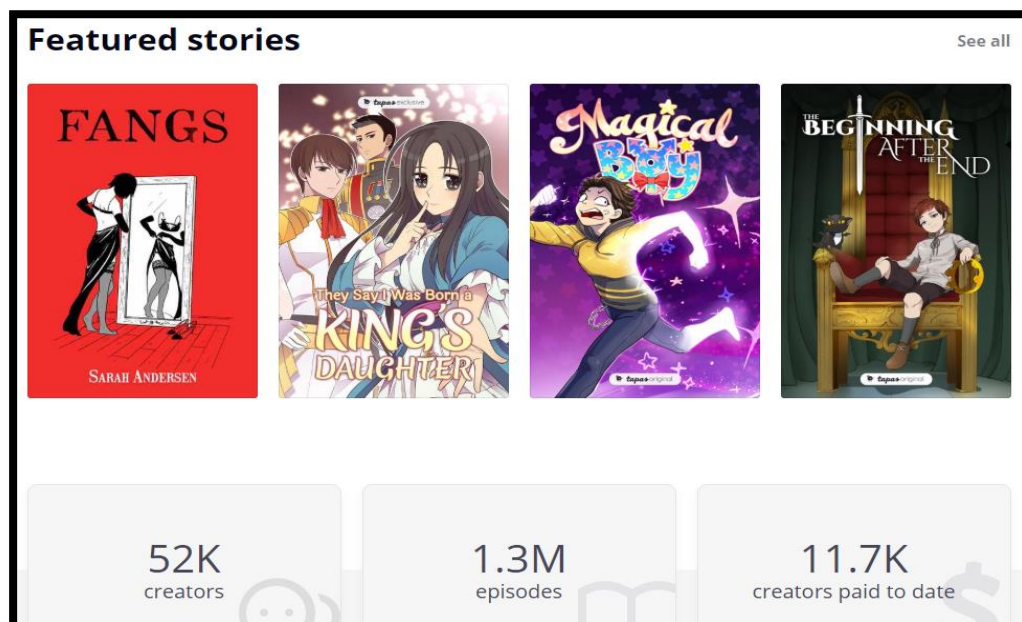


Figura 1. 26 Catálogo de Tapas

Los cómics se dividen en capítulos, la visualización del detalle es simple.

Al lector se le da la posibilidad de suscribirse a sus obras favoritas para así poder recibir notificaciones de las actualizaciones realizadas en la aplicación.

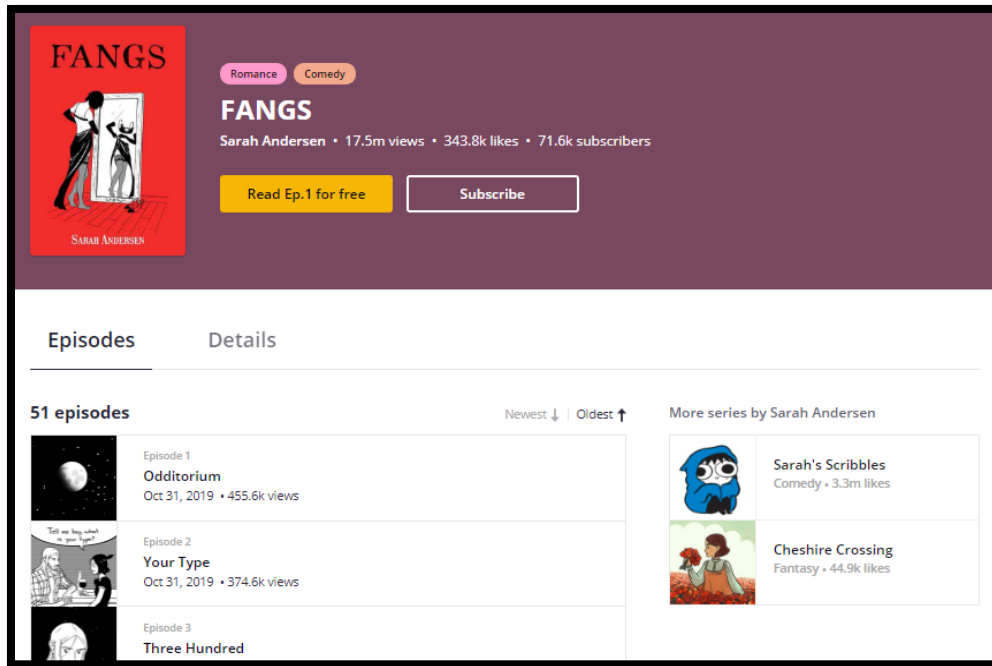


Figura 1. 27 Detalle de obra publicada.

El lector digital de cómics es sencillo y agradable a la vista. mientras se lee un cómic se puede valorar el mismo mediante *like* y ver comentarios de personas que lo han leído. Se pueden agregar comentarios a las obras.

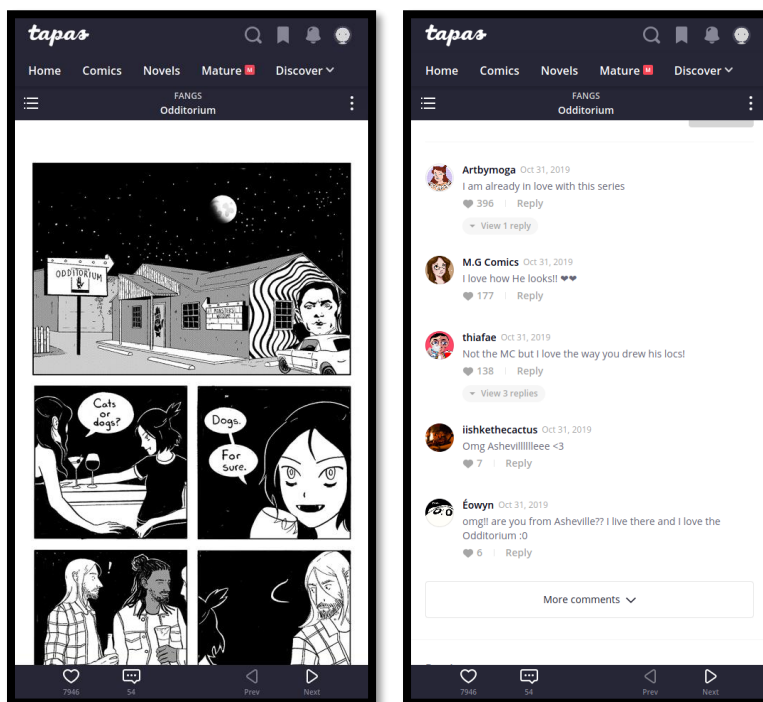


Figura 1. 28 Lector digital de Tapas.

**Funcionalidad 4: Función de publicación de obras**

Cualquier usuario puede publicar sus obras. El proceso es sencillo: crear el nuevo cómic con título, portada y URL personalizada. Esta última posibilidad es importante para que los creadores puedan distribuir sus obras independientemente de la plataforma, para que posibles lectores accedan directamente.

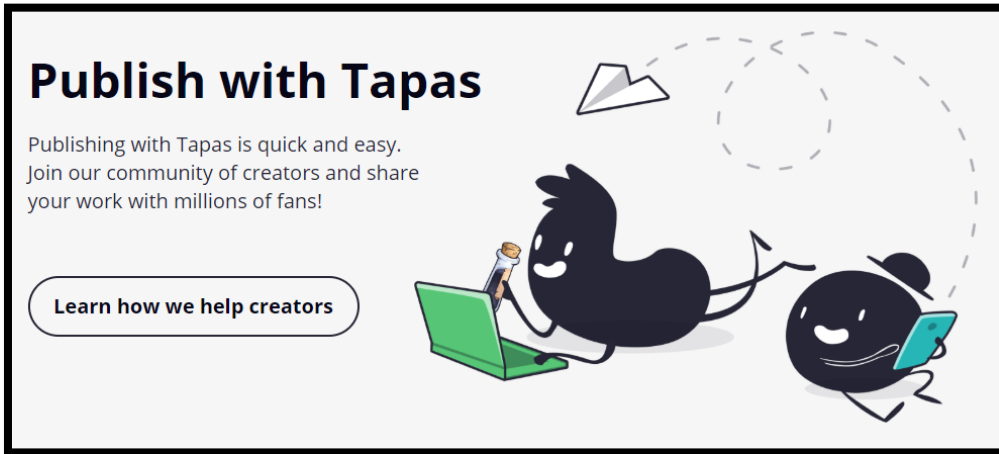


Figura 1. 29 Publicar en Tapas.

### New comic series

**Title**

**Url**

This is required.

**Thumbnail** 300 X 300px **Description**

Thumbnail must be safe for all ages.

**Series Banner** (Optional) 1280 X 480px

**Series Banner Link** (Optional) Up To 100 Characters

### Custom Ad Banner

(Optional) 280 X 180px  
This ad will appear below each of your episodes.

**Custom Ad Banner Link** (Optional) Up To 100 Characters

**Genre** (Select The Genre That Best Describes Your Series)

Action   
  BL   
  Comedy  
 Drama   
  Fantasy   
  Gaming  
 Horror   
  Mystery   
  Romance  
 Science fiction   
  Slice of life   
  GL  
 LGBTQ+

**Tags** (Optional)

**Website** (Optional) If you already publish your series elsewhere, you can add a link.

**Allow Embedding?** People will be able to embed this comic on other websites.

**Descending Order** New readers will start from your most recent episode.

Figura 1. 30 Publicar nueva obra en Tapas.

También se considera el agregado de etiquetas a una obra para su futuro filtrado en búsquedas del catálogo.

Otro punto importante es que el autor puede agregar publicidad externa en sus obras mediante un banner publicitario y un *enlace* personalizado.

**Funcionalidad 5: Función de donación a los autores**

La donación es el punto fuerte de Tapas. Maneja su propio sistema de créditos en moneda virtual llamada "Tinta", que los lectores pueden comprar para donar a sus creadores favoritos y así apoyarlos.

Los autores, posteriormente, pueden convertir estos créditos en dinero real. La plataforma les realiza pagos mediante servicios externos como PayPal.

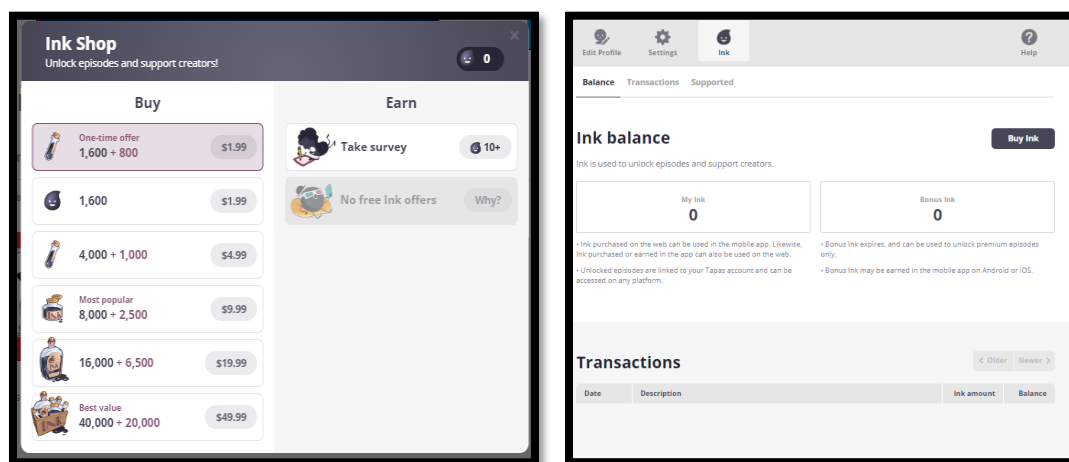


Figura 1. 31 Comprar tinta en Tapas para realizar donaciones.

**Funcionalidad 6: Marcadores**

Otro concepto interesante es el de marcadores o favoritos.

Un usuario lector posee una biblioteca personal en la que agrega sus cómics favoritos a partir de marcadores, también permite agregar aquellos que planea leer en un futuro.

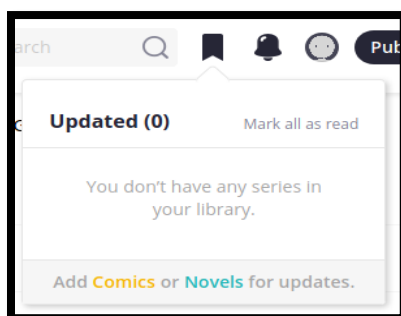




Figura 1. 32 Marcadores en Tapas.

### Funcionalidad 7: Notificaciones

Es destacable el módulo de notificaciones. Como se describió, cada usuario puede suscribirse a una obra en particular para recibir notificaciones, así también puede obtener notificaciones de actividades en su perfil o perfiles de otras personas.

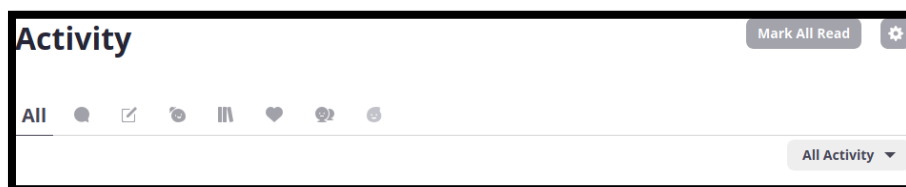


Figura 1. 33 Notificaciones en Tapas.

## 2.2. Modelo Lógico del Sistema Actual

Se puede observar que existen dos niveles de usuario: el lector y el autor. El autor es un usuario común, lector, que sube algún trabajo propio.

Un lector puede visualizar y valorar una historia (cómic o novela) a través de comentarios y "me gusta".

Una historia se compone, además de título, descripción, autor, visualizaciones, likes y suscriptores, de una portada y múltiples episodios.

Cada episodio puede estar conformado por una o más imágenes, las cuales se muestran de forma consecutiva, horizontalmente, en el lector digital.

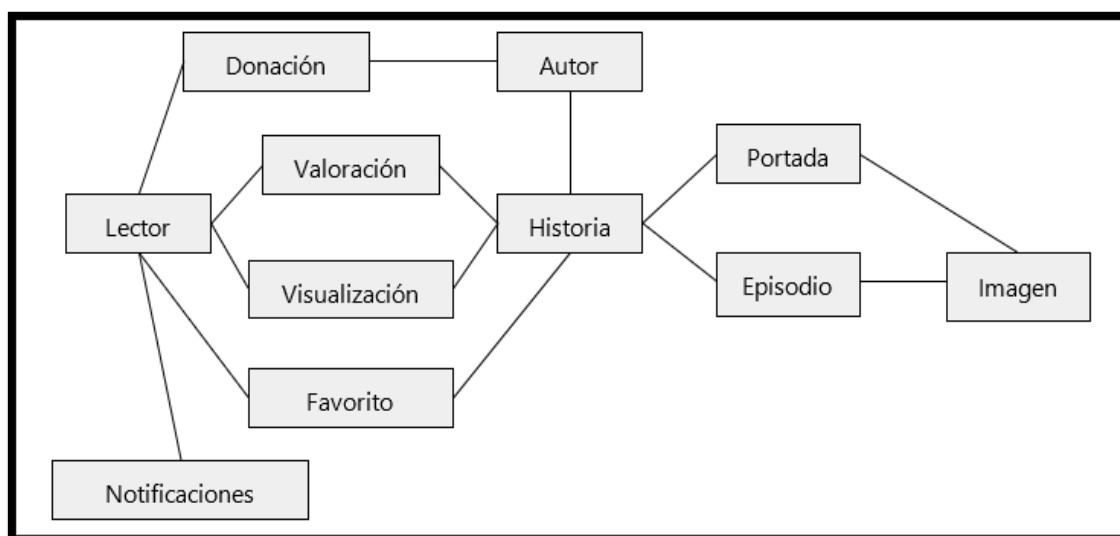


Diagrama 1. 5 Modelo de entidades de Tapas.

El **DIAGRAMA 1.5** es una aproximación genérica de la relación de entidades que se puede encontrar en la aplicación. El cual es suficiente para destacar la clase de usuarios que existe, y los datos básicos de un cómics o novela.

Entre los casos de uso de *Tapas*, a diferencia del relevamiento anterior, se destaca "Administración de perfil" ya que el usuario posee un perfil para publicar sus actividades y "Administrar marcadores" que le permiten agregar cómics a favoritos para llevar un seguimiento de estos.

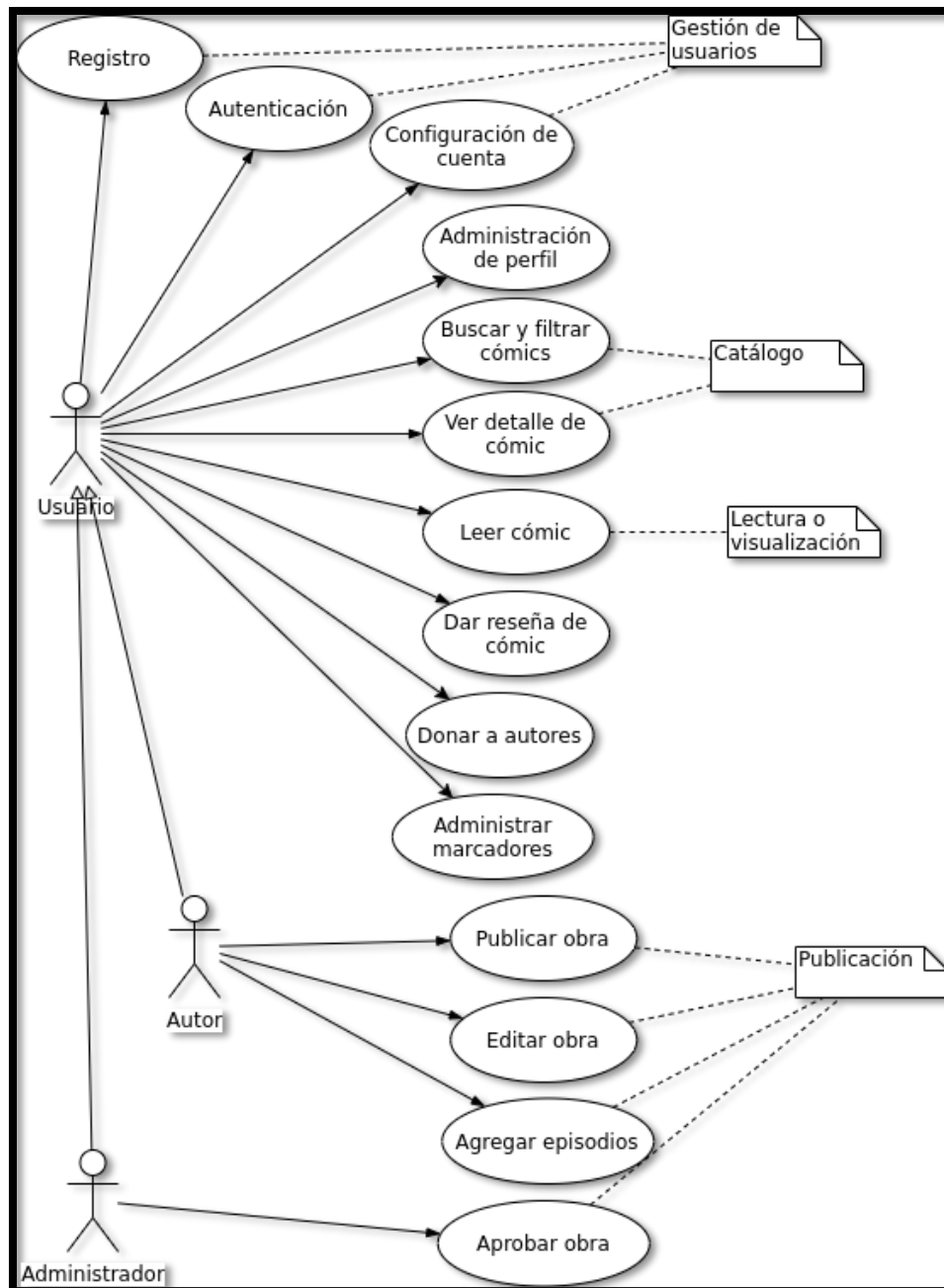


Diagrama 1.6 Diagrama de casos de uso de Tapas

## 2.3. Problemas y necesidades detectadas en las funciones relevadas en detalle y en su entorno organizacional

### **Funcionalidad 2: Autenticación y recuperación de contraseña**

#### Necesidad de notificar al usuario cuando su contraseña cambie:

Cuando un usuario cambia su contraseña, al ingresar una nueva luego de acceder mediante un enlace autogenerado y único para tal fin, no se le notifica acerca de esto mediante email. Puede representar un problema de seguridad en caso de que algún atacante cambie su contraseña y se apropie de su cuenta de usuario.

Un usuario cuya cuenta fue robada podría reaccionar tarde ante tal inconveniente al no ser notificado de la intrusión a su cuenta.

### **Funcionalidad 3: Función de lectura**

#### Problema de lectura secuencial:

Las imágenes de los episodios de cada cómic se cargan secuencialmente, siendo poco óptimo.

#### Necesidad de procesamiento de imágenes:

Las imágenes de los episodios deberían procesarse con algún servicio o librería especializada para generar imágenes de múltiples resoluciones, que puedan ser cargadas de acuerdo con la resolución del dispositivo visor.

## C. Webtoon

webtoons.com

### 1. Relevamiento General

#### 1.1. De la Organización

*Webtoon (WEBTOON - Cómic en línea, s/f)* es una plataforma abierta para todos los que tienen historias para compartir. Las historias que ofrece son contadas por creadores autónomos de todo el mundo. Pone a disposición su catálogo completo bajo diversas categorías (romance, comedia, fantasía, horror, otros).

Su objetivo principal es la distribución de contenido, no provee soluciones de pago: suscripciones Vips o donación a autores.

Cualquier usuario tiene acceso a todo el contenido gratis.

#### 1.2. Funciones Detectadas e Interfaces

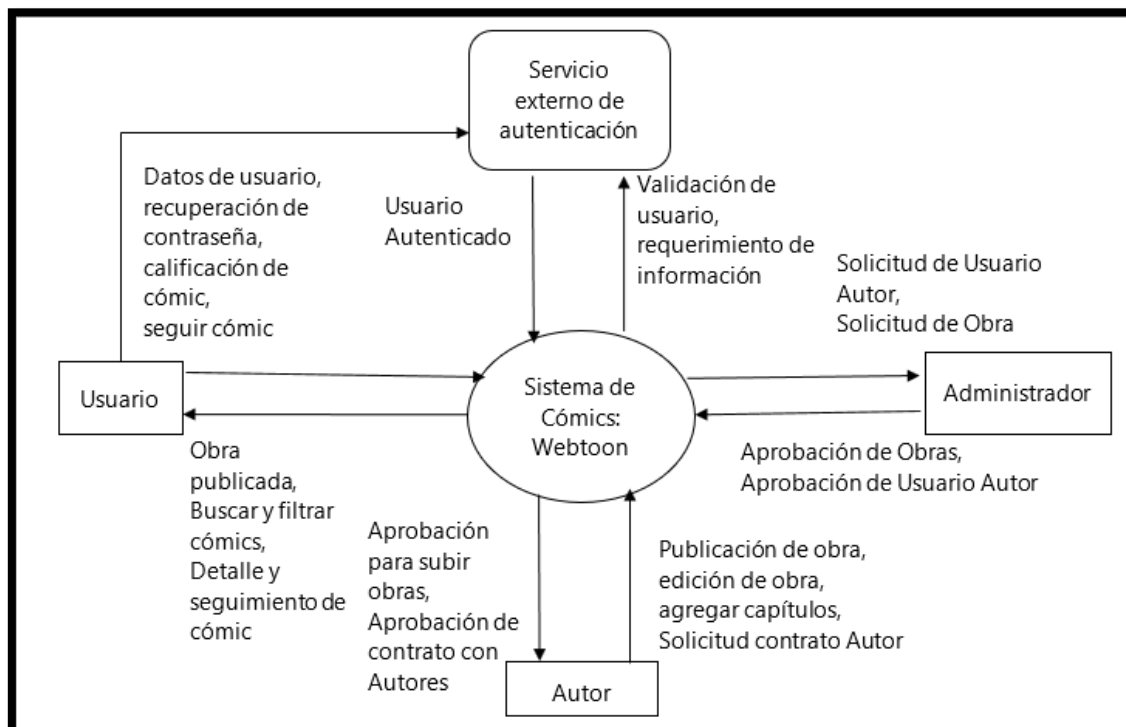


Diagrama 1. 7 Diagrama de contexto de interfaces detectadas en Webtoon.

Es notable la interfaz gráfica sencilla que presenta la aplicación de *Webtoon*.

Existe un único tipo de usuario: el lector. Los autores deben contactar directamente con la organización para publicar sus obras. No está claro el rédito, más allá de la distribución de contenido, que estos puedan recibir.

Se destacan las siguientes funcionalidades:

#### **Funcionalidad 1: Función de catálogo y detalle**

- ❖ Filtrado por día de la semana.
- ❖ Filtrado por categorías y géneros.
- ❖ Detalle del cómic: Calificación, visualizaciones e información del autor.

#### **Funcionalidad 2: Lectura de cómics**

- ❖ Lector digital sencillo.

#### **Funcionalidad 3: Seguimiento de series**

- ❖ Se puede seguir a una serie (cómic por episodio) para recibir notificaciones de actualizaciones.

#### **Funcionalidad 4: Calificación de cómics**

- ❖ Comentario acerca de obras.
- ❖ Puntaje de 1 a 10.

#### **Funcionalidad 5: Autenticación y registro de usuario**

- ❖ Autenticación mediante dirección de email y contraseña.
- ❖ Registro mediante dirección de email y contraseña, únicamente.
- ❖ Recuperación de contraseña.
- ❖ Inicio de sesión mediante servicios externos: Facebook, Twitter, Line, Google y Apple.

### **1.3. Tecnología de Información**

El front-end es completamente renderizado en servidor (back-end). No utiliza CDN externo para subir y servir archivos estáticos, como imágenes, que conforman el 99% del contenido ofrecido por el sitio.

El back-end utiliza NGINX como servidor de aplicaciones, PHP como lenguaje de desarrollo y PHP-FPM para interconectar el servidor y el lenguaje. Además, hace uso de Varnish-Cache (HTTP reverse proxy) para mejorar el tiempo de respuesta, ya que almacena temporalmente las solicitudes.

Utilizan LINE como método de comunicación directo con sus clientes.

## 2. Relevamiento Detallado y Análisis del Sistema

### 2.1. Detalle, explicación y documentación detallada de todas las funciones seleccionadas

#### Funcionalidad 1: Función de catálogo

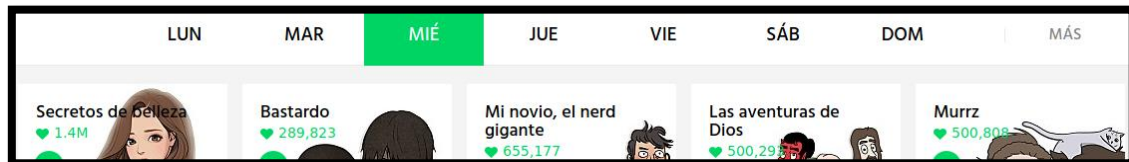


Figura 1. 34 Recomendación por día de la semana en Webtoon.

El catálogo presenta cómics recomendados por día de la semana.

El detalle de cada obra muestra la calificación de esta y los episodios. Los episodios son conforme a la forma en las que estas se componen.



Figura 1. 35 Detalle de una obra en Webtoon, por episodios.

Además de la información del cómic, se puede obtener un breve resumen con información del autor y otras publicaciones de su autoría.

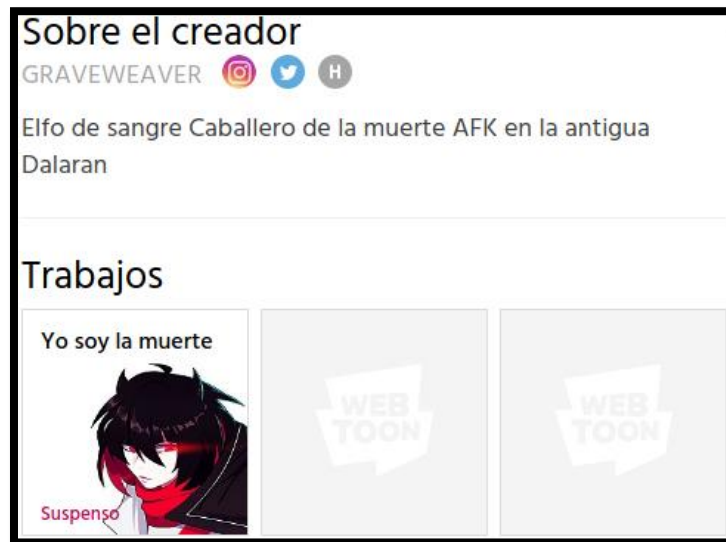


Figura 1. 36 Información del creador en Webtoon.

### Funcionalidad 2: Función de lectura

El lector no tiene más funcionalidad que permitir avanzar entre episodios del mismo cómic. Y las imágenes se muestran en secuencia, como una sola imagen que ocupa toda la pantalla.

No se adapta correctamente a los dispositivos móviles.

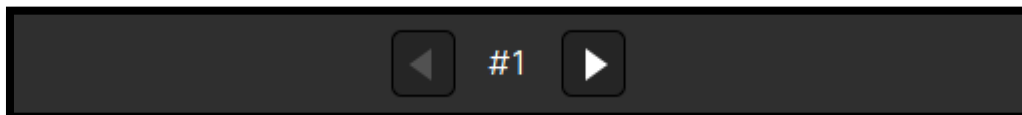


Figura 1. 37 Barra de navegación en lector de Webtoon.

### Funcionalidad 3: Seguimiento de series

El lector puede agregar a su lista de "Siguiendo" aquellos cómics que les resulte más interesantes, para así poder estar pendiente de sus avances o nuevas publicaciones de episodios.

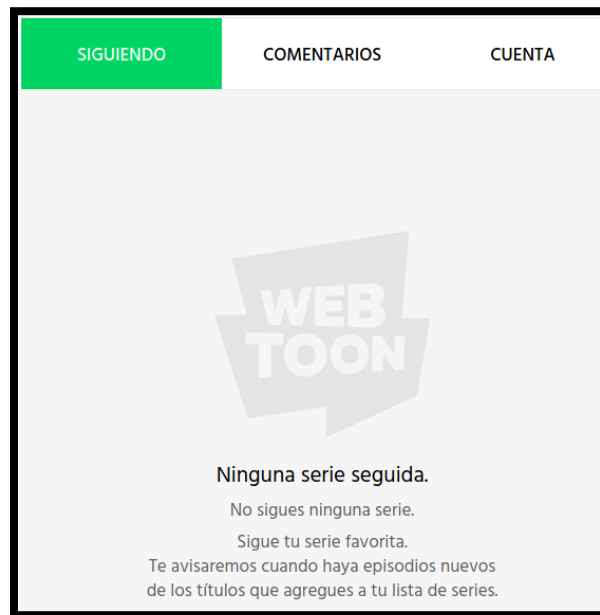


Figura 1. 38 Siguiendo en Webtoon.

#### Funcionalidad 4: Calificación de cómics

Se destaca esta funcionalidad porque provee un sistema de calificaciones que guían a los nuevos lectores a aventurarse en nuevas lecturas. Permite calificar cada cómic con notas del 1 al 10.

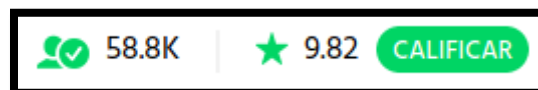


Figura 1. 39 Calificación de cómic en Webtoon.

#### Funcionalidad 5: Autenticación y registro de usuario

Para el registro inicial del usuario se deriva al mismo a un formulario donde, además de especificar la dirección de email y contraseña, puede especificar un apodo, usado como nombre de usuario (*nickname*). Al registrarse se aceptan los términos de uso y políticas de privacidad automáticamente.



Figura 1. 40 Registro de usuario en Webtoon.

La autenticación puede realizarse mediante servicios externos. Estas opciones utilizan tecnología OAuth, donde el proveedor de autenticación notifica a la aplicación que un usuario que se ha registrado y le brinda información de este.

Además, puede registrarse e identificarse mediante dirección de correo y contraseña. Sin embargo, esto puede resultar en el olvido de la contraseña por parte del usuario, por lo tanto, cuenta con una función para recuperar contraseñas.

Si el usuario lo requiere, puede recuperar la contraseña, también puede cambiarla si así lo desea, ingresando una nueva. Para esto ingresa únicamente su dirección de correo y un email generado automáticamente se le envía. Este posee un enlace particular, cuyos parámetros *authNo* y *hashValue* son fundamentales para detectar al usuario y verificar que realmente desea cambiar su contraseña. Ingresar una nueva contraseña para restablecerla.

Figura 1. 41 Recuperación de contraseña en Webtoon.

## 2.2. Modelo Lógico del Sistema Actual

El diagrama de entidades de la aplicación es sencillo. El único usuario es el lector, más allá de los posibles administradores encargados de cargar el contenido manualmente, esto último no está vinculado con la aplicación web. Los autores de los cómics no poseen perfil ni usuario dentro de la plataforma.

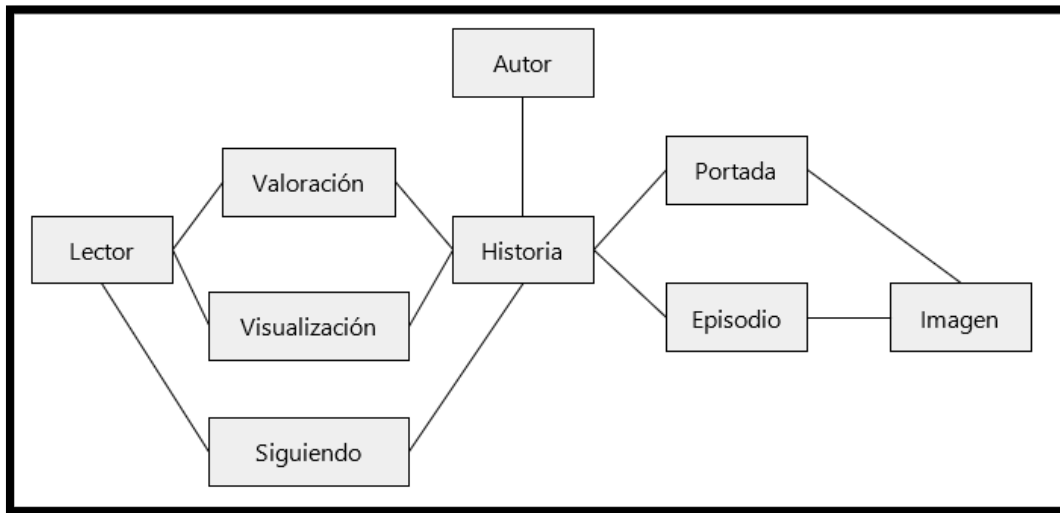


Diagrama 1. 8 Diagrama de entidades de Webtoon

## 2.3. Problemas y necesidades detectadas en las funciones relevadas en detalle y en su entorno organizacional

### Funcionalidad 2: Función de lectura

#### Necesidad de visor o lector digital:

Los episodios se presentan únicamente como secuencia de imágenes cargadas secuencialmente.

La única opción de navegabilidad es un botón para ir al siguiente episodio.

Presenta problema de adaptabilidad a dispositivos móviles y pantallas de resoluciones pequeñas.

## D. Crunchyroll

crunchyroll.com

### 1. Relevamiento General

#### 1.1. De la Organización

*Crunchyroll (Read Popular Manga Online - Crunchyroll, s/f)* es una compañía de Warner Media (como parte de Otter Media) y una marca de distribución de anime muy reconocida a nivel mundial. Es uno de los distribuidores más grandes en el momento de Occidente y hace pocos años arribó a Latinoamérica.

Se encargan de subtítular y traducir contenido conocido en Japón y Sur Corea a Occidente, obteniendo las licencias necesarias por región. También hace que los miembros compartan opiniones de sus animes y mangas favoritos.

Además de distribución de videos (que actúa como “el Netflix del anime”) tienen una aplicación en inglés móvil para los mangas que tienen licencias en occidente. Esta aplicación es la que vamos a tomar como modelo, para relevar ya que los mangas son como se conocen los comics en Oriente, con sus pequeñas diferencias.

El modelo de negocios de *Crunchyroll* abarca, tanto anime (videos) como manga (cómic). Para los animes tiene tanto acceso por suscripción como algunos gratuitos por publicidad, en tanto que para los cómics (el cual es de nuestro interés), sólo se tiene acceso gratuito al primer número de cada obra, teniendo acceso a una prueba gratuita de 14 días. Funciona para todo su contenido (visual y gráfico) mediante suscripción mensual, trimestral o anual.



Figura 1. 42 Logo de Crunchyroll.

## 1.2. Funciones Detectadas e Interfaces

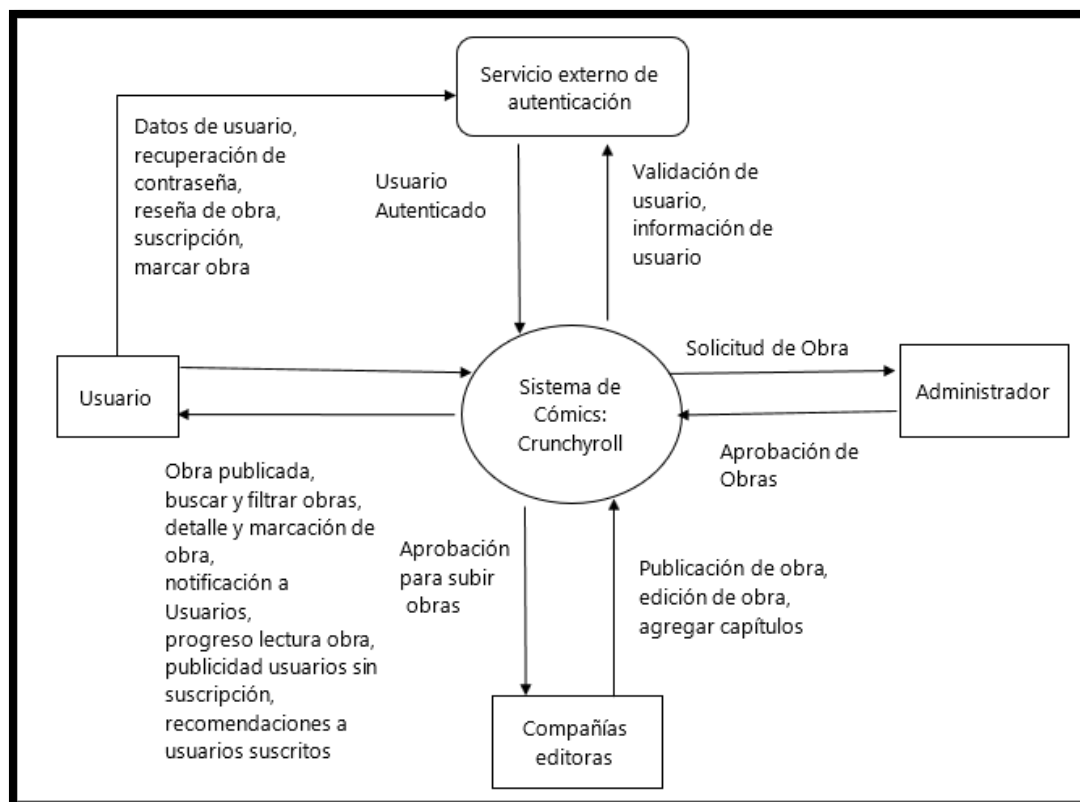


Diagrama 1.9 Diagrama de contexto de interfaces detectadas en Crunchyroll.

Las funciones observadas en *Crunchyroll*, son las que se detallan a continuación:

### Funcionalidad 1: Función de catálogo

- ❖ Se tiene una vista global de todo lo que ofrece la plataforma: Recomendados, lanzamientos recientes, y los más populares resaltados.
- ❖ Puede ordenarse el catálogo en lista por filas o cuadrícula con una previsualización de la portada de la obra.
- ❖ Se usa como medio para publicitar contenido para los usuarios no premium, y recomendaciones especiales para los que tienen cuenta premium.

### Funcionalidad 2: Función de búsqueda y filtros

- ❖ Pueden buscarse a través de palabras claves diferentes mangas que se relacionan en título o contenido con lo escrito por el usuario.
- ❖ Pueden utilizarse algunos filtros como ver los populares, los que tienen alguna realización en conjunto con alguna editorial, de lanzamiento simultaneo en su

país de origen, de lanzamiento reciente o por géneros: acción/aventura, comedia, drama, fantasía, romance, ciencia ficción, misterio, histórico, horror y deportes.

### **Funcionalidad 3: Función de lectura**

- ❖ Se pueden acceder a la información general del manga.
- ❖ Cada manga está dividido en capítulos teniendo el primero gratuito para todos los usuarios y la mayoría de los subsiguientes para usuarios premium.
- ❖ Puede visualizarse el progreso que se ha realizado en cada capítulo.
- ❖ Dispone de dos modos de lectura:
  - Página completa y deslizar el dedo (swipe) para ir a la siguiente o anterior.
  - Usar guía: Remarca la primera viñeta y oscurece el resto para resaltar con un acercamiento, y al presionar a los costados se va a la siguiente o anterior.
- ❖ Se tiene una opción para añadirlo a la lista de marcadores.

### **Funcionalidad 4: Función de favoritos y marcadores**

- ❖ El usuario puede guardar información de sus mangas en una lista de favoritos.
- ❖ Pueden tenerse una lista de marcadores para saber dónde se iba leyendo para continuar desde ese punto.
- ❖ La lista de favoritos es útil para recibir notificaciones de nuevas entregas de sus obras favoritas.

### **Funcionalidad 5: Configuraciones**

- ❖ Opciones para registrarse a la aplicación.
- ❖ Manejo de las notificaciones deseadas.
- ❖ Información sobre los términos de privacidad.
- ❖ Detalles de la cuenta del usuario.

### **Funcionalidad 6: Función de suscripción**

- ❖ En el caso de no ser un usuario con "Cuenta Premium" y seleccionar algún capítulo u obra que requiere que sea de ese tipo de usuario, se envía a una página web de *Crunchyroll* para seleccionar un tipo de suscripción: Premium normal o Premium Plus (la única diferencia con la anterior, son accesos a material físico para coleccionar que solo está disponible en Norteamérica). Una vez que se haya

cochado el tipo de selección, se puede acceder a todo el contenido de la plataforma, tanto de la aplicación de mangas como la de videos.

### **1.3. Tecnología de Información**

Es una aplicación nativa, desarrollada en Java en Android y Swift en iOS. Se observa que utilizan un servidor específico para alojar las imágenes y las obras, y en otro servidor toda la información de los usuarios y sus datos; estos son de terceros, pero no se aclara en lado alguno quién es el proveedor.

Algo interesante que hacen para evitar la copia de los mangas, es bloqueando cualquier posibilidad de realizar capturas de pantalla nativas desde el dispositivo móvil, esto se aclara en los permisos solicitados al instalar la aplicación al usuario.

Actualmente está mostrándose en la plataforma web la información de las obras mediante React (librería de JavaScript) junto a HTML y CSS.

## 2. Relevamiento Detallado y Análisis del Sistema

### 2.1. Detalle, explicación y documentación detallada de todas las funciones seleccionadas

#### Funcionalidad 1: Función de catálogo

El usuario luego de realizar un Loguin, utilizando email y contraseña o cuenta de Facebook o Google, se ingresa a la pantalla principal donde se remarca en la parte superior las obras recomendadas o promocionadas por *Crunchyroll*. Debajo de la mencionada cartelera, se muestran mangas en orden de popularidad del momento pudiendo visualizarlos en forma de lista o grilla cuadriculada.

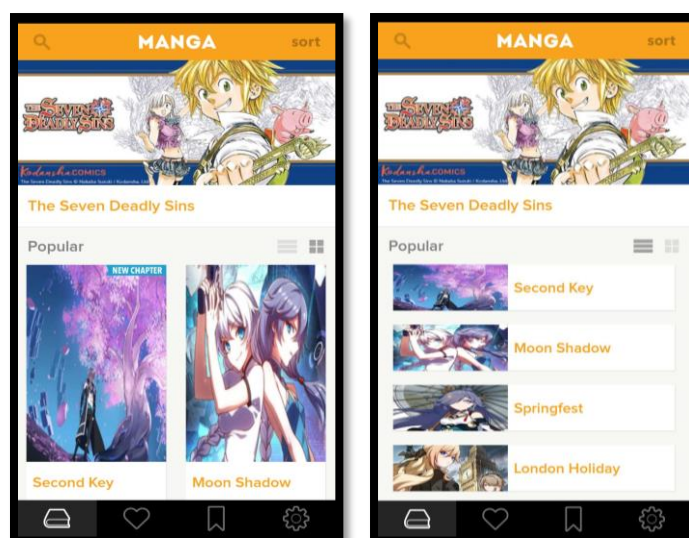


Figura 1. 43 Inicio de Crunchyroll Manga.

Izquierda: Vista en forma cuadriculada. Derecha: Vista en forma lista.

#### Funcionalidad 2: Función de búsqueda y filtros

Seleccionando el ícono de lupa, ubicado arriba a la izquierda de la pantalla, se puede ingresar un filtro por búsqueda que muestra el catálogo que tenga alguna relación en su título o contenido con lo que ingresa el usuario.

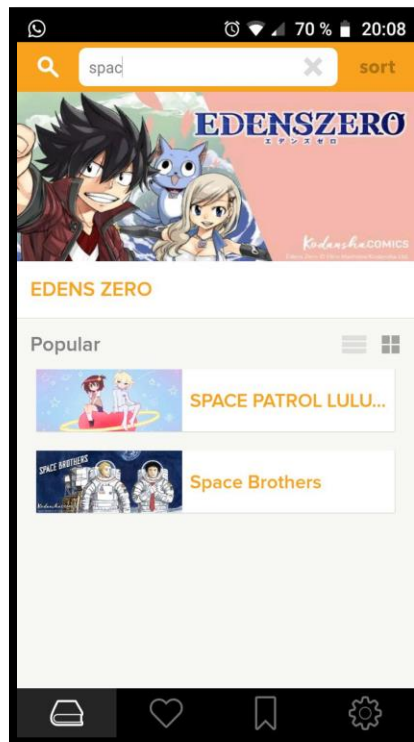


Figura 1. 44 Filtros de búsqueda por coincidencia de texto de Crunchyroll.

También podemos hacer click, en la página de inicio de la aplicación, sobre la palabra "sort" (ordenar) ubicado en la parte superior derecha de la pantalla. Al ingresar se nos muestra diversas opciones para filtrar el catálogo que tiene disponible la aplicación, las cuales son las siguientes:

- ❖ Popular
  - Clasificación por la popularidad en ese momento de lo que más se selecciona para leer.
- ❖ Joint Promotion
  - Catálogos de obras promocionadas y realizadas en conjunto con los editores.
- ❖ Simul Pub
  - Catálogos de obras que se publican en simultáneo con Crunchyroll junto a los editores de la misma.
- ❖ Recently Launched
  - Catálogo de obras actualizados recientemente.
- ❖ Genres
  - Catálogo dividido por diferentes géneros en los cuales se encuentran: Acción/Aventura, Comedia, Drama, Fantasía, Romance, Ciencia Ficción, Misterio, Histórico, Horror y Deportes.



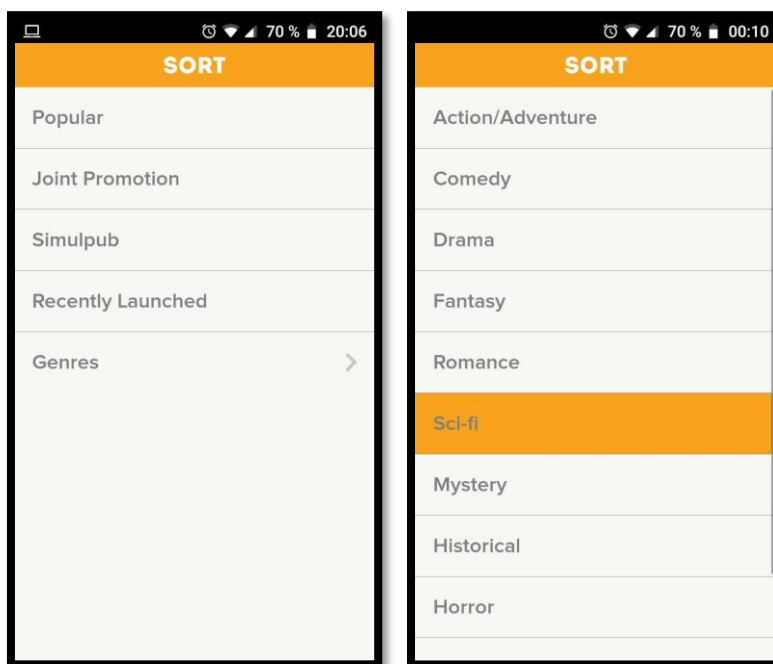


Figura 1. 45 Filtros de diferentes formas de mostrar el catálogo en Crunchyroll.

### Funcionalidad 3: Función de lectura

La función principal que tiene el lector de mangas de *Crunchyroll* es la de lectura. Al seleccionar un título, este muestra una portada con una breve descripción y su título en el parte superior seguido de un botón para “ver más” que al seleccionarlo expande el panel y se muestra: descripción completa, autor, artista, editor, año en que fue por primera vez publicado, detalles del Copyright de la obra. También se visualiza un botón en la esquina superior derecha de la pantalla en forma de corazón que es para agregar a nuestra lista de favoritos.

Debajo de lo anterior señalado, se muestran los capítulos (o tomos) ordenados desde el más reciente hasta el más antiguo. Aquellos capítulos que tienen una corona son exclusivos para quienes son usuarios premium. Se puede apreciar una barra de progreso debajo de cada número para mostrar cuánto el usuario ha leído del mismo.

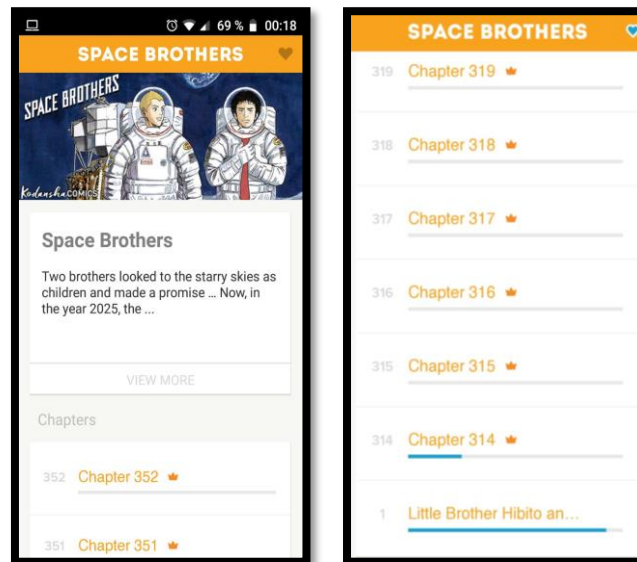


Figura 1. 46 Progreso en lecturas en Crunchyroll.

Una vez que seleccionamos un capítulo del manga a leer, éste se muestra desde la primera página si no lo habíamos accedido antes, o se muestra desde la página que habíamos leído con anterioridad.

Se tiene la opción para que el usuario pueda dejarlo en su lista de marcadores, y la opción "Empezar Guía" (que luego cambia a "Salir de la Guía") que muestra resaltado la primera viñeta y hace un acercamiento (oscureciendo todo lo que sea alrededor), para ir a la siguiente viñeta, simplemente se presiona en el lado izquierdo para continuar y para volver en el lado derecho.

Sin la guía, se hace un swipe (deslizar el dedo) de izquierda a derecha para ir a la siguiente página. En la parte superior se muestra el capítulo y página en la cual el usuario está ubicado, y en la parte inferior se muestra una barra de progreso con el que también el usuario puede moverse entre páginas.

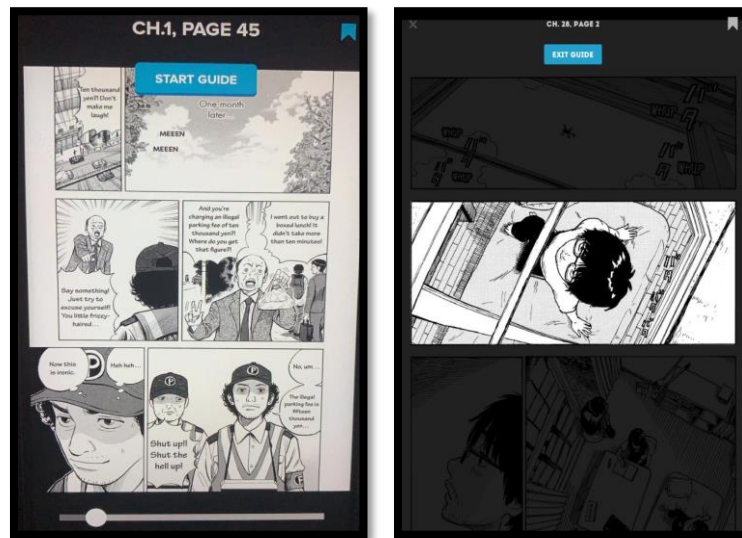


Figura 1. 47 La forma de leerse los mangas en Crunchyroll: Completa y siguiendo una guía que resalta la viñeta, una por una.

#### Funcionalidad 4: Función de favoritos y marcadores

Al seleccionar el botón con el ícono de corazón aparece nuestra una "Lista de favoritos" que el usuario haya marcado previamente. Si seleccionamos el botón del icono de corazón azul, se remueven de la lista, la cual se refresca una vez que el usuario sale de la pantalla.

Si el usuario selecciona el botón que se encuentra a la derecha del ícono de corazón, se muestran todos los marcadores que el usuario (bookmarks) que ha ido marcando, para así poder retomar desde ese punto de lectura y se borra ese marcador de la lista.

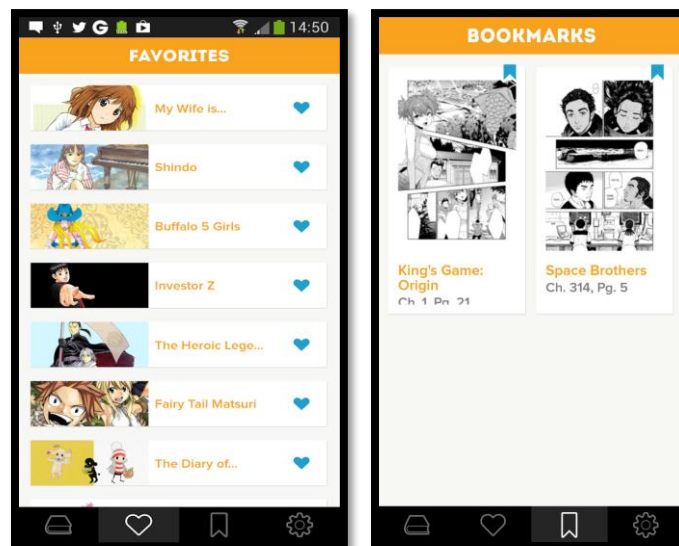
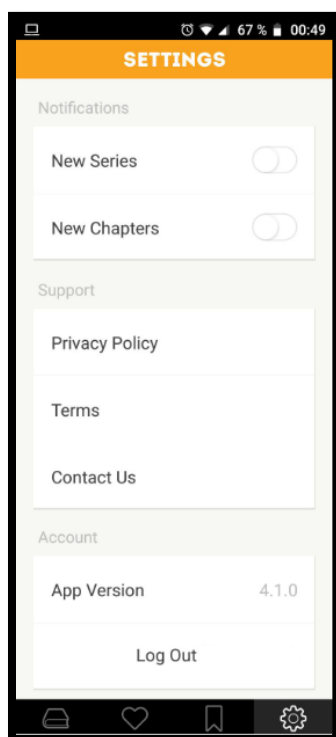


Figura 1. 48 (Izquierda) Lista de páginas marcadas. (Derecha) Lista de favoritos.

## Funcionalidad 5: Configuraciones

La última opción del menú inferior son las configuraciones (el botón con el ícono de engranaje) el cual lleva al usuario a una pantalla con algunas opciones que puede colocar e información extra. La primera son las notificaciones que quiere el usuario recibir en el móvil, que pueden ser sobre nuevas series/obras en la plataforma o nuevos capítulos de alguno de los mangas añadidos a su "Lista de favoritos".

Luego se muestran las opciones de soporte con los términos de privacidad de la aplicación y una forma de contactar al usuario (vía email en este caso). Luego se visualiza información sobre la versión de la aplicación y finalmente la opción para salir de la cuenta, pudiendo luego el usuario loguearse con alguna otra cuenta, con configuraciones y permisos diferentes.



*Figura 1. 49 Todas las opciones e información que aparece en la pantalla Configuraciones.*

## 2.2. Modelo Lógico del Sistema Actual

El cómic (en este caso manga) se compone de un título, descripción, editor, género que pertenece, portada, página y episodios, además de otros atributos que componen a la obra en sí que están dentro de un catálogo que accede el usuario con su "Lista de favoritos", y marcadores. La portada y la página tienen una imagen asociada, mediante ella el usuario puede acceder a la plataforma.

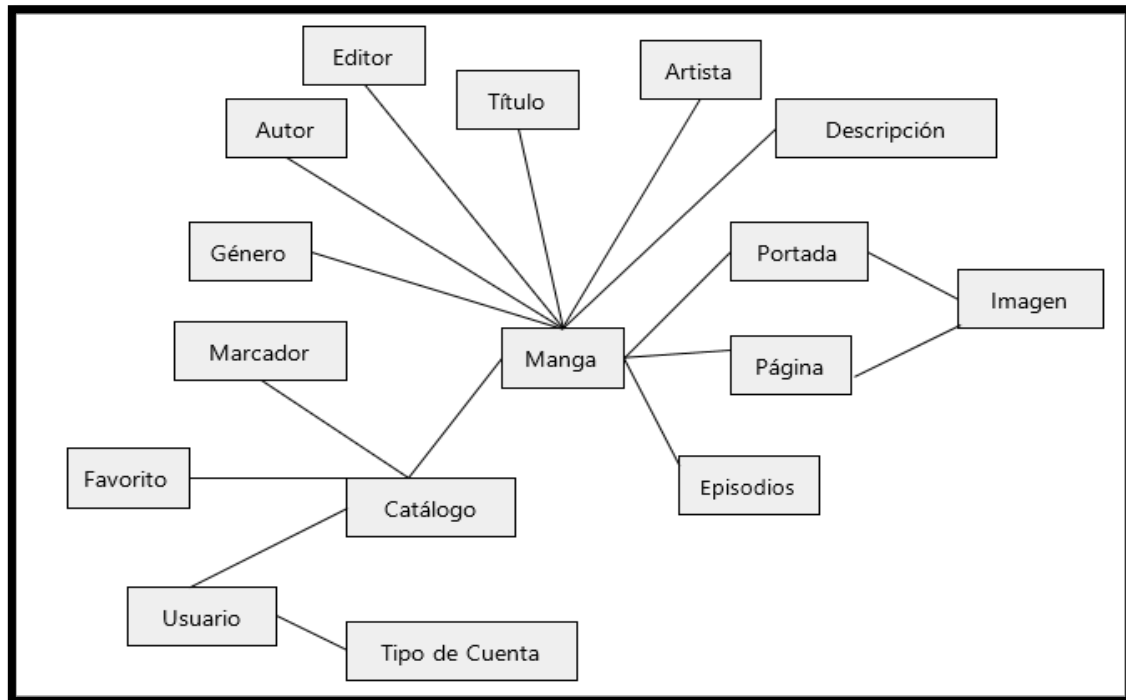


Diagrama 1. 10 Modelo de entidades de Crunchyroll.

El **DIAGRAMA 1. 10** es una aproximación a lo que se puede encontrar en la plataforma *Crunchyroll*. Lo importante a destacar son las entidades relacionadas con el manga/comic, y lo que el usuario requiere para acceder al contenido de la misma.

Las principales funcionalidades visualizadas como caso de uso pueden verse en el siguiente diagrama. Se han agregado notas para indicar la posible agrupación de las mismas, como si de módulos de la aplicación se tratase:

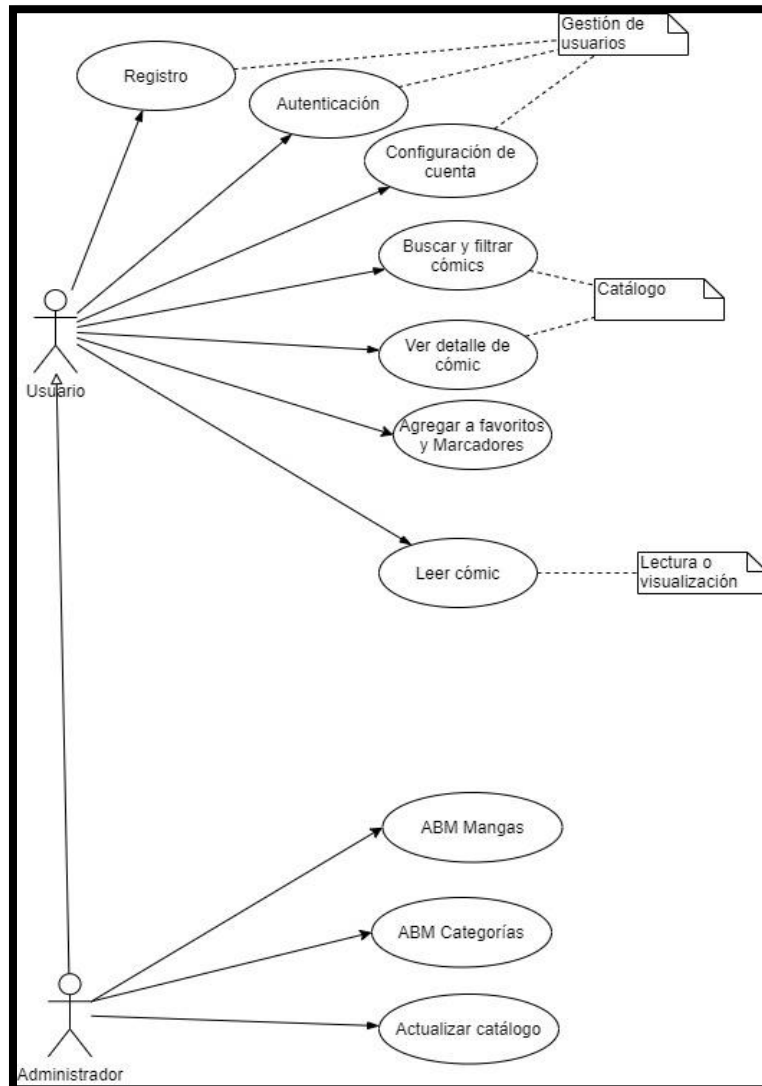


Diagrama 1. 11 Diagrama de Casos de Uso general de Crunchyroll.

## 2.3. Problemas y necesidades detectadas en las funciones relevadas en detalle y en su entorno organizacional

### Funcionalidad 1: Función de catálogo

#### Problema continuidad con algunas obras:

- ❖ Se observa que hay muchos mangas que comienzan con el primer capítulo, y luego por temas de licencias saltan, por ejemplo, al capítulo 300. Esto, basado en comentarios de algunos usuarios, suele hacer que abandonen la plataforma.

#### Necesidad de avisos sobre algunas obras:

- ❖ Señalar si algunos mangas no tienen los números completos, ya que al acceder y pagar el contenido premium puede resultar frustrante cuando un lector no encuentra algo de su interés.

### **Funcionalidad 2: Función de búsqueda y filtros**

#### Problema de comprensión en filtros:

- ❖ Algunos de los filtros no son fáciles de entender. Sería bueno que tenga un botón de "más información" para poder entender de qué trata. Por ejemplo, con algunas que tuvimos que preguntar y buscar han sido: Joint Promotion y Simul Pub.

#### Necesidad de opciones búsqueda:

- ❖ Faltan categorías orientadas a público objetivo como rangos de edad (mayores, menores, adolescentes, etcétera) y regiones (surcoreana, china, japonesa, etcétera).
- ❖ Un sistema de etiquetas para filtrar por tipo de contenido dentro del catálogo.
- ❖ No hay forma de buscar en la aplicación por autor ni editorial.
- ❖ No hay un filtro o recomendaciones para nuevos lectores de algún género en particular.

### **Funcionalidad 3: Función de lectura**

#### Problema para nuevos lectores:

- ❖ Para el usuario que no suele leer mangas/comics no existe guía para entender cómo usar la aplicación, ni las opciones existentes en la misma.

#### Necesidad de opciones de idiomas:

- ❖ A pesar de estar ubicados en Latinoamérica, no hay contenido en otro idioma que no sea en inglés.

#### Necesidad recomendaciones de otros usuarios:

- ❖ No existen comentarios ni un porcentaje, o puntaje sobre los mangas que hay en la plataforma de parte de los usuarios.
- ❖ Basándose en otras plataformas como de juegos, se podría ver una recomendación o puntaje según cómo empieza la obra y cómo continúa.

### **Funcionalidad 4: Función de favoritos y marcadores**

#### Problema al guardar favoritos:

- ❖ Actualmente algunos mangas al ser guardados en favoritos desaparecen de la "Lista de favoritos" al reiniciar la aplicación.

## Funcionalidad 5: Configuraciones

### Necesidad de mayores opciones de configuración:

- ❖ No existen muchas opciones que pueda el usuario controlar desde la aplicación. Para ello debe dirigirse a la página web de *Crunchyroll*, y configurar su categoría de usuario, como por ejemplo la información de su pago.
- ❖ Sería muy bueno para el lector tener guardado preferencias de lo que suele querer leer para que el sistema le recomiende contenido basado en sus gustos.



## E. Rootinks

rootinks.com

### 1. Relevamiento General

#### 1.1. De la Organización

*Rootinks* (*ROOTINKS*, s/f) es una aplicación de cómics de autores independientes argentinos, que brinda un servicio de entretenimiento, permite conocer las obras de los autores de cómics. El contenido está conformado por: comics prestigiosos, cómics independientes, y novelas gráficas. Es una organización perteneciente a "Comics by TMob", con domicilio en Buenos Aires, Argentina.

Los contenidos (imágenes, textos, actividades y otros) se ajustan al formato digital, respetando los estilos típicos del cómic, pero también desarrollados para que el mismo pueda ser leído y consumido desde cualquier dispositivo móvil. Además, brinda la posibilidad de disfrutar los cómics sin estar conectados a internet, con sólo descargar los cómics para leer off-line.

*Rootinks* es una aplicación móvil disponible para App Store y Google Play. Está destinada al usuario lector para que pueda visualizar sus cómics preferidos, y por categorías pudiendo armar su biblioteca de cómics preferidos en la opción "Mi Rootinks".

El lector también puede ver los cómics destacados y sus cómics favoritos en "Favoritos". Además, brinda la posibilidad de disfrutar los cómics sin estar conectados a internet con sólo descargar los cómics para leer offline, por ejemplo, durante un viaje.



Figura 1. 50 Pantalla de bienvenida a la página web Rootinks.

## 1.2. Funciones Detectadas e Interfaces

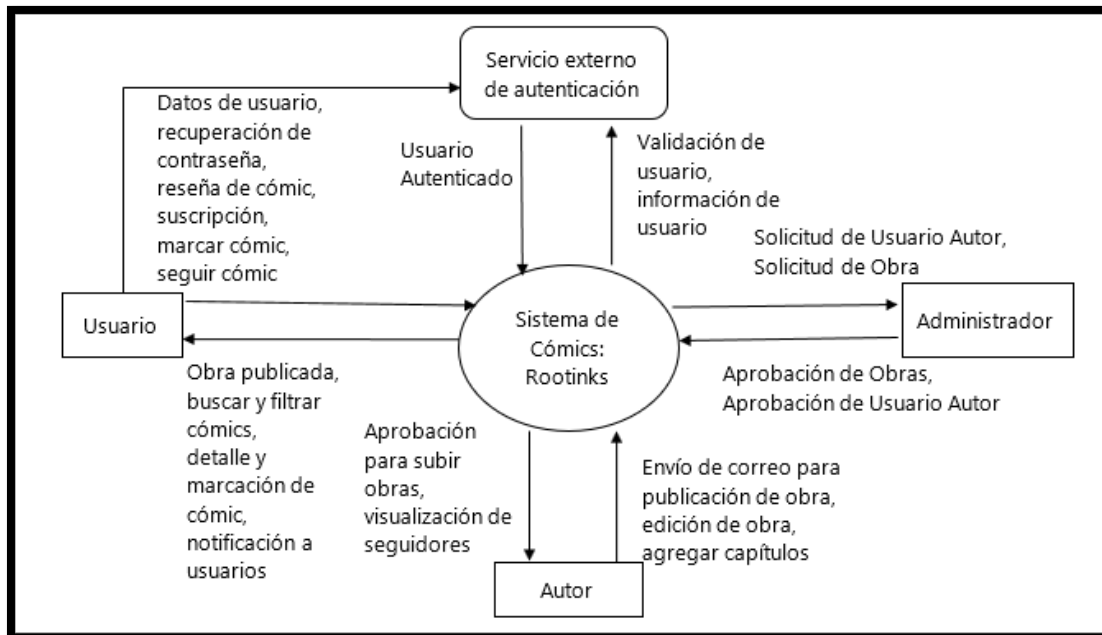


Diagrama 1. 12 Diagrama de contexto de interfaces detectadas en Rootinks.

En la aplicación de *Rootinks* se pueden detectar las siguientes funcionalidades:

### Funcionalidad 1: Función de login

- ❖ El lector tiene dos formas de acceder al contenido de la aplicación:
  - De forma gratuita (limitado a 7 días).
  - Por medio de una suscripción.

### Funcionalidad 2: Función de suscripción y usuario

El servicio de suscripción Premium brinda al usuario lector acceso ilimitado a todos los contenidos y le habilita todas las funcionalidades de la aplicación. Dicha suscripción puede ser cancelada en cualquier momento, pero se mantiene activa hasta el fin del mes ya cobrado. Ofrece un período gratuito de 7 días, luego de pasado dicho período, se pagará el plan que elija el usuario (mensual o anual).

### Funcionalidad 3: Función de catálogo

- ❖ Presentación de obras filtradas por categorías.
- ❖ Cada obra tiene una sinopsis, la cantidad de páginas que la componen, información sobre el autor, las visitas, los likes, la cantidad de usuarios interesados en la misma, y en cuáles otros comics dichos usuarios han visitado.
- ❖ Destacados: Permite visualizar los cómics recomendados, con buenas críticas, autores destacados, y cómics gratuitos para cualquier usuario.

#### **Funcionalidad 4: Función de favoritos y marcadores**

Esta funcionalidad le posibilita al lector tener marcadores de cómics, pudiendo armar su biblioteca de cómics preferidos en la opción "Mi Rootinks".

#### **Funcionalidad 5: Función de lectura**

Está destinada al usuario lector para que puedan visualizar sus cómics preferidos por página.

#### **Funcionalidad 6: Comunicación con el usuario**

La comunicación con el usuario se realiza mediante el correo electrónico, para brindar atención personalizada tanto a los lectores, como a los autores de los cómics. Además, la aplicación permite el ingreso de las "opiniones de usuarios", con la finalidad de tener un feedback con los lectores, y así poder realizar mejoras en la aplicación.

#### **Funcionalidad 7: Función de publicación de obra**

Si un autor de cómics desea difundir su material en Rootinks, debe enviar un email a la dirección de correo electrónico marcado en la página web oficial de la aplicación.

### **1.3. Tecnología de Información**

El front-end está elaborado utilizando el framework visual EmberJs, un framework de JavaScript utilizado el lenguaje Typescript. Debido a que es una aplicación nativa no es posible acceder a más información detallada. Sin embargo, por lo visto se puede observar que *Rootinks* utiliza algún otro framework para uso nativo en dispositivos móviles como Ionic o Cordova (acceso a contactos, llamadas, cámara, etcétera).

Para el uso de back-end, se observa el uso de un servidor de terceros para alojar las imágenes y un host para su sitio web.

## 2. Relevamiento Detallado y Análisis del Sistema

### 2.1. Detalle, explicación y documentación detallada de todas las funciones seleccionadas

#### Funcionalidad 1: Función de login

Ingreso de usuario mediante email y contraseña o por OAuth, utilizando Facebook.

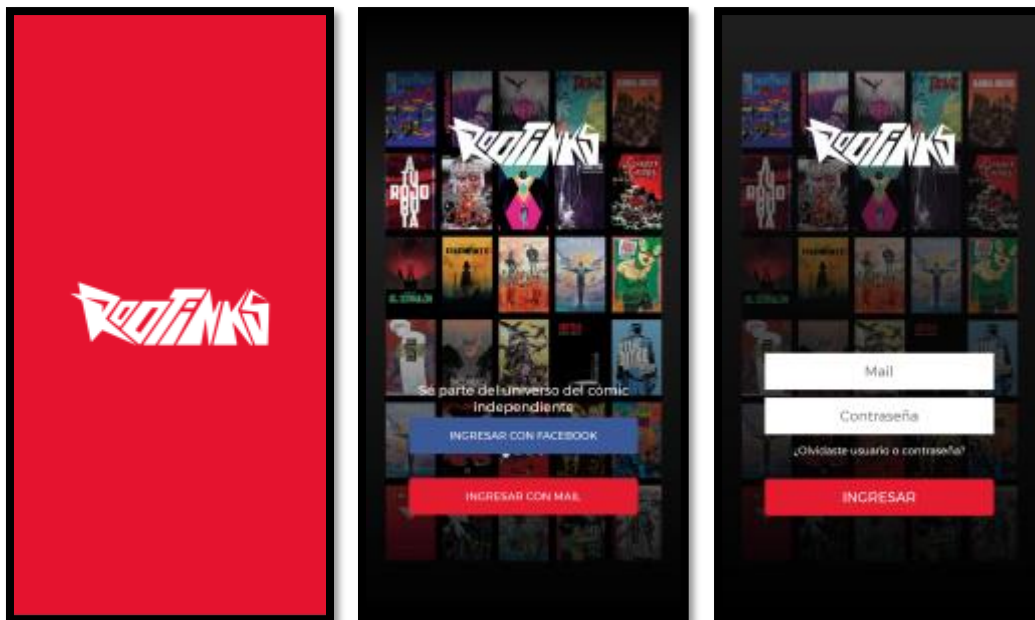


Figura 1. 51 Pantalla de login de usuarios en Rootinks.

#### Funcionalidad 2: Función de suscripción y usuario

La gestión de cuenta permite al usuario poder cambiar contraseña y realizar suscripción de pago para que pueda acceder a todo el contenido de la aplicación. El usuario también tiene acceso a las descargas para saber cuáles cómics se tienen sin acceso a internet, además de contar con información de la versión de la aplicación.

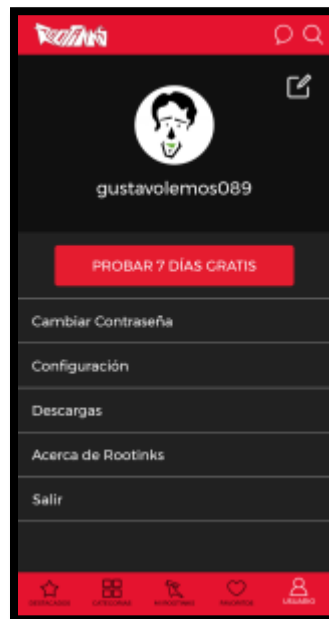


Figura 1. 52 Usuario registrado en Rootinks.

### Funcionalidad 3: Función de catálogo

El catálogo es bastante visual y dinámico, el mismo muestra las recomendaciones y novedades, además de los autores destacados de la aplicación. En la parte superior se muestran en un banner las noticias de los desarrolladores de la plataforma. En la sección categorías se pueden visualizar las obras divididas por: aventura, fantasía, humor, infantiles, etcétera.

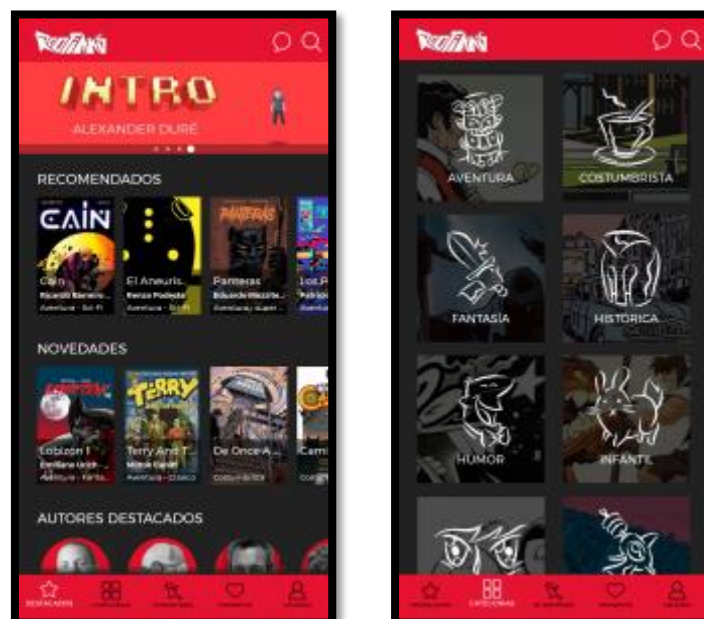


Figura 1. 53 Pantalla de cómics y categorías.

Otra de las opciones a la que puede acceder el usuario, es a la información de los autores y sus obras. Los usuarios autores pueden ver a los seguidores que tiene, una breve biografía y la lista de comics en los que ha participado de alguna forma.

Los usuarios lectores (que son seguidores del autor) son notificados cuando el autor sube nuevo contenido.



Figura 1. 54 Pantalla que muestra la información de las publicaciones del autor en Rootinks.

#### Funcionalidad 4: Función de favoritos y marcadores

Cada usuario puede armar su propia biblioteca con marcadores y favoritos. Además, puede continuar la lectura de un cómic desde un punto señalado, o simplemente seguir el desarrollo de una obra en particular.



Figura 1. 55 Pantalla de marcadores, biblioteca del lector.

### Funcionalidad 5: Función de lectura

El lector de la aplicación es muy completo, tiene una previsualización para cada página del cómic, y de esa manera el usuario poder acceder a todas las secciones. También el usuario tiene la posibilidad de añadir notas, y visualizar de forma completa una obra, o una página en particular.

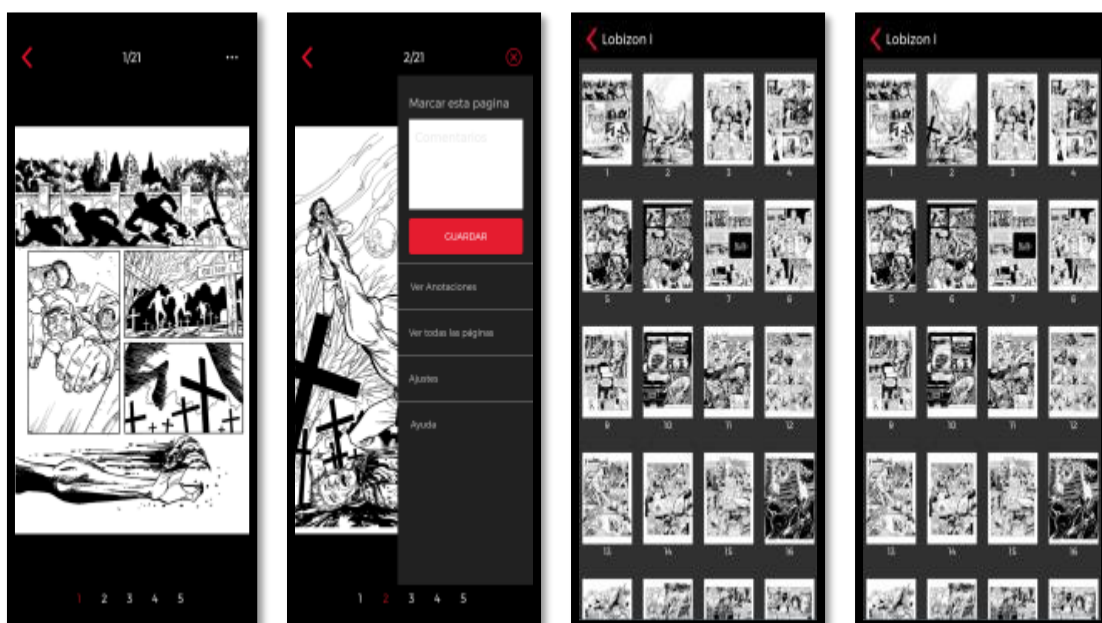


Figura 1. 56 Pantalla de visualización de cómics en Rootinks.

## 2.2. Modelo Lógico del Sistema Actual

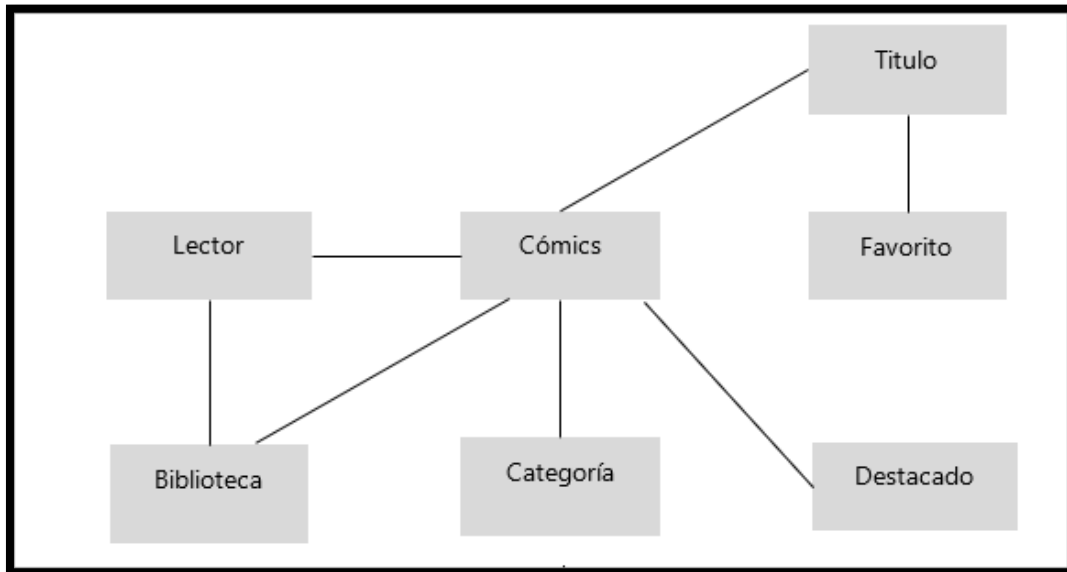


Diagrama 1. 13 Modelo de entidades de Rootinks.

El **DIAGRAMA 1. 13** es una aproximación genérica del modelo de relación de entidades que tiene la aplicación de Rootinks. Permite visualizar la relación entre el usuario y los cómics, además de mostrar los datos que poseen los cómics.

## 2.3. Problemas y necesidades detectadas en las funciones relevadas en detalle y en su entorno organizacional

### Funcionalidad 1: Función de login

Problema al realizar Loguin con Facebook:

- ❖ Al querer el usuario loguearse utilizando su cuenta de Facebook, la aplicación de *Rootinks* muestra un error que lo impide.



Figura 1. 57 Pantalla de error de logueo con Facebook.



## Funcionalidad 7: Función de publicación de obra

### Problema al querer subir un cómic:

- ❖ Si el autor desea subir un cómic, no tiene una sección para hacerlo. El usuario debe enviar un email al administrador de la aplicación, lo cual no es práctico ya que genera una demora de tiempo considerable entre el tiempo en que dicha obra será revisada (para ser aprobada o no) y luego publicada en caso de ser aprobada.
- ❖ Una solución posible puede ser agregar un botón que sea intuitivo, basado en un formulario, y le permite al autor subir su material, y poder luego corroborar el estado del mismo, es decir si fue revisado, rechazado o ha sido aceptado.

## Comparación y Conclusiones:

Luego de analizar cada aplicación, podemos observar que *Mangatoon* y *Tapas* destacan por su disponibilidad en múltiples plataformas, ya que se basan en una arquitectura sencilla, en donde el renderizado se realiza del lado del servidor, y los visores integrados del lado del front-end son simples, cuya navegación se basa únicamente en el pasaje de capítulos o episodios.

Cada capítulo o episodio son imágenes presentadas (y cargadas) consecutivamente, el usuario navega verticalmente (scroll) en el capítulo.

*Tapas* se destaca por su forma de apoyo a autores, permite al usuario poder realizar donaciones directas al autor/es de su preferencia, el cual cuenta con un perfil similar al de una red social.

*Webtoon* no ofrece ninguna novedad en cuanto a la forma de suscripción, todo su esfuerzo se invierte como distribuidor de contenido.

*Crunchyroll* y *Rootinks* son aplicaciones nativas para móviles, lo que les da un plus en personalización y performance.

*Crunchyroll* cuenta con una enorme comunidad, a pesar de esto, mantiene una interfaz de usuario simple, sobre todo en su visor. Además, recuerda hasta dónde ha visualizado cada usuario, demostrando un buen nivel de personalización.

*Rootinks*, si bien tiene una comunidad más reducida, presenta un visor de cómics totalmente configurable, es simple y eficiente.

Se puede destacar el sistema de apoyo a autores (donaciones) de *Tapas*, el visor o lector digital de *Rootinks*, y la simpleza y personalización de *Crunchyroll*.

	<i>Mangatoon</i>	<i>Tapas</i>	<i>Webtoon</i>	<i>Crunchyroll</i>	<i>Rootinks</i>
Gestión de usuario	-----	Incluye perfil estilo Microblogging.	Básico	Cuenta de lector.	Lector
Catálogo y filtros	Contenido pobre y poco variado.	Variado. Filtros de categorías y tendencias.	Variado	Variado. Múltiples filtros.	Contenido pobre. Relleno.
Lectura y lector de cómics	Básico	Sencillo. Permite calificar y comentar obra.	Sólo visualización de imágenes.	Completo en funcionalidades	Completo en funcionalidades. Previsualización de páginas.
Publicación	Sí.	Sí.	No	No	No

<b>de obras mediante plataforma</b>	Aprobación por administrador. Solicitud de contrato para obras reconocidas.	Aprobación por administrador			
<b>Remuneración a autores</b>	Donación únicamente por aplicación móvil. Contrato por obra.	Donación en plataforma con créditos (moneda virtual del sitio)	Contrato fuera de plataforma.	Contrato fuera de plataforma.	Contrato fuera de plataforma.
<b>Suscripción paga</b>	No	No	No	Sí	Sí
<b>Reseñas</b>	No	<i>Like</i>	Calificación	<i>Like</i>	<i>Like</i>
<b>Marcadores o favoritos</b>	No	Sí	Sí	Sí	Sí

Tabla 1. 1 Comparación de funcionalidades de sistemas relevados.

## 2.4. Nuevo Sistema

### 1. Objetivos y alcances preliminares del nuevo Sistema

*Omic*s es una plataforma web que permite a los usuarios acceder y leer los cómics que ofrecen diversos autores desde un mismo lugar, y pudiendo apoyar a sus obras favoritas.

A su vez, provee la posibilidad a distintos autores independientes de cómics de publicar sus trabajos, pasando por un proceso de selección.

Es el "Spotify" de los cómics donde pueden darse a conocer artistas que no tienen el alcance de grandes editoriales como DC, Marvel o Dark Horse. Enfatiza su cercanía al usuario, el cual no tiene que invertir en papel, y puede seguir sus gustos según autor, categoría, estilo, etcétera.

El sistema se compone de 2 tipos de usuarios: lector-autor, y administradores. Un lector es el usuario final de la aplicación, el autor es un usuario que ha publicado alguna obra en la plataforma, y el administrador es designado por los administradores de *Omic*s, para realizar la selección de obras cargadas por los autores.

#### 1.1. Objetivo general

Distribuir contenido (obras, cómics, historietas) ofreciéndose como catálogo a usuarios lectores. Permitirles filtrar el contenido bajo múltiples criterios.

Brindar la posibilidad de reseñar, suscribirse y agregar a favoritos los cómics.

Por otro lado, los autores deben poder publicar sus obras, con una previa aprobación por parte del administrador.

Los lectores pueden donar a los autores creadores de contenido.

#### 1.2. Funcionalidades y Requisitos

##### Lector

- ❖ Ver catálogo de cómics.
  - Búsqueda.
  - Filtro.
    - Por categoría.
    - Por fecha de publicación (Desde-Hasta).
  - Ordenamiento.
    - Valoración.
    - Likes.

- Títulos.
- Visualizaciones.
- ❖ Lectura.
  - Lector digital con funcionalidades agregadas para valorar o reseñar obra.
- ❖ Donar a autores.
- ❖ Suscripción a la plataforma.
- ❖ Marcadores, seguimiento, y reseña de obras.
  - Agregar obras a favoritos.
  - Seguir Autor.
  - Calificar, reseñar y comentar obra.

### **Autor**

- ❖ Subir sus obras. Carga de contenido.
  - Quedan pendiente de aprobación.
- ❖ Baja de sus obras.
- ❖ Donar a otros autores.
- ❖ Gestión de los datos y del contenido de la obra.
- ❖ Suscripción a la plataforma.

### **Administrador**

- ❖ Visualizar obras pendientes de aprobación.
- ❖ Aprobar o rechazar obras.
- ❖ Configuraciones de reglas de negocio
- ❖ Reportes e indicadores.
  - Visualización de publicaciones en tendencia.
  - Cantidad de visualizaciones.
  - Cantidad de comentarios.
  - Cantidad de usuarios (totales y parciales).
  - Cantidad de publicaciones y colecciones.
  - Ingresos y egresos de la plataforma.

## **1.3. Alcance**

Para poder especificar el alcance que tiene la plataforma *Omics*, se incluyen los procesos necesarios para asegurar que el proyecto integra todas las actividades que permitan completarlo de manera exitosa. Esto se logra cumpliendo con los requerimientos específicos y alcanzando los objetivos propuestos.

La funcionalidad de la plataforma hace hincapié en la división de *back-end* y *front-end*. En el *back-end* se proporciona una API (Application Program Interface) para satisfacer las necesidades, las consultas y solicitudes del *front-end*. El *front-end*, es decir la capa de presentación, estará conformado por las vistas del usuario.

El sistema se basa en el desarrollo de una plataforma web, adaptado a móvil, para la visualización de cómics y la carga de obras de los autores. Está formada por un panel de gestión de cómics que permitirá la subida de los cómics por parte de los usuarios autores, previamente que se hayan registrado.

Cada obra que se suba a la plataforma está pendiente de aprobación por un "Encargado de control de cómics", esto hace que todo el material subido a la plataforma cumpla las normas impuestas por *Omic*s para su contenido. La aplicación móvil es la encargada de brindar la posibilidad de lectura de cómics de una manera más cómoda y de fácil acceso.

Se realiza la gestión de roles, permisos y distintos niveles de usuarios para el manejo de las cuentas, y así brindar seguridad.

La plataforma brinda un apoyo económico a los autores mediante la API de Mercado Pago, la cual permite que parte del dinero ingresado quede para *Omic*s en concepto de distribuidor de contenido.

El usuario lector, puede realizar una donación a uno o varios autores según sus preferencias. A su vez, consigue valorar el trabajo del autor a través de un botón de "me gusta" y "seguidores", para recibir notificaciones y ver si ha publicado obras nuevas.

Un lector se convierte en autor creando una obra para publicar en la plataforma. También recibe notificaciones y puede ver si otros autores han publicado obras nuevas.

El usuario autor tiene una o muchas obras publicadas en la plataforma, para lo cual tiene creada una cuenta en Mercado Pago y anexada en *Omic*s. La cuenta de Mercado Pago le permite al autor poder cobrar las donaciones que los lectores le realizan al autor y/o a una obra, para tal fin hay un botón de "donación" en la publicación del autor.

Una obra está formada por una portada (imagen que se relaciona con lo que se publica), uno o muchos episodios, que a su vez están relacionadas con una o muchas imágenes que acompañan la temática que se está contando.

El sistema cuenta con un repositorio de imágenes y archivos, del cual existe un servicio para procesamiento de las mismas.

## 1.4. Módulos

### Módulos:

#### **1. Registro y Login de Usuario**

Los usuarios se registran directamente con un email y estableciendo una contraseña. Se pide: email, nombre de usuario (con el que se identifica en *Omic*s), contraseña y reescribir contraseña. Si el usuario ya está registrado, ingresa directamente con su nombre de usuario y contraseña. De esta manera se lo dirige al sector principal de la plataforma visualizando novedades y funcionalidades que hay en la misma.

#### **2. Seguridad y Roles**

Los niveles de usuario son los siguientes:

- ❖ Administrador: Es el Superusuario, y tiene permisos globales.
- ❖ Administrador Controlador de contenido: Es el encargado de revisar el material enviado por los autores de cómics.
- ❖ Autor: Tiene permisos para poder enviar material para ser publicado en la plataforma y acceder a información sobre las reacciones y reseñas de sus obras.
- ❖ Lector: Es el usuario que tiene permisos para visualizar contenido, agregar publicaciones a favoritos, seguir comics, calificar, reseñar, comentar las obras, y realizar pagos como apoyo a los autores.

#### **3. Pago de Apoyo a Obras**

El pago a los autores de obras (comics) se realiza a través de la API de Mercado Pago, la actual actúa de intermediario.

El usuario lector tiene dos opciones para apoyar económicamente a los autores. Una es mediante la suscripción a la plataforma, y otra forma es mediante donaciones que puede hacer. Para ello elige el monto a donar al autor, y se le notifica en caso de ser exitosa o no la transacción realizada.

#### **4. Catálogo**

El contenido de la plataforma está compuesto por cómics de los diferentes autores con sus diversos géneros y estilos. La composición principal de estos es: portadas, página, género, título, año de publicación original y una breve descripción de este.

El catálogo será donde el usuario lector visualiza todo el contenido disponible, el cual puede filtrar, buscar y agregar a su colección de favoritos.

## **5. Publicación de Obras**

Esta función la realiza el autor de los cómics, gestionada desde la plataforma. Una vez realizada el alta como autor, cada obra a subir a la plataforma pasa por un formulario y un período de revisión, para controlar que no infrinja algunas normas y políticas de *Omics*. El panel de publicación es independiente al catálogo.

## **6. Reseña de Cómics**

Esta función la realiza el cualquier usuario (lector o autor) que esté suscripto a la plataforma y para ello tiene una limitación en cantidad de caracteres. Además, tiene la opción de decir si es positivo o negativo el cómic.

La reseña y calificaciones en su conjunto conforman un promedio en la obra para que el autor pueda apreciar tanto si es del agrado de los lectores o no, para poder así hacer modificaciones futuras, ya que las mismas se miden según el tiempo (al principio tener reseñas negativas y luego positivas, o viceversa).

## **7. Repositorio de Cómics**

Es un repositorio de imágenes y archivos, almacenamiento, y procesamiento gestionado en un servidor de terceros (contratado). Este repositorio tiene la información detallada de imágenes de cada obra, y con sus respectivos autores para su visualización en la plataforma.

## **8. Reportes**

La plataforma *Omics* recibe constantemente información de los diferentes tipos de usuarios y del contenido que estos generan, por lo que es importante generar reportes para poder tomar futuras decisiones sobre la comunidad que utiliza la plataforma, como así también generar reportes para los autores.

Por lo tanto, se puede diferenciar dentro de este módulo, tres categorías de reportes:

- ❖ Reportes para la Organización: Estos reportes son de índole privados, y tienen toda la información sobre el uso de la plataforma, los usuarios (lectores y autores) y como ésta va variando en el tiempo. En una primera instancia lo importante es recopilar la mayor cantidad posible de datos de la comunidad de *Omics* para poder corregir errores, y poder hacer una plataforma atractiva tanto para lectores como para los autores. Los datos importan son: rango de edad, sexo, preferencias de lectura, autores más seguidos, tiempo de uso de la plataforma en un período de tiempo.



- Reporte rango de edades de los lectores y autores.
- Reporte cantidad de cada género de los lectores y autores.
- Reporte cantidad de cada región de los lectores y autores.
- Reporte cantidad de seguidores del autor.
- Reporte categorías de cómics más leídas.

Del Módulo "Reportes para la organización" se desprenden los siguientes **Indicadores:**

- **Edad** del lector y autor.
  - **Género** del lector y autor.
  - **Región** del lector y autor (país, provincia, ciudad, estado).
  - **Número de Seguidores** que tiene el autor.
  - **Número de Categorías de cómics** más leídas por los lectores.
  - **Número de Colecciones** de cada categoría de cómics más leídas por los lectores
  - **Contenido de los cómics** más leídos por los lectores
- ❖ Reportes Públicos: Estos se desprenden de los anteriores, pero se visualiza información menos sensible y más general sobre el uso de la plataforma, ya que dichos reportes están dirigidos a la comunidad de lectores de *Omic*s. La misma posee información de los cómics más leídos entre rangos de tiempo y regiones, cómo también los autores más populares. Realizar estos reportes públicos mensuales y darlos a conocer, da un sentido de pertenencia y transparencia que hace que más usuarios quieran utilizar la plataforma.
- Reporte de seguidores del autor.
  - Reporte categorías más leídas.
  - Reporte sobre cantidad de positivos/negativos que tiene el cómic.

Del Módulo "Reportes Públicos" se desprenden los siguientes **Indicadores:**

- **Número de Seguidores** que tiene un autor.
  - **Número de Categorías** de cómics más leídas por los lectores.
  - **Número de Colecciones** de cada categoría de cómics más leídas por los lectores.
  - **Número de Positivos/Negativos** dejados por los lectores por cada obra.
- ❖ Reporte de autores: Esta información también se desprende del primero mencionado, pero es más detallados y específico para cada autor, ya que son los que generan el contenido. Estos reportes tienen el objetivo de mantener informados a los autores sobre cómo fue la respuesta de los lectores con respecto a sus creaciones, para que con dicha información los autores hagan mejoras con

el fin de atraer a mayores lectores a la plataforma. Algunos de los datos de interés que se incluyen en estos reportes accesibles en cualquier momento son: cantidad de seguidores, rango de edad, interactividad con sus obras (veces que realizan reseñas, comentan y valoran el comic), rango de edades según las obras publicadas, montos generados por períodos de tiempo del dinero que dan como apoyo los lectores y visualizaciones de cada uno de sus cómics.

- Reporte cantidad de seguidores que tiene el autor.
- Reporte cantidad de positivos/negativos que tiene el cómic.
- Reporte rango de donaciones que tiene el autor.
- Reporte categorías más leídas.
- Reporte colecciones más leídas.

Del Módulo "Reporte de autores" se desprenden los siguientes **Indicadores**:

- **Número de Seguidores** que tiene un autor.
- **Número de Categorías** de cómics más leídas por los lectores.
- **Número de Colecciones** de cada categoría de cómics más leídas por los lectores.
- **Contenido de los cómics** más leídos por los lectores.
- **Número de Autores** de obras que reciben más donaciones.
- **Número de Autores** de obras que tienen más suscriptores (en la siguiente iteración se tendrán en cuenta)
- **Número de Positivos/Negativos** dejados por los lectores en cada obra.

# Diseño

## 1. Objetivos y alcances definitivos del nuevo sistema

### 1.1. Objetivos

Los objetivos del desarrollo de la plataforma *Omic*s se centran en los siguientes puntos:

- ❖ Proporcionar una plataforma de cómics con un catálogo de calidad.
- ❖ Ofrecer un sitio de distribución de contenido para los creadores de cómics.
- ❖ Acercar la comunidad de lectores de cómics con los autores de estos.
- ❖ Mejorar la visibilidad de escritores y dibujantes de cómics regionales.
- ❖ Brindar un lector digital de cómics intuitivo y sencillo a los lectores.
- ❖ Otorgar un medio digital económicamente sustentable para los creadores de cómics.
- ❖ Disminuir la percepción de "baja calidad" de autores independientes frente a los grandes de la industria internacional.
- ❖ Brindar información medible a los autores de contenido sobre sus lectores y seguidores.

### 1.2. Alcances

El alcance definitivo de *Omic*s está definido por los siguientes módulos:

- ❖ **Módulo de Registro y Actividad de Usuario:**
  - El usuario se puede registrar en la plataforma con el e-mail (con nombre de usuario, correo electrónico y contraseña); todas las formas de registrarse requieren luego añadir más información personal. Al iniciar sesión, se pide al usuario una contraseña. Para configurar datos y cambios de contraseña, el usuario puede realizarlo desde su página de perfil.
- ❖ **Módulo de Catálogo y Búsqueda:**
  - El usuario puede realizar búsquedas tanto por palabras claves, como por etiquetas que representen a alguna obra (género, autor, calificación por likes, etcétera) para poder filtrar contenido y encontrar alguno que se adecue a sus preferencias. Cada obra y colección (que es un conjunto de obras en sí) tiene detalles sobre

su fecha de publicación, autor, descripción, género, y otros que el usuario puede acceder para conocer más sobre la misma.

❖ **Módulo de Visualización y Seguimiento:**

- El usuario al acceder a una obra puede visualizarla en pantalla completa hoja por hoja, señalar la página deseada, y moverse entre anterior o siguiente cómic dentro de una colección (en caso de estar en una). Desde la misma pantalla de visualización se puede realizar un seguimiento del autor como volver desde al punto que se encontraba desde la última vez que accedió a esa publicación. Otra de las formas de tener fácil acceso a los comics es mediante la biblioteca de favoritos, que se agrega o elimina en la pantalla de publicación de la obra.

❖ **Módulo de Reseñas:**

- La forma de realizar reseñas de una colección y/o publicación en la plataforma es mediante dos interacciones (se pueden realizar ambas o solo una): un comentario de forma de análisis junto a una clasificación del 1 a 5; o mediante "likes" sin ningún otro comentario agregado (estos son binarios: están o no están, se suma la cantidad total de todos los usuarios).

❖ **Módulo de Suscripción y Donación:**

- *Omic*s y los autores que participan en la plataforma, obtienen rédito económico tanto en donaciones que se realizan directamente de los usuarios lectores, como de las suscripciones. La donación se realiza desde la pantalla de la publicación que está visualizando el lector, o desde la página del perfil del autor, mediante la API de Mercado Pago donde se coloca un monto mínimo acompañado de un mensaje (opcional). El sistema de suscripción se hace mediante la misma API, pero desde el perfil del usuario o desde el menú de inicio. Se realiza un débito automático y cambia su estado de Usuario a Suscripto.

❖ **Módulo de Publicación de Obras:**

- Cualquier usuario puede ser autor al acceder a la pantalla de "Publicar Obra". Si es la primera vez, se pide que ingrese información necesaria para poder recibir pago de donaciones o de la plataforma. Luego de enviar su Solicitud para publicar una obra

y recibir la aprobación de esta, el autor puede gestionarla dentro de una colección, o modificar alguno de los detalles de esta.

❖ **Módulo de Notificaciones:**

- La plataforma envía notificaciones a los usuarios sobre nuevo contenido, cambios en la misma, actualización de algún autor y/o publicación que el lector esté realizando seguimiento, como así también si ha habido alguna noticia importante dada por el administrador del sistema.

❖ **Módulo de Contrato:**

- Una vez que una publicación de un autor alcanza cierta cantidad de visualizaciones y se activa la opción de solicitar contrato. En caso de requerir la solicitud, se espera la aprobación y el autor puede empezar a cobrar según sus vistas. Además, se reciben resúmenes generados por la plataforma con el monto obtenido por cada publicación que tenga contrato correspondiente al autor.

❖ **Módulo de Reportes:**

- El administrador del sistema puede Generar Reportes sobre el contenido de la plataforma, los movimientos que han hecho los usuarios sobre las Publicaciones. En estos reportes se pueden visualizar: Las publicaciones que tienen o no Contrato vigente, la cantidad de Publicaciones por Categorías, y la cantidad de Publicaciones según su estado actual (Borrador, Rechazada, Aprobada, En Espera de Aprobación).

❖ **Módulo de Configuración:**

- El administrador puede Gestionar las Categorías de las Publicaciones, Gestionar las Publicaciones y los Contratos de los autores. Gestionar los Roles y permisos de los usuarios. Gestionar las Reglas de Negocio, y realizar el Backup de la plataforma.

## 2. Salidas del Sistema

Se organizan las salidas del sistema en un orden correspondiente con los Casos de Uso agrupados de manera general por módulos, de la siguiente forma:

### ❖ **Módulo de Registro y Actividad de Usuario**

- CU 01 - Registrar Cuenta de Usuario.
- CU 02 - Iniciar Sesión.
- CU 03 - Recuperar Contraseña.
- CU 04 - Cambiar Contraseña.
- CU 05 - Editar Cuenta de Usuario.
- CU 06 - Ver Perfil de Usuario.
- CU 07 - Desactivar Cuenta de Usuario.

### ❖ **Módulo de Catálogo y Búsqueda**

- CU 08 - Buscar y Filtrar Contenido.
- CU 09 - Ver Detalle de Contenido.
- CU 10 - Ver Favoritos.

### ❖ **Módulo de Visualización y Seguimiento**

- CU 11 - Leer Publicación.
- CU 12 - Añadir y Eliminar Contenido de Favoritos.
- CU 13 - Seguir/Dejar de Seguir a Usuario.

### ❖ **Módulo de Reseñas**

- CU 14 - Calificar Publicación.
- CU 15 - Dar Like a Publicación.

### ❖ **Módulo de Suscripción y Donación**

- CU 16 - Donar a Autor.
- CU 17 - Realizar Suscripción.
- CU 18 - Cancelar Suscripción.
- CU 19 - Pagar Suscripción.

### ❖ **Módulo de Publicación de Obras**

- CU 20 - Publicar Obra.
- CU 21 - Modificar Publicación.
- CU 22 - Gestionar Colecciones.
- CU 23 - Aceptar/Rechazar Publicación.
- CU 24 - Ver Estadísticas de Publicaciones.

❖ **Módulo de Notificaciones**

- CU 25 - Notificar Actividad de Autores y Contenido.
- CU 26 - Notificar Resumen de Pago.

❖ **Módulo de Contrato**

- CU 27 - Requerir Contrato.
- CU 28 - Rescindir Contrato.
- CU 29 - Cobrar por Resumen de Publicación.
- CU 30 - Generar Resumen de Publicación.
- CU 31 - Aceptar/Rechazar Contrato.

❖ **Módulo de Reportes**

- CU 32 - Generar Reportes de Usuarios.
- CU 33 - Generar Reportes de Publicaciones.
- CU 34 - Generar Reportes de Ingresos y Egresos.

❖ **Módulo de Configuración**

- CU 35 - Configurar Reglas de Negocio.
- CU 36 - Gestionar Categorías.
- CU 37 - ABM Roles.

A continuación, se muestran las salidas de información producidas por el sistema, que generan los módulos, con sus respectivos Casos de Uso. La mayoría de las salidas consisten en información mostrada por pantalla en la interacción con el usuario:

## 2.1. Módulo de Registro y Actividad de Usuario

Salida Nº 1	Acerca de Nosotros
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre: String</li> <li>• Descripción de Omics: String</li> </ul>
<b>CU Relacionado</b>	CU 02 - Iniciar Sesión.

Salida Nº 2	Registrarse en la Plataforma
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Contraseña: String</li> <li>• Confirmar contraseña: String</li> <li>• Correo electrónico: String</li> <li>• Token: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 01 - Registrar Cuenta de Usuario.

Salida Nº 3	Inicio de Sesión
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Contraseña: String</li> <li>• Token: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 02 - Iniciar Sesión.

Salida Nº 4	Recuperar la Contraseña
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Contraseña: String</li> <li>• Nueva contraseña: String</li> <li>• Confirmar contraseña: String</li> <li>• Código de Validación único: String</li> <li>• Correo electrónico: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 03 - Recuperar Contraseña.



Salida Nº 5	Cambiar la Contraseña
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Nueva contraseña: String</li> <li>• Confirmar contraseña: String</li> <li>• Correo electrónico: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 04 - Cambiar Contraseña.

Salida Nº 6	Editar Cuenta
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre: String</li> <li>• Apellido: String</li> <li>• Correo electrónico: String</li> <li>• Foto de Perfil: Buffer</li> <li>• Descripción Biografía: String</li> <li>• Contraseña: String</li> <li>• Nueva contraseña: String</li> <li>• Confirmar contraseña: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 05 - Editar Cuenta de Usuario.

Salida Nº 7	Ver Perfil
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Nombre: String</li> <li>• Apellido: String</li> <li>• Foto de Perfil: Buffer</li> <li>• Descripción Biografía: String</li> <li>• Contraseña: String</li> <li>• Correo electrónico: String</li> <li>• Fecha: Date</li> </ul>

<b>CU Relacionado</b>	CU 06 - Ver Perfil de Usuario.
-----------------------	--------------------------------

Salida N° 8	Desactivar Cuenta
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Nombre: String</li> <li>• Apellido: String</li> <li>• Foto de Perfil: Buffer</li> <li>• Correo electrónico: String</li> <li>• Descripción Biografía: String</li> <li>• Contraseña: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 07 - Desactivar Cuenta de Usuario.

## 2.2. Módulo de Catálogo y Búsqueda

Salida N° 9	Mis Filtros y Contenidos
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre de publicación: String</li> <li>• Nombre de colección: String</li> <li>• Nombre de categoría: String</li> <li>• Fecha desde: Date</li> <li>• Fecha hasta: Date</li> </ul>
<b>CU Relacionado</b>	CU 08 - Buscar y Filtrar Contenido.

Salida N° 10	Ver el Contenido de la Obra
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre de publicación: String</li> <li>• Nombre de colección: String</li> <li>• Nombre de categoría: String</li> <li>• Nombre de etiqueta: String</li> <li>• Portada: Buffer</li> </ul>

	<ul style="list-style-type: none"> <li>• Fecha de publicación: Date</li> <li>• Fecha de última modificación: Date</li> <li>• Sinopsis: String</li> <li>• Nombre del autor: String</li> <li>• Estrellas: Integer</li> <li>• Likes: Integer</li> <li>• Fecha desde: Date</li> <li>• Fecha hasta: Date</li> </ul>
<b>CU Relacionado</b>	CU 09 - Ver Detalle de Contenido.

Salida N° 11	Ver Contenido
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre de publicación: String</li> <li>• Nombre de categoría: String</li> <li>• Nombre de etiqueta: String</li> <li>• Portada: Buffer</li> <li>• Fecha de publicación: Date</li> <li>• Nombre del autor: String</li> <li>• Likes: Integer</li> <li>• Estrellas: Integer</li> </ul>
<b>CU Relacionado</b>	CU 10 - Ver Favoritos.

### 2.3. Módulo de Visualización y Seguimiento

Salida N° 12	Ver las Publicaciones
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre de publicación: String</li> <li>• Descripción: String</li> <li>• Estado: String</li> <li>• Nombre del autor: String</li> <li>• Imágenes: Buffer</li> <li>• Nombre de categoría: String</li> <li>• Nombre de colección: String</li> <li>• Nombre de etiqueta: String</li> <li>• Fecha: Date</li> </ul>

<b>CU Relacionado</b>	CU 11 - Leer Publicación.
-----------------------	---------------------------

<b>Salida N° 13</b>	<b>Agregar y Eliminar de Favoritos Contenido</b>
---------------------	--

<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre de publicación: String</li> <li>• Likes: Integer</li> <li>• Portada: Buffer</li> <li>• Nombre de colección: String</li> <li>• Nombre de categoría: String</li> <li>• Nombre del autor: String</li> <li>• Fecha: Date</li> </ul>
----------------------------------	---

<b>CU Relacionado</b>	CU 12 - Añadir y Eliminar Contenido de Favoritos.
-----------------------	---

<b>Salida N° 14</b>	<b>Ver Información del Usuario</b>
---------------------	------------------------------------

<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre de publicación: String</li> <li>• Likes: Integer</li> <li>• Nombre de colección: String</li> <li>• Nombre de categoría: String</li> <li>• Nombre del autor: String</li> <li>• Fecha: Date</li> <li>• Cantidad de seguidores: Integer</li> </ul>
----------------------------------	---

<b>CU Relacionado</b>	CU 13 - Seguir/Dejar de Seguir a Usuario.
-----------------------	---

## 2.4. Módulo de Reseñas

<b>Salida N° 15</b>	<b>Calificar Publicaciones</b>
---------------------	--------------------------------

<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre de publicación: String</li> <li>• Nombre de colección: String</li> <li>• Nombre de categoría: String</li> <li>• Nombre de etiqueta: String</li> </ul>
----------------------------------	---

	<ul style="list-style-type: none"> <li>• Nombre del autor: String</li> <li>• Portada: Buffer</li> <li>• Sinopsis: String</li> <li>• Likes: Integer</li> <li>• Vistas: Integer</li> <li>• Comentario: String</li> <li>• Estrellas: Integer</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 14 - Calificar Publicación.

Salida N° 16	Dar Like a Publicación
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre de publicación: String</li> <li>• Nombre de colección: String</li> <li>• Nombre de categoría: String</li> <li>• Nombre del autor: String</li> <li>• Nombre de etiqueta: String</li> <li>• Portada: Buffer</li> <li>• Sinopsis: String</li> <li>• Likes: Integer</li> <li>• Vistas: Integer</li> <li>• Estrellas: Integer</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 15 - Dar Like a Publicación.

## 2.5. Módulo de Suscripción y Donación

Salida N° 17	Realizar Donación a Autor
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre de publicación: String</li> <li>• Sinopsis: String</li> <li>• Likes: Integer</li> <li>• Nombre de colección: String</li> <li>• Nombre de categoría: String</li> <li>• Nombre del autor: String</li> <li>• Vistas: Integer</li> </ul>

	<ul style="list-style-type: none"> <li>• Monto: Float</li> <li>• Fecha: Date</li> <li>• Comentario: String</li> <li>• Nombre usuario: String</li> </ul>
<b>CU Relacionado</b>	CU 16 - Donar a Autor.

Salida N° 18	Suscribirse a la Plataforma
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Nombre: String</li> <li>• Apellido: String</li> <li>• Foto de Perfil: Buffer</li> <li>• Descripción Biografía: String</li> <li>• Contraseña: String</li> <li>• Correo electrónico: String</li> <li>• Código de Suscripción: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 17 - Realizar Suscripción.

Salida N° 19	Cancelar la Suscripción
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Nombre: String</li> <li>• Apellido: String</li> <li>• Foto de Perfil: Buffer</li> <li>• Descripción Biografía: String</li> <li>• Contraseña: String</li> <li>• Correo electrónico: String</li> <li>• Código de Suscripción: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 18 - Cancelar Suscripción.

Salida N° 20	Pagar Suscripción
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Contraseña: String</li> <li>• Monto Total: Float</li> <li>• Correo electrónico: String</li> <li>• Código de Suscripción: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 19 - Pagar Suscripción.

## 2.6. Módulo de Publicación de Obras

Salida N° 21	Publicar mi Obra
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Correo electrónico: String</li> <li>• Nombre de publicación: String</li> <li>• Nombre de colección: String</li> <li>• Nombre de categoría: String</li> <li>• Nombre de etiqueta: String</li> <li>• Nombre estado: String</li> <li>• Sinopsis: String</li> <li>• Portada: Buffer</li> <li>• Imágenes: Buffer</li> <li>• Cantidad de páginas: Integer</li> <li>• Tamaño: Float</li> <li>• Url: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 20 - Publicar Obra.

Salida N° 22	Editar Obra
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Correo electrónico: String</li> <li>• Nombre de publicación: String</li> </ul>

	<ul style="list-style-type: none"> <li>• Nombre de categoría: String</li> <li>• Nombre de etiqueta: String</li> <li>• Sinopsis: String</li> <li>• Portada: Buffer</li> <li>• Imágenes: Buffer</li> <li>• Cantidad de páginas: Integer</li> <li>• Tamaño: Float</li> <li>• Url: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 21 - Modificar Publicación.

Salida N° 23	Editar Colecciones
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Nombre de publicación: String</li> <li>• Nombre de colección: String</li> <li>• Nombre de categoría: String</li> <li>• Sinopsis: String</li> <li>• Cantidad de publicaciones: Integer</li> <li>• Estrellas: Integer</li> <li>• Likes: Integer</li> <li>• Fecha: Date</li> <li>• Url: String</li> </ul>
<b>CU Relacionado</b>	CU 22 - Gestionar Colecciones.

Salida N° 24	Agregar o Rechazar Publicación
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Nombre de publicación: String</li> <li>• Nombre estado: String</li> <li>• Comentario: String</li> <li>• Fecha desde: Date</li> <li>• Fecha hasta: Date</li> </ul>



<b>CU Relacionado</b>	CU 23 - Aceptar/Rechazar Publicación.
-----------------------	---------------------------------------

Salida N° 25	Estadísticas de Publicaciones
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Nombre de publicación: String</li> <li>• Nombre de colección: String</li> <li>• Nombre de categoría: String</li> <li>• Estrellas: Float</li> <li>• Likes: Integer</li> <li>• Vistas: Integer</li> <li>• Comentario: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 24 - Ver Estadísticas de Publicaciones.

## 2.7. Módulo de Notificaciones

Salida N° 26	Recibir Notificaciones de Autor
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre de publicación: String</li> <li>• Nombre de colección: String</li> <li>• Nombre de categoría: String</li> <li>• Nombre del autor: String</li> <li>• Descripción: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 25 - Notificar Actividad de Autores y Contenido.

Salida N° 27	Recibir Notificación de Resumen de Pago
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Vistas: Integer</li> <li>• Vistas únicas: Integer</li> <li>• Fecha desde: Date</li> </ul>

	<ul style="list-style-type: none"> <li>• Fecha hasta: Date</li> <li>• Subtotal: Float</li> <li>• Monto Total: Float</li> <li>• Nombre de publicación: String</li> </ul>
<b>CU Relacionado</b>	CU 26 - Notificar Resumen de Pago.

## 2.8. Módulo de Contrato

Salida N° 28	Solicitar Contrato
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Nombre de publicación: String</li> <li>• Sinopsis: String</li> <li>• Estrellas: Float</li> <li>• Nombre estado: String</li> <li>• Likes: Integer</li> <li>• Vistas: Integer</li> <li>• Comentario: String</li> <li>• Número de contrato: Integer</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 27 - Requerir Contrato.

Salida N° 29	Rescindir Contrato
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Nombre de publicación: String</li> <li>• Sinopsis: String</li> <li>• Estrellas: Float</li> <li>• Likes: Integer</li> <li>• Número de contrato: Integer</li> <li>• Vistas: Integer</li> <li>• Nombre estado: String</li> <li>• Comentario: String</li> <li>• Fecha: Date</li> </ul>

<b>CU Relacionado</b>	CU 28 - Rescindir Contrato.
-----------------------	-----------------------------

Salida N° 30	Cobrar Publicaciones
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Nombre de publicación: String</li> <li>• Sinopsis: String</li> <li>• Estrellas: Float</li> <li>• Likes: Integer</li> <li>• Comentario: String</li> <li>• Vistas: Integer</li> <li>• Vistas únicas: Integer</li> <li>• Subtotal: Float</li> <li>• Monto Total: Float</li> <li>• Fecha desde: Date</li> <li>• Fecha hasta: Date</li> <li>• Fecha cobro: Date</li> </ul>
<b>CU Relacionado</b>	CU 29 - Cobrar por Resumen de Publicación.

Salida N° 31	Resumen Publicaciones
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre usuario: String</li> <li>• Nombre de publicación: String</li> <li>• Sinopsis: String</li> <li>• Estrellas: Float</li> <li>• Likes: Integer</li> <li>• Comentario: String</li> <li>• Vistas: Integer</li> <li>• Vistas únicas: Integer</li> <li>• Subtotal: Float</li> <li>• Monto Total: Float</li> <li>• Fecha desde: Date</li> <li>• Fecha hasta: Date</li> <li>• Fecha Generación Resumen: Date</li> </ul>

<b>CU Relacionado</b>	CU 30 - Generar Resumen de Publicación.
-----------------------	---

Salida N° 32	Verificar Contrato
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre del autor: String</li> <li>• Nombre estado contrato: String</li> <li>• Comentario: String</li> <li>• Número de contrato: Integer</li> <li>• Fecha desde: Date</li> <li>• Fecha hasta: Date</li> </ul>
<b>CU Relacionado</b>	CU 31 - Aceptar/Rechazar Contrato.

## 2.9. Módulo de Reportes

Salida N° 33	Ver Reporte de Usuarios
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Cantidad por edad: Integer</li> <li>• Edad: Integer</li> <li>• Cantidad por género: Integer</li> <li>• Nombre del género: String</li> <li>• Cantidad de usuarios total: Integer</li> <li>• Cantidad de lectores: Integer</li> <li>• Cantidad de autores: Integer</li> <li>• Cantidad nuevos usuarios: Integer</li> <li>• Cantidad abandono usuarios: Integer</li> <li>• Cantidad de seguidores: Integer</li> <li>• Cantidad de suscriptores: Integer</li> <li>• Fecha reporte: Date</li> <li>• Fecha desde: Date</li> <li>• Fecha hasta: Date</li> </ul>
<b>CU Relacionado</b>	CU 32 - Generar Reportes de Usuarios.

<b>Salida N° 34</b>	<b>Ver Reporte de Publicaciones</b>
<p><b>Elementos que lo componen</b></p>	<ul style="list-style-type: none"> <li>• Cantidad de publicaciones: Integer</li> <li>• Cantidad de colecciones: Integer</li> <li>• Cantidad por categoría: Integer</li> <li>• Nombre de categoría: String</li> <li>• Cantidad de visualizaciones: Integer</li> <li>• Cantidad de likes: Integer</li> <li>• Cantidad nuevas publicaciones: Integer</li> <li>• Cantidad de contratos total: Integer</li> <li>• Cantidad nuevos contratos: Integer</li> <li>• Fecha reporte: Date</li> <li>• Fecha desde: Date</li> <li>• Fecha hasta: Date</li> </ul>
<p><b>CU Relacionado</b></p>	<p>CU 33 - Generar Reportes de Publicaciones.</p>

<b>Salida N° 35</b>	<b>Ver Reporte de Ingresos y Egresos</b>
<p><b>Elementos que lo componen</b></p>	<ul style="list-style-type: none"> <li>• Montos pago a autores: Float</li> <li>• Montos de Suscripción: Float</li> <li>• Montos de Donación: Float</li> <li>• Monto total: Float</li> <li>• Fecha reporte: Date</li> <li>• Fecha desde: Date</li> <li>• Fecha hasta: Date</li> </ul>
<p><b>CU Relacionado</b></p>	<p>CU 34 - Generar Reportes de Ingresos y Egresos.</p>

## 2.10. Módulo de Configuración

Salida N° 36	Administrar Reglas de Negocio
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre de publicación: String</li> <li>• Nombre de categoría: String</li> <li>• Nombre de colección: String</li> <li>• Montos Mínimo Cobrar: Float</li> <li>• Montos Mínimo Donación: Float</li> <li>• Visualizaciones Mínimas Contrato: Integer</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 35 - Configurar Reglas de Negocio.

Salida N° 37	Administrar Categoría de Publicaciones
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre de categoría: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 36 - Gestionar Categorías.

Salida N° 38	Administrar Roles de usuario
<b>Elementos que lo componen</b>	<ul style="list-style-type: none"> <li>• Nombre del rol: String</li> <li>• Permisos asignados: String</li> <li>• Lista de permisos: String</li> <li>• Fecha: Date</li> </ul>
<b>CU Relacionado</b>	CU 37 - ABM Roles.

### 3. Modelo Funcional

#### 3.1. Diagrama de Casos de Uso

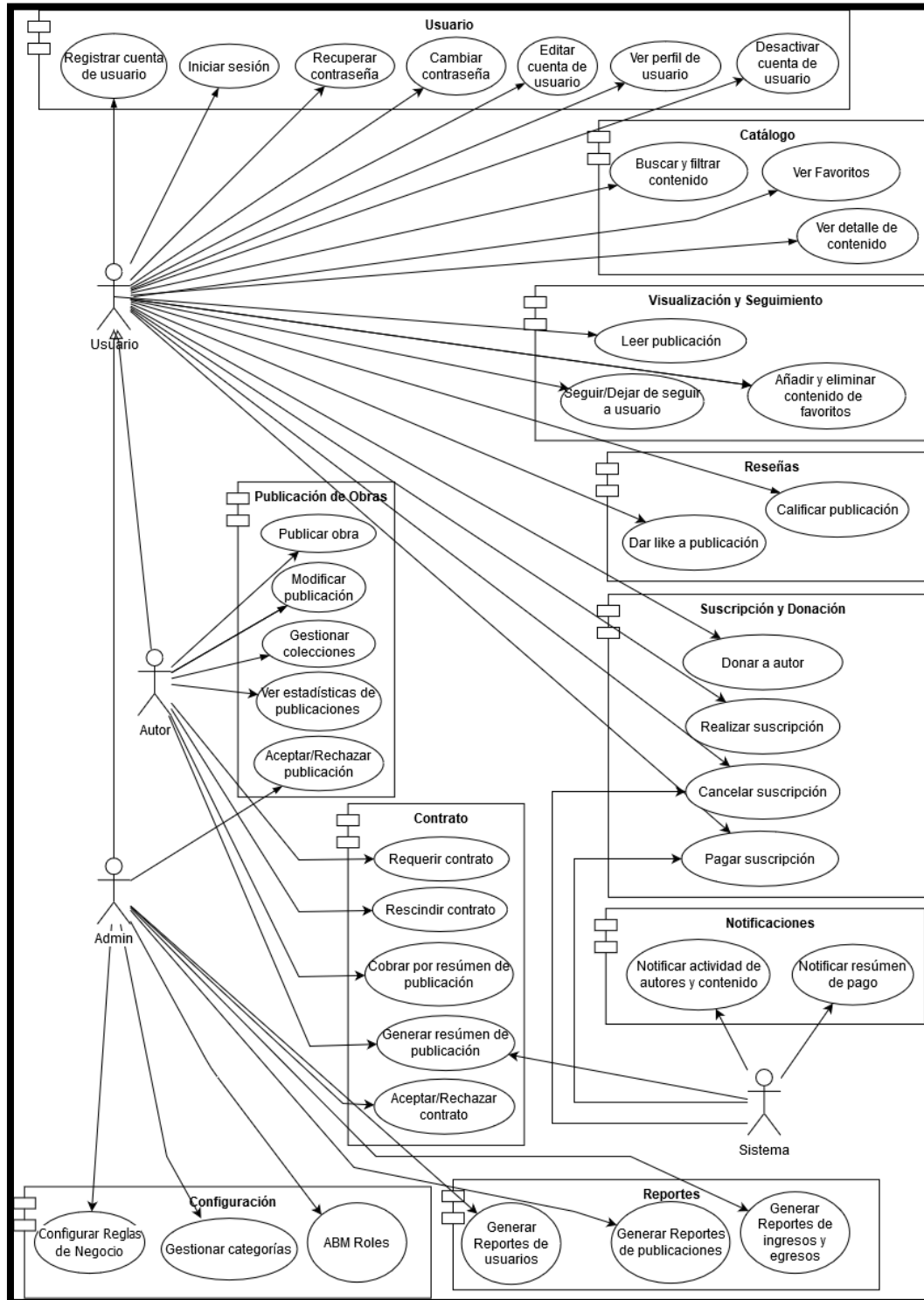


Diagrama 1. 14 Diagrama de Casos de Uso.

## 3.2. Historias de Usuario

Debido a que se aplica Scrum como metodología ágil para el desarrollo del sistema, se realiza el diseño del modelo funcional utilizando Historias de Usuario (User Story) la cual permite especificar los requisitos del sistema de una forma simple, aplicando entre los miembros del equipo el método INVEST.

Método INVEST: Utilizado para asegurar la calidad en la escritura de historias de usuario. El método sirve para comprobar la calidad de una historia de usuario revisando que cumpla una serie de características:

- ❖ I - Independent (independiente)
- ❖ N - Negotiable (negociable)
- ❖ V - Valuable (valiosa)
- ❖ E - Estimable (estimable)
- ❖ S - Small (pequeña)
- ❖ T - Testable (comprobable)

- **Independent (independiente)**

Es ventajoso que cada historia de usuario pueda ser planificada e implementada en cualquier orden. Para ello las historias deben ser totalmente independientes (lo cual facilita el trabajo posterior del equipo). Resaltar que las dependencias entre historias de usuario pueden reducirse combinándolas en una o dividiéndolas de manera diferente.

- **Negotiable (negociable)**

Una historia de usuario es una descripción corta de una necesidad que no incluye detalles. Las historias deben ser negociables ya que sus detalles serán acordados con el cliente o el usuario durante la fase de conversación. Por tanto, se debe evitar historias de usuario con demasiados detalles porque limitaría la conversación acerca de las mismas.

- **Valuable (valiosa)**

Una historia de usuario tiene que ser valiosa para el cliente o el usuario. Una manera de hacer una historia valiosa es que la escriba el mismo.

- **Estimable (estimable)**

Una buena historia de usuario debe de poder ser estimada con la precisión suficiente para ayudar al cliente, usuario o propietario del producto a priorizar y planificar su implementación. La estimación generalmente la realiza el equipo de trabajo y está



directamente relacionada con el tamaño de la historia de usuario (una historia de usuario de gran tamaño es más difícil de estimar) y con el conocimiento del equipo de la necesidad expresada (en el caso de falta de conocimiento, serán necesarias más fases de conversación acerca de la misma).

- **Small (pequeña)**

Las historias de usuario deben englobar como mucho unas pocas semanas/persona de trabajo. Incluso hay equipos que las restringen a días/persona. Una descripción corta ayuda a disminuir el tamaño de una historia de usuario facilitando así su estimación.

- **Testable (comprobable)**

La historia de usuario debe ser capaz de ser probada (fase confirmación de la historia de usuario). Si el cliente o usuario no sabe cómo probar la historia de usuario significa que no es del todo clara o que no es valiosa. Si el equipo no puede probar una historia de usuario nunca sabrá si la ha terminado o no.

### 3.2.1. Módulo de Registro y Actividad de Usuario

Historia de Usuario N.º 01 "Registrar Cuenta de Usuario"	
<b>Prioridad:</b>	Baja
<b>Estimación:</b>	5 días
<b>Dependiente de:</b>	-----
Como futuro usuario del sistema <b>quiero</b> poder registrarme <b>para</b> utilizar la plataforma y visualizar las obras que contiene la misma.	
Criterios de Aceptación	
<b>Cuando</b> seleccione la opción "Registrarse" en la página de inicio.	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Aparezca la opción para poder registrarme por e-mail.</li> <li>✓ Pueda ingresar los datos de usuario para registrarse por e-mail.</li> <li>✓ Aparezca la opción para leer los términos de servicios y políticas de seguridad.</li> </ul>
<b>Cuando</b> registre por e-mail.	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Pueda ingresar: e-mail, nombre de usuario y</li> </ul>

	<p>contraseña.</p> <ul style="list-style-type: none"> <li>✓ Pueda confirmar la contraseña en un campo distinto (reescribir contraseña).</li> <li>✓ Aparezca un botón para confirmar el registro.</li> </ul>
<p><b>Cuando</b> confirme registro.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se haya aceptado los términos y condiciones.</li> <li>✓ Valide que los datos ingresados tengan un formato correcto.</li> <li>✓ Valide que el email no haya sido usado anteriormente.</li> <li>✓ Valide que el nombre de usuario no exista previamente.</li> <li>✓ Valide que las contraseñas ingresadas sean iguales.</li> <li>✓ Me indique cuál sea el error, en caso de haber ingresado un dato incorrecto.</li> <li>✓ Me indique cuál sea el error, en caso de haber ingresado un formato incorrecto.</li> <li>✓ Si el registro es exitoso, recibir un correo electrónico al email ingresado, para confirmar mi cuenta e iniciar sesión automáticamente.</li> </ul>

**Referencias**

**Pantallas:**

- (1).Figura 1. 58 Pantalla N° 1: Pantalla de Bienvenida a Omics.
- (2).Figura 1. 59 Pantalla N° 2: Pantalla de Inicio Con Barra Lateral. Sin loguearse.
- (3).Figura 1. 60 Pantalla N° 3: Pantalla de Inicio Sin Barra Lateral. Sin loguearse.
- (4).Figura 1. 61 Pantalla N° 4: Pantalla para Registrarse.

<b>Historia de Usuario N.º 02 "Iniciar Sesión"</b>	
<b>Prioridad:</b>	Baja
<b>Estimación:</b>	3 días
<b>Dependiente de:</b>	01
<p><b>Como</b> usuario del sistema <b>quiero</b> poder iniciar sesión <b>para</b> acceder a las funcionalidades de la plataforma.</p>	
<b>Criterios de Aceptación</b>	

<p><b>Cuando</b> seleccione la opción "Iniciar sesión" en la página principal.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Aparezca la opción de iniciar sesión con email.</li> <li>✓ Aparezca la opción de recuperar contraseña en caso de que no la recuerde.</li> </ul>
<p><b>Cuando</b> inicie sesión con e-mail.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda ingresar mi nombre de usuario o e-mail y contraseña.</li> <li>✓ Aparezca un botón para confirmar el ingreso.</li> </ul>
<p><b>Cuando</b> seleccione la opción de "Ingresar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Valide que los datos ingresados coinciden con una cuenta existente.</li> <li>✓ Valide que los datos ingresados tengan un formato correcto.</li> <li>✓ Si el inicio de sesión no es exitoso, me indique cuál es el error en caso de haber ingresado un dato o formato incorrecto.</li> <li>✓ Si el inicio de sesión es exitoso, me redireccione a la página principal de mi cuenta.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b> (1).Figura 1. 62 Pantalla N° 5: Pantalla de Login.</p>	

<p><b>Historia de Usuario N.º 03</b> <b>"Recuperar Contraseña"</b></p>	
<p><b>Prioridad:</b></p>	<p>Baja</p>
<p><b>Estimación:</b></p>	<p>3 días</p>
<p><b>Dependiente de:</b></p>	<p>01, 02</p>
<p><b>Como</b> usuario del sistema <b>quiero</b> recuperar la contraseña que he olvidado <b>para</b> poder ingresar a la plataforma.</p>	
<p><b>Criterios de Aceptación</b></p>	
<p><b>Cuando</b> seleccione la opción "Recuperar Contraseña".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda ingresar el correo al que quiero que me envíen el enlace de restauración de contraseña.</li> </ul>

	<ul style="list-style-type: none"> <li>✓ Pueda validar que soy una persona.</li> <li>✓ Aparezca un botón para confirmar el envío del e-mail.</li> </ul>
<b>Cuando</b> seleccione en la opción que No soy un Robot.	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Se pueda continuar para que me envíen el e-mail al correo con el enlace de restauración de la contraseña.</li> </ul>
<b>Cuando</b> seleccione el botón "Enviar".	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Me envíe un e-mail al correo electrónico que ingresé para cambiar la contraseña por una nueva.</li> <li>✓ Me salga un error en caso de no haber validado que soy una persona.</li> </ul>
<b>Cuando</b> ingrese al e-mail que me han enviado al correo.	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Aparezca el enlace para restaurar la contraseña</li> </ul>
<b>Cuando</b> seleccione el enlace para restaurar la contraseña.	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Pueda ingresar la nueva contraseña.</li> <li>✓ Pueda confirmar la contraseña en un campo distinto (reescribir contraseña).</li> <li>✓ Aparezca un botón para reiniciar la contraseña.</li> </ul>
<b>Cuando</b> seleccione la opción "Reiniciar contraseña".	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Valide que las contraseñas ingresadas sean iguales.</li> <li>✓ Me indique un error si no se cumple con la cantidad de caracteres establecidos.</li> <li>✓ Si la restauración de la contraseña es exitosa, recibir un correo electrónico al email y que me dirija a la página principal de la plataforma para "Iniciar sesión".</li> </ul>
<b>Referencias</b>	
<b>Pantallas:</b> <b>(1).</b> Figura 1. 63 Pantalla N° 6: Pantalla de Enviar Recuperar Contraseña.	

<b>Historia de Usuario N.º 04 "Cambiar Contraseña"</b>	
<b>Prioridad:</b>	Baja
<b>Estimación:</b>	3 días
<b>Dependiente de:</b>	01, 02
<p><b>Como</b> usuario del sistema <b>quiero</b> cambiar la contraseña de mi cuenta <b>para</b> actualizarla.</p>	
<b>Criterios de Aceptación</b>	
<p><b>Cuando</b> seleccione la opción "Modificar Contraseña".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se visualice el campo para que pueda ingresar mi nombre de usuario.</li> <li>✓ Se visualice el campo para que pueda ingresar la contraseña actual.</li> <li>✓ Se visualice el campo para que pueda ingresar la nueva contraseña.</li> <li>✓ Se visualice el campo para que pueda ingresar nuevamente la nueva contraseña.</li> <li>✓ Aparezca un botón de "Confirmar".</li> </ul>
<p><b>Cuando</b> seleccione el botón "Aceptar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Me indique un error si la contraseña actual ingresada no es correcta.</li> <li>✓ Valide que las nuevas contraseñas ingresadas sean iguales.</li> <li>✓ Se visualice un error si alguna o ambas contraseñas ingresadas, no cumplen con la cantidad de caracteres establecidos.</li> <li>✓ Se visualice un error si las contraseñas ingresadas no son iguales.</li> <li>✓ Si la modificación de la contraseña es exitosa, recibir un correo electrónico al email, indicando el cambio de contraseña.</li> </ul>
<b>Referencias</b>	
<p><b>Pantallas:</b></p> <p>(1).Figura 1. 64 Pantalla N° 7: Pantalla de Nueva Contraseña.</p> <p>(2).Figura 1. 65 Pantalla N° 8: Pantalla de Usuarios Logueados.</p> <p>(3).Figura 1. 66 Pantalla N° 9: Pantalla de Permisos de Usuario, con menú</p>	

desplegable (lo de fondo violeta es los de acceso de administradores, y lo de azul para usuarios lectores y/o autores).

Historia de Usuario N.º 05 "Editar Cuenta de Usuario"	
<b>Prioridad:</b>	Media
<b>Estimación:</b>	6 días
<b>Dependiente de:</b>	01, 02
<p><b>Como</b> usuario del sistema <b>quiero</b> poder editar mi cuenta de usuario, seleccionando el dato a modificar <b>para</b> poder actualizar esa información.</p>	
Criterios de Aceptación	
<p><b>Cuando</b> seleccione la opción "Editar cuenta de usuario" dentro de mi cuenta.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se abra un formulario donde se muestre mis datos ingresados en la plataforma (usuario, dirección de correo, nombre, apellido, biografía y foto de perfil).</li> <li>✓ Pueda modificar cualquiera de estos datos: nombre, apellido, biografía y foto de perfil.</li> <li>✓ Pueda agregar una foto dentro de mi perfil de usuario en caso de no tener ninguna colocada.</li> <li>✓ Aparezca un botón para modificar.</li> </ul>
<p><b>Cuando</b> ingrese el/los datos que quiero modificar.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Queden completos el/los campos con la información que quiero actualizar.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Modificar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Valide que los datos ingresados tengan un formato correcto.</li> <li>✓ Me indique cuál sea el error, en caso de haber ingresado un dato o formato incorrecto.</li> <li>✓ Me muestre los cambios actualizados si no hubo ningún error al ingresar algún dato o formato.</li> </ul>
Referencias	

<p><b>Pantallas:</b>  <b>(1).</b>Figura 1. 67 Pantalla N° 10: Pantalla de Editar</p>
--

<b>Historia de Usuario N.º 06</b> <b>“Ver Perfil de Usuario”</b>	
<b>Prioridad:</b>	Media
<b>Estimación:</b>	6 días
<b>Dependiente de:</b>	05
<p><b>Como</b> usuario del sistema <b>quiero</b> poder ver mi perfil o la de otro usuario <b>para</b> ver de manera más detallada la información.</p>	
<b>Criterios de Aceptación</b>	
<p><b>Cuando</b> seleccione el botón “Mi Perfil” dentro de mi cuenta.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda ver mis datos: nombre de usuario, nombre y apellido, biografía, mis publicaciones con los datos de cada una de las publicaciones (portada, nombre de la publicación, descripción, autor, categoría, etiquetas, puntuación(estrellas)).</li> </ul>
<p><b>Cuando</b> seleccione el nombre o foto del autor.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Me redirija al perfil de ese autor.</li> <li>✓ Pueda ver sus datos: nombre de usuario, nombre y apellido, biografía, sus publicaciones con los datos de cada una de las publicaciones (portada, nombre de la publicación, descripción, autor, categoría, etiquetas, puntuación (estrellas)).</li> </ul>
<b>Referencias</b>	
<p><b>Pantallas:</b>  <b>(1).</b>Figura 1. 68 Pantalla N° 11: Pantalla de Expandir Opciones de Perfil.  <b>(2).</b>Figura 1. 69 Pantalla N° 12: Pantalla de Ver Perfil.  <b>(3).</b>Figura 1. 70 Pantalla N° 13: Pantalla de Gestionar Medio de Pago.</p>	

<b>Historia de Usuario N.º 07</b> <b>“Desactivar Cuenta de Usuario”</b>	
<b>Prioridad:</b>	Media
<b>Estimación:</b>	4 días
<b>Dependiente de:</b>	05
<p><b>Como</b> usuario del sistema <b>quiero</b> poder desactivar mi cuenta de usuario <b>para</b> que mis datos y publicaciones sean eliminadas lógicamente de la plataforma, y no aparezcan en las búsquedas.</p>	
<b>Criterios de Aceptación</b>	
<p><b>Cuando</b> seleccione el botón “Eliminar Cuenta”.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se desactive mi cuenta de usuario.</li> <li>✓ Se desactiven mis publicaciones.</li> <li>✓ Se cancelen los contratos en curso.</li> <li>✓ Se mantenga toda mi actividad, en caso de que vuelva a registrarme con el mismo usuario o email para reactivar mi cuenta.</li> <li>✓ Ni mi usuario ni mis publicaciones aparezcan en la plataforma en búsquedas o enlaces.</li> </ul>
<b>Referencias</b>	
<p><b>Pantallas:</b> (1).Figura 1. 67 Pantalla N° 10: Pantalla de Editar</p>	

### 3.2.2. Módulo de Catálogo y Búsqueda

<b>Historia de Usuario N.º 08</b> <b>“Buscar y Filtrar Contenido”</b>	
<b>Prioridad:</b>	Alta
<b>Estimación:</b>	13 días
<b>Dependiente de:</b>	20
<p><b>Como</b> usuario del sistema <b>quiero</b> buscar y filtrar <b>para</b> poder acceder al contenido que sea de mi interés de manera más rápida.</p>	



Criterios de Aceptación	
<p><b>Cuando</b> seleccione la opción "Buscar" en la página principal.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda ingresar el nombre de la publicación.</li> <li>✓ Me muestre las publicaciones que coincidan con el nombre que he ingresado.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Publicaciones".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Me muestre todas las publicaciones disponibles en la plataforma con un orden desde las más valoradas a las menos (por defecto).</li> </ul>
<p><b>Cuando</b> seleccione la opción "Colecciones".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Me muestre todas las colecciones disponibles en la plataforma con un orden desde las más valoradas a las menos (por defecto).</li> </ul>
<p><b>Cuando</b> seleccione la opción "Publicaciones" + algún filtro.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda seleccionar una o varias opciones de filtrado (Valoración, Likes, Título, Visualizaciones, Fecha Desde - Fecha Hasta).</li> <li>✓ Pueda anular la selección de una o varias opciones de filtrado.</li> <li>✓ Pueda realizar la búsqueda con los filtros aplicados.</li> <li>✓ Me muestre todas las publicaciones disponibles en la plataforma con ese filtro seleccionado, ya sea por valoración, likes, título, fecha de publicación (desde-hasta) o categoría.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Colecciones" + algún filtro.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda seleccionar una o varias opciones de filtrado.</li> <li>✓ Pueda anular la selección de uno o varias opciones de filtrado.</li> <li>✓ Pueda realizar la búsqueda con los filtros aplicados.</li> <li>✓ Me muestre todas las colecciones disponibles en la plataforma con ese filtro seleccionado, ya sea por valoración, likes, título, fecha de publicación (desde-hasta) o categoría.</li> </ul>
Referencias	
<p><b>Pantallas:</b></p> <p>(1).Figura 1. 71 Pantalla N° 14: Pantalla de Biblioteca General de Obras.</p>	

<b>Historia de Usuario N.º 09 "Ver Detalle de Contenido"</b>	
<b>Prioridad:</b>	Media
<b>Estimación:</b>	12 días
<b>Dependiente de:</b>	20
<p><b>Como</b> usuario del sistema <b>quiero</b> acceder al detalle de cierto contenido <b>para</b> poder ver información específica del mismo.</p>	
<b>Criterios de Aceptación</b>	
<p><b>Cuando</b> seleccione una imagen/nombre de una publicación.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Me muestre el detalle de publicación con el nombre de usuario del autor, nombre de la obra, sinopsis de la obra, cantidad de visualizaciones, valoración de estrellas y cantidad de likes, nombre de categoría y etiquetas.</li> </ul>
<p><b>Cuando</b> seleccione una imagen/nombre de una colección.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Me muestre el detalle de la colección con el nombre de usuario del autor, nombre de la colección, sinopsis, categoría y etiquetas y las publicaciones que integran esta colección.</li> </ul>
<b>Referencias</b>	
<p><b>Pantallas:</b></p> <p>(1).Figura 1. 72 Pantalla N° 15: Pantalla de Colecciones.            (2).Figura 1. 73 Pantalla N° 16: Pantalla de Ver Detalle de Colección.            (3).Figura 1. 74 Pantalla N° 17: Pantalla de Ver Detalle de Publicación.</p>	

<b>Historia de Usuario N.º 10 "Ver Favoritos"</b>	
<b>Prioridad:</b>	Baja
<b>Estimación:</b>	10 días
<b>Dependiente de:</b>	12

<p><b>Como</b> usuario <b>quiero</b> poder ver las Publicaciones que he marcado como favorito en un mismo lugar <b>para</b> acceder a su contenido más rápido.</p>	
<p><b>Criterios de Aceptación</b></p>	
<p><b>Cuando</b> ingrese a la sección "Favoritos"</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda visualizar las Publicaciones que haya marcado como mis favoritos.</li> <li>✓ Se muestre la información de las publicaciones marcadas como favorito: portada, nombre de la publicación, descripción, autor, categoría, etiquetas y valoración de estrellas.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b> (1).Figura 1. 75 Pantalla N° 18: Pantalla de Ver Favoritos Publicaciones.</p>	

### 3.2.3. Módulo de Visualización y Seguimiento

<p><b>Historia de Usuario N.º 11</b> <b>"Leer Publicación"</b></p>	
<b>Prioridad:</b>	Alta
<b>Estimación:</b>	19 días
<b>Dependiente de:</b>	09, 20
<p><b>Como</b> usuario del sistema <b>quiero</b> acceder a una publicación <b>para</b> poder leer su contenido.</p>	
<p><b>Criterios de Aceptación</b></p>	
<p><b>Cuando</b> seleccione la publicación que quiero leer (nombre de la publicación o portada).</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Aparezca la información de la publicación: portada, nombre de la publicación, autor, fecha, detalle, categoría, etiquetas, valoración, visualizaciones, likes.</li> <li>✓ Aparezca la opción de donar al autor.</li> <li>✓ Aparezca la opción de leer la publicación.</li> </ul>
<p><b>Cuando</b> seleccione el botón de "Leer".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Aparezca la primera imagen de la obra para comenzar</li> </ul>

	<p>a leerla.</p> <ul style="list-style-type: none"> <li>✓ Pueda ir deslizando la página hacia adelante o volver a la anterior.</li> <li>✓ Aparezca la opción de volver para abandonar la lectura.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Volver".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Aparezca la información nuevamente de la publicación que estaba leyendo: portada, nombre de la publicación, autor, fecha, detalle, categoría, etiquetas, valoración, visualizaciones, likes.</li> <li>✓ Pueda volver a ingresar a la lectura.</li> <li>✓ Pueda donar al autor.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b> (1).Figura 1. 76 Pantalla N° 19: Pantalla de Lector de Cómics.</p>	

<p><b>Historia de Usuario N.º 12</b> <b>"Añadir y Eliminar Contenido de Favoritos"</b></p>	
<b>Prioridad:</b>	Baja
<b>Estimación:</b>	6 días
<b>Dependiente de:</b>	09, 20
<p><b>Como</b> usuario del sistema <b>quiero</b> añadir o eliminar como favoritos una o varias publicaciones o colecciones <b>para</b> tenerlas en mi sección Favoritos y acceder al contenido más rápido para leer o eliminarlo una vez que ya lo haya leído.</p>	
<p><b>Criterios de Aceptación</b></p>	
<p><b>Cuando</b> seleccione el botón "Estrella" de Favoritos dentro del detalle de la publicación.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se marque el botón de Estrella de la publicación.</li> <li>✓ Se haya agregado dicha Publicación en la sección de Favoritos.</li> </ul>
<p><b>Cuando</b> seleccione el botón "Estrella" de favoritos dentro del detalle de la</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se marque el botón de Estrella de la colección.</li> <li>✓ Se haya agregado esa Colección en la sección de Favoritos.</li> </ul>

colección.	
<b>Cuando</b> elimine la selección del botón estrella de favoritos de una publicación.	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Se desmarque el botón de estrella de la publicación.</li> <li>✓ No aparezca más la Publicación en la sección de Favoritos.</li> </ul>
<b>Cuando</b> elimine la selección del botón estrella de favoritos de una colección.	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Se desmarque el botón de estrella de la colección.</li> <li>✓ No aparezca más la Colección en la sección de Favoritos.</li> </ul>
<b>Referencias</b>	
<b>Pantallas:</b> (1).Figura 1. 76 Pantalla N° 19: Pantalla de Lector de Cómics.	

<b>Historia de Usuario N.º 13</b> <b>“Seguir/Dejar de Seguir a Usuario”</b>	
<b>Prioridad:</b>	Baja
<b>Estimación:</b>	6 días
<b>Dependiente de:</b>	20
<b>Como</b> usuario del sistema <b>quiero</b> poder seguir a un usuario <b>para</b> que me notifique cuando haya un nuevo contenido de su autoría o una modificación de este.	
<b>Criterios de Aceptación</b>	
<b>Cuando</b> seleccione el botón “Seguir” en el perfil del autor.	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Cambie el nombre del botón a ‘Siguiendo’</li> <li>✓ Llegue una notificación cuando se haya realizado algún cambio en la información relacionada al autor al que estoy siguiendo.</li> <li>✓ Llegue una notificación de las publicaciones nuevas que estén relacionadas a ese autor al que estoy siguiendo.</li> <li>✓ Llegue una notificación de las publicaciones que hayan tenido un cambio en la información y que estén relacionadas a ese autor al que estoy siguiendo.</li> </ul>

	<ul style="list-style-type: none"> <li>✓ Llegue una notificación de las colecciones nuevas que estén relacionadas a ese autor al que estoy siguiendo.</li> <li>✓ Llegue una notificación de las colecciones que hayan tenido un cambio en la información y que estén relacionadas a ese autor al que estoy siguiendo.</li> </ul>
<p><b>Cuando</b> seleccione el botón "Siguiendo" en el perfil del autor (que estoy siguiendo).</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Cambie el nombre del botón a 'Seguir'.</li> <li>✓ No lleguen más notificaciones de ese autor que estaba siguiendo.</li> </ul>
<b>Referencias</b>	
<p><b>Pantallas:</b> (1).Figura 1. 76 Pantalla N° 19: Pantalla de Lector de Cómic.</p>	

### 3.2.4. Módulo de Reseñas

<b>Historia de Usuario N.º 14 "Calificar Publicación"</b>	
<b>Prioridad:</b>	Media
<b>Estimación:</b>	5 días
<b>Dependiente de:</b>	09, 20
<p><b>Como</b> usuario del sistema <b>quiero</b> poder calificar una publicación <b>para</b> dejar reflejada mi opinión sobre el contenido leído.</p>	
<b>Criterios de Aceptación</b>	
<p><b>Cuando</b> seleccione la opción "Estrella" de una publicación.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Permita seleccionar la cantidad de estrellas que quiero ponerle a la publicación.</li> <li>✓ Se realice la calificación y se marquen la cantidad de estrellas que he seleccionado.</li> <li>✓ Se visualice el campo para escribir un comentario (opcional para las personas suscriptas a la plataforma).</li> </ul>
<b>Referencias</b>	
<p><b>Pantallas:</b></p>	

(1).Figura 1. 77 Pantalla N° 20: Pantalla de Realizar Reseña.

Historia de Usuario N.º 15 "Dar Like a Publicación"	
<b>Prioridad:</b>	Baja
<b>Estimación:</b>	5 días
<b>Dependiente de:</b>	09, 20
<p><b>Como</b> usuario del sistema <b>quiero</b> poder darle like a una publicación <b>para</b> valorar de manera rápida ese contenido.</p>	
Criterios de Aceptación	
<b>Cuando</b> seleccione el botón "Like" de una publicación (ícono en forma de corazón).	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se marque el botón de "Like" de la publicación seleccionada.</li> <li>✓ Se incrementa en uno la cantidad total de likes de dicha publicación.</li> </ul>
Referencias	
<p><b>Pantallas:</b> (1).Figura 1. 77 Pantalla N° 20: Pantalla de Realizar Reseña.</p>	

### 3.2.5. Módulo de Suscripción y Donación

Historia de Usuario N.º 16 "Donar a Autor"	
<b>Prioridad:</b>	Alta
<b>Estimación:</b>	10 días
<b>Dependiente de:</b>	20
<p><b>Como</b> usuario de la plataforma <b>quiero</b> poder donar dinero <b>para</b> apoyar a los autores de mi interés.</p>	
Criterios de Aceptación	

<p><b>Cuando</b> seleccione la opción "Donar" en el perfil del Autor.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda ingresar el monto que quiero donar.</li> <li>✓ Pueda ingresar algún comentario para que luego lo vea el autor.</li> <li>✓ En caso de haber ingresado el monto con un formato incorrecto, me indique cuál sea el error.</li> <li>✓ Pueda confirmar la donación que quiero realizar.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Donar" en el Detalle de la Publicación.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda ingresar el monto que quiero donar.</li> <li>✓ Pueda ingresar algún comentario para que luego lo vea el autor.</li> <li>✓ Quede asentado a qué publicación le realicé la donación.</li> <li>✓ En caso de haber ingresado el monto con un formato incorrecto, me indique cuál sea el error.</li> <li>✓ Pueda confirmar la donación que quiero realizar, clickeando el botón "Confirmar".</li> </ul>
<p><b>Cuando</b> seleccione el botón "Confirmar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Me dirija al check-out de Mercado Pago.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b> (1).Figura 1. 78 Pantalla N° 21: Pantalla de Donar a Autor.</p>	

<p><b>Historia de Usuario N.º 17</b> <b>"Realizar Suscripción"</b></p>	
<p><b>Prioridad:</b></p>	Alta
<p><b>Estimación:</b></p>	5 días
<p><b>Dependiente de:</b></p>	01, 05
<p><b>Como</b> usuario del sistema <b>quiero</b> poder suscribirme a la plataforma <b>para</b> poder tener acceso al contenido premium de todos los autores.</p>	
<p><b>Criterios de Aceptación</b></p>	
<p><b>Cuando</b> seleccione el botón de "Suscribirme".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda ingresar los datos correspondientes de la tarjeta: nombre (como aparece en la tarjeta), número</li> </ul>



	<p>de la tarjeta, fecha de vencimiento y CVC (código de verificación).</p> <ul style="list-style-type: none"> <li>✓ Se visualice el monto total a pagar.</li> <li>✓ Aparezca el botón "Pagar".</li> </ul>
<p><b>Cuando</b> seleccione el botón "Pagar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Me dirija a la sección de pago (check-out de Mercado Pago)</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b></p> <p>(1).Figura 1. 79 Pantalla N° 22: Pantalla de Suscripción a la plataforma.</p>	

<p style="text-align: center;"><b>Historia de Usuario N.º 18</b> <b>"Cancelar Suscripción"</b></p>	
<b>Prioridad:</b>	Media
<b>Estimación:</b>	5 días
<b>Dependiente de:</b>	17, 19
<p><b>Como</b> usuario o sistema <b>quiero</b> cancelar la suscripción <b>para</b> dejar de acceder al contenido premium de los autores o por falta de fondo en la cuenta.</p>	
<p style="text-align: center;"><b>Criterios de Aceptación</b></p>	
<p><b>Cuando</b> seleccione el botón "Cancelar suscripción" desde la sección "Editar cuenta usuario" [usuario].</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Llegue una notificación confirmando la cancelación de mi suscripción a la plataforma.</li> <li>✓ Llegue un email a mi correo con una notificación avisando de la cancelación de la suscripción a la plataforma.</li> <li>✓ Se vuelva a habilitar la opción de "Suscribirse" en caso de querer realizarlo nuevamente.</li> </ul>
<p><b>Cuando</b> se detecten suscripciones con estado "Esperando pago" de cierta cantidad días [sistema].</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se cancele la suscripción del usuario.</li> <li>✓ Llegue al usuario una notificación confirmando la cancelación de su suscripción a la plataforma.</li> <li>✓ Llegue un mail al correo electrónico del usuario avisando de la cancelación de su suscripción a la plataforma.</li> </ul>

	✓ Se vuelva a habilitar la opción de "Suscribirse" para el usuario en caso de querer realizarlo nuevamente.
<b>Referencias</b>	
<b>Pantallas:</b> (1).Figura 1. 80 Pantalla N° 23: Pantalla de Cancelar Suscripción.	

<b>Historia de Usuario N.º 19 "Pagar Suscripción"</b>	
<b>Prioridad:</b>	Alta
<b>Estimación:</b>	10 días
<b>Dependiente de:</b>	17
<p><b>Como</b> usuario o sistema <b>quiero</b> pagar la suscripción <b>para</b> comenzar a acceder al contenido premium de distintos autores (además del gratuito) dentro de la plataforma o en el caso del sistema para que se realice el pago de la suscripción automáticamente.</p>	
<b>Criterios de Aceptación</b>	
<p><b>Cuando</b> seleccione el botón "Confirmar Pago" [usuario].</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se verifique que el pago haya sido realizado.</li> <li>✓ Llegue un email a mi correo confirmando que la suscripción fue realizada con éxito.</li> <li>✓ Llegue un email a mi correo con el comprobante de pago.</li> <li>✓ Pueda acceder a todo el contenido Premium de los autores de la plataforma (Publicaciones y Colecciones).</li> <li>✓ Llegue a la plataforma una notificación de pago con el monto abonado.</li> </ul>
<p><b>Cuando</b> se detecte que ha pasado un mes desde la fecha de suscripción [sistema].</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ La suscripción pase de estado "Activo" a "Esperando pago".</li> </ul>
<p><b>Cuando</b> se detecten suscripciones con estado "Esperando</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se envíe el token de usuario con los datos de cobro para la realización del pago de la suscripción</li> </ul>

<p>pago" [sistema].</p>	<p>automáticamente.</p> <ul style="list-style-type: none"> <li>✓ Se verifique que el pago haya sido realizado</li> <li>✓ Llegue un email a mi correo confirmando que la suscripción fue realizada con éxito.</li> <li>✓ Me llegue un email a mi correo con el comprobante de pago.</li> <li>✓ Pueda acceder a todo el contenido Premium de los autores de la plataforma (Publicaciones y Colecciones).</li> <li>✓ Me llegue a la plataforma una notificación de pago con el monto abonado.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b> (1).Figura 1. 81 Pantalla N° 24: Pantalla de Introducir Pago.</p>	

### 3.2.6. Módulo de Publicación de Obras

<p><b>Historia de Usuario N.º 20 "Publicar Obra"</b></p>	
<p><b>Prioridad:</b></p>	<p>Alta</p>
<p><b>Estimación:</b></p>	<p>10 días</p>
<p><b>Dependiente de:</b></p>	<p>34</p>
<p><b>Como</b> autor <b>quiero</b> poder publicar mis obras <b>para</b> que los lectores puedan acceder al contenido en la plataforma.</p>	
<p><b>Criterios de Aceptación</b></p>	
<p><b>Cuando</b> ingrese a la sección "Publicar Obra/Modificar Obra" desde mi cuenta.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda ingresar los siguientes datos: nombre de la obra y descripción.</li> <li>✓ Pueda seleccionar del menú la "Categoría" de la obra.</li> <li>✓ Pueda ingresar Etiquetas, para facilitar la búsqueda.</li> <li>✓ Aparezcan los botones "Publicar", "Modificar" y "Guardar como borrador".</li> </ul>
<p><b>Cuando</b> seleccione el botón "Publicar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ La obra quede guardada con estado "Esperando aprobación".</li> </ul>

<p><b>Cuando</b> seleccione el botón "Guardar como Borrador".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ La obra quede guardada con estado "Borrador".</li> <li>✓ No aparezca la obra en búsquedas ni se pueda acceder a su detalle.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b></p> <p>(1).Figura 1. 82 Pantalla N° 25: Pantalla de Solicitar Publicar Obra.</p> <p>(2).Figura 1. 83 Pantalla N° 26: Pantalla de Publicación Espera Aprobación.</p>	

<p><b>Historia de Usuario N.º 21</b> <b>"Modificar Publicación"</b></p>	
<p><b>Prioridad:</b></p>	<p>Media</p>
<p><b>Estimación:</b></p>	<p>6 días</p>
<p><b>Dependiente de:</b></p>	<p>20</p>
<p><b>Como</b> autor <b>quiero</b> poder realizar una modificación en una publicación <b>para</b> actualizar información o algún tipo de cambio.</p>	
<p><b>Criterios de Aceptación</b></p>	
<p><b>Cuando</b> ingrese a la sección "Publicar Obra/Modificar Obra" desde mi cuenta.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda visualizar de la publicación los siguientes datos: nombre de la publicación, descripción, imágenes, el menú de selección de categorías, y las etiquetas.</li> <li>✓ Aparezca el botón "Modificar".</li> </ul>
<p><b>Cuando</b> presione el botón "Modificar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda modificar alguno o todos los datos visualizados anteriormente, tales como: nombre de la publicación, descripción, imágenes, categoría, y etiquetas.</li> <li>✓ Llegue una notificación si los cambios que quise realizar fueron aceptados o no.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b></p> <p>(1).Figura 1. 84 Pantalla N° 27: Pantalla de Editar Publicación.</p>	

<b>Historia de Usuario N.º 22 "Gestionar Colecciones"</b>	
<b>Prioridad:</b>	Media
<b>Estimación:</b>	10 días
<b>Dependiente de:</b>	20
<p><b>Como autor quiero</b> poder gestionar colecciones que contengan varias publicaciones <b>para</b> que los lectores puedan acceder al contenido de manera más ordenada según el tema de las obras al que pertenezcan.</p>	
<b>Criterios de Aceptación</b>	
<p><b>Cuando</b> ingrese a la sección "Gestionar colecciones" desde mi cuenta.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda ver "Mis Colecciones" con los siguientes datos: portada, nombre de colección, descripción, autor, categoría y etiquetas.</li> <li>✓ Pueda modificar o eliminar una colección de la Lista de Colecciones.</li> <li>✓ Pueda agregar una nueva colección a mi Lista.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Modificar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda cambiar la foto de portada, nombre de la colección, descripción, categoría y etiquetas.</li> <li>✓ Pueda agregar o quitar publicaciones de mi Colección.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Eliminar" colección de mi Lista de Colecciones.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda eliminar la colección completa de mi Lista de Colecciones.</li> <li>✓ No aparezca esa colección en mi Lista de Colecciones ni en el sistema.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Agregar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda crear una nueva Colección.</li> <li>✓ Pueda ingresar los datos de la nueva Colección: nombre de la colección, descripción, portada y etiquetas.</li> <li>✓ Pueda seleccionar del menú la "Categoría".</li> <li>✓ Aparezca la opción para Buscar las obras publicadas de mi autoría para agregarlas a mi colección.</li> <li>✓ Pueda eliminar la/las publicaciones que he agregado a mi colección.</li> </ul>

<p><b>Cuando</b> seleccione la opción "Crear" publicación de la lista.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se guarden los cambios realizados.</li> <li>✓ Aparezca la nueva colección en mi Lista de Colecciones.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b></p> <p>(1).Figura 1. 85 Pantalla N° 28: Pantalla de Gestionar Colecciones.          (2).Figura 1. 86 Pantalla N° 29: Pantalla de Crear Colecciones.          (3).Figura 1. 87 Pantalla N° 30: Pantalla de Editar Colecciones.</p>	

<p><b>Historia de Usuario N.º 23</b>  <b>"Aceptar/Rechazar Publicación"</b></p>	
<p><b>Prioridad:</b></p>	<p>Alta</p>
<p><b>Estimación:</b></p>	<p>7 días</p>
<p><b>Dependiente de:</b></p>	<p>20</p>
<p><b>Como</b> administrador <b>quiero</b> aceptar o rechazar una publicación que se quiere subir a la plataforma <b>para</b> analizar que el contenido sea apto para la misma.</p>	
<p><b>Criterios de Aceptación</b></p>	
<p><b>Cuando</b> ingrese a la sección "Gestión de contenido" desde mi cuenta.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se visualice un listado con las Publicaciones que estén pendientes de aprobación.</li> <li>✓ Pueda visualizar por cada Publicación de la lista, los siguientes datos: nombre de la publicación, portada, autor, fecha, descripción, categoría, etiquetas y valoración (estrellas).</li> <li>✓ Pueda rechazar la publicación o aceptarla.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Rechazar publicación".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se abra una ventana donde se pueda dejar una descripción del motivo por el cual ha sido rechazado.</li> <li>✓ Se envíe un email al correo al autor para informarle que su publicación fue rechazada, y los motivos.</li> <li>✓ El autor pueda visualizar el estado de los cambios que quiere realizar (en este caso, Rechazado).</li> </ul>

<p><b>Cuando</b> seleccione la opción "Aceptar Publicación".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ La publicación se realice correctamente con los cambios pedidos por el autor.</li> <li>✓ La publicación aparezca en el catálogo.</li> <li>✓ El autor pueda visualizar el estado de los cambios que realizó como "Publicado".</li> <li>✓ Se envíe un email al correo al autor para informarle que su publicación fue aceptada y que ya está publicada en la plataforma.</li> <li>✓ En caso de ser su primer cómic publicado en la plataforma de Omics le aparezca un mensaje de felicitaciones.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b></p> <p>(1).Figura 1. 88 Pantalla N° 31: Pantalla de Aceptar/Rechazar Publicaciones.</p> <p>(2).Figura 1. 89 Pantalla N° 32: Pantalla de Ver Estadísticas de Publicación.</p> <p>(3).Figura 1. 90 Pantalla N° 33: Pantalla de primera publicación aprobada.</p>	

<p><b>Historia de Usuario N.º 24</b> <b>"Ver Estadísticas de Publicaciones"</b></p>	
<p><b>Prioridad:</b></p>	<p>Baja</p>
<p><b>Estimación:</b></p>	<p>7 días</p>
<p><b>Dependiente de:</b></p>	<p>16, 20, 30</p>
<p><b>Como</b> autor <b>quiero</b> ver las estadísticas de las obras publicadas <b>para</b> analizar datos importantes de la misma.</p>	
<p><b>Criterios de Aceptación</b></p>	
<p><b>Cuando</b> seleccione la opción "Ver Estadísticas de Publicación".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se visualicen todas mis publicaciones con la información importante de cada una.</li> <li>✓ Pueda ver el nombre de cada publicación, descripción, mi nombre, portada con una imagen, categoría, etiquetas, visualizaciones que tiene actualmente la publicación, cantidad de likes, y valoración (estrellas).</li> <li>✓ Aparezca la opción de poder ver los comentarios.</li> </ul>

	<ul style="list-style-type: none"> <li>✓ Aparezca la cantidad de donaciones y el total de dinero recibido.</li> <li>✓ En caso de superar cierta cantidad de visualizaciones, se active el botón de "Solicitar contrato".</li> <li>✓ En las publicaciones que tienen contrato se muestren los resúmenes generados anteriormente (sea manual o automáticamente) pendientes de cobro.</li> <li>✓ Se visualice el total que pueda cobrar de acuerdo a mis publicaciones con contratos y con resúmenes pendientes de cobro.</li> </ul>
<p><b>Cuando</b> seleccione la opción de "Ver Comentarios".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se puedan ver todos los comentarios que han realizado los usuarios sobre cada una de mis publicaciones.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b>  <b>(1).</b>Figura 1. 89 Pantalla N° 32: Pantalla de Ver Estadísticas de Publicación.</p>	

### 3.2.7. Módulo de Notificaciones

<p><b>Historia de Usuario N.º 25</b>  <b>"Notificar Actividad de Autores y Contenido"</b></p>	
<b>Prioridad:</b>	Media
<b>Estimación:</b>	10 días
<b>Dependiente de:</b>	05, 09, 13, 20
<p><b>Como</b> sistema <b>quiero</b> notificar a los usuarios sobre la actividad de los autores y su contenido <b>para</b> que les llegue un aviso cuando haya nuevo contenido o un cambio en el mismo dentro de la plataforma.</p>	
<p><b>Criterios de Aceptación</b></p>	
<p><b>Cuando</b> se realice un cambio en el contenido o haya alguna publicación nueva de un autor.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se manden notificaciones al usuario cuando haya contenido nuevo de autores favoritos.</li> <li>✓ Se manden notificaciones al usuario cuando haya contenido modificado de autores favoritos.</li> </ul>



<b>Referencias</b>
<b>Pantallas:</b> (1).Figura 1. 91 Pantalla N° 34: Pantalla de Notificaciones.

<b>Historia de Usuario N.º 26 "Notificar Resumen de Pago"</b>	
<b>Prioridad:</b>	Media
<b>Estimación:</b>	10 días
<b>Dependiente de:</b>	30
<p><b>Como</b> sistema <b>quiero</b> notificar a los autores que posean contrato con la plataforma <b>para</b> que les llegue un aviso cuando se encuentre generado su resumen de pago mensual.</p>	
<b>Criterios de Aceptación</b>	
<p><b>Cuando</b> termine el período correspondiente a un mes y se llegue a la fecha establecida.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se le mande una notificación al autor sobre el resumen de pago mensual que está disponible.</li> <li>✓ El autor pueda ver sus estadísticas a partir de la notificación.</li> <li>✓ Se le mande un mail al autor con el resumen de pago mensual que está disponible.</li> </ul>
<b>Referencias</b>	
<b>Pantallas:</b> (1).Figura 1. 91 Pantalla N° 34: Pantalla de Notificaciones.	

### 3.2.8. Módulo de Contrato

<b>Historia de Usuario N.º 27 "Requerir Contrato"</b>	
<b>Prioridad:</b>	Alta
<b>Estimación:</b>	12 días
<b>Dependiente de:</b>	20

**Como** autor **quiero** poder requerir el contrato en base a la cantidad de visualizaciones que ha tenido el contenido de mi autoría **para** acceder a un beneficio económico mayor a las donaciones.

**Criterios de Aceptación**

<b>Cuando</b> ingrese a la sección "Ver Estadísticas de Publicación" dentro de mi cuenta.	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Pueda visualizar la lista de todas mis publicaciones.</li> <li>✓ Por cada una de las Publicaciones, se pueda ver los siguientes datos: nombre de la publicación, descripción, portada, autor, categoría, etiquetas, cantidad de visualizaciones, cantidad de likes y cantidad de estrellas.</li> <li>✓ Me aparezca la opción de poder ver los comentarios</li> <li>✓ Tenga la opción de solicitar contrato en cada publicación que haya alcanzado cierta cantidad de visualizaciones.</li> </ul>
<b>Cuando</b> seleccione la opción de "Ver Comentarios".	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Se puedan ver todos los comentarios que han realizado los usuarios sobre cada una de mis publicaciones.</li> </ul>
<b>Cuando</b> seleccione la opción "Solicitar Contrato".	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ El contrato quede pendiente de aprobación para que un administrador me responda mi solicitud.</li> </ul>

**Referencias**

**Pantallas:**

- (1).Figura 1. 92 Pantalla N° 35: Pantalla de Publicación Sin Contrato.
- (2).Figura 1. 93 Pantalla N° 36: Pantalla de Mis Comics.
- (3).Figura 1. 94 Pantalla N° 37: Pantalla de Comentarios.

**Historia de Usuario N.º 28  
"Rescindir Contrato"**

<b>Prioridad:</b>	Media
<b>Estimación:</b>	6 días
<b>Dependiente de:</b>	20, 27

<p><b>Como autor quiero</b> poder rescindir mi contrato <b>para</b> dejar sin efecto el mismo y así desligarme de los derechos sobre la publicación, dejando de percibir el beneficio económico.</p>	
<p><b>Criterios de Aceptación</b></p>	
<p><b>Cuando</b> ingrese a la sección "Ver estadísticas de publicación".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda visualizar la lista de todas mis publicaciones.</li> <li>✓ Por cada una de las Publicaciones, se pueda ver los siguientes datos: nombre de la publicación, descripción, portada, autor, categoría, etiquetas, cantidad de visualizaciones, cantidad de likes y cantidad de estrellas.</li> <li>✓ Aparezca la opción de poder ver los comentarios</li> <li>✓ Tenga la opción de rescindir contrato en cada publicación que tenga contrato.</li> </ul>
<p><b>Cuando</b> seleccione la opción de "Ver comentarios".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se puedan ver todos los comentarios que han realizado los usuarios sobre cada una de mis publicaciones.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Rescindir Contrato".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se cancele el contrato.</li> <li>✓ Se active nuevamente la opción para solicitar el contrato.</li> <li>✓ Deje de tener derecho sobre los cobros por la cantidad de visualizaciones de la publicación.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b></p> <p>(1).Figura 1. 95 Pantalla N° 38: Pantalla de Publicación Con Contrato.</p>	

<p><b>Historia de Usuario N.º 29</b> <b>"Cobrar por Resumen de Publicación"</b></p>	
<b>Prioridad:</b>	Alta
<b>Estimación:</b>	6 días
<b>Dependiente de:</b>	20, 30

<p><b>Como</b> autor <b>quiero</b> cobrar por resumen de publicación <b>para</b> recibir el dinero correspondiente de acuerdo con las visualizaciones que hayan tenido las publicaciones con contrato vigente.</p>	
<p><b>Criterios de Aceptación</b></p>	
<p><b>Cuando</b> seleccione la opción "Cobrar" en la sección de "Ver estadísticas de publicación".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se verifique mi token de Mercado Pago en caso de que exista.</li> <li>✓ Se visualice el monto total a cobrar.</li> <li>✓ Aparezca la opción para cobrar.</li> <li>✓ Se realice un pago directamente a mi cuenta de Mercado Pago (de acuerdo con el token registrado).</li> </ul>
<p><b>Cuando</b> seleccione la opción "Cobrar" y no tenga registrado el medio de cobro (Mercado Pago Token).</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se dirija a Editar perfil de usuario para dar de alta un token único generado por Mercado Pago (de forma externa).</li> <li>✓ Pueda ingresar los datos correspondientes de la cuenta de Mercado Pago: usuario, DNI y CVU (clave virtual uniforme) en Mercado Pago (externo) para que genere mi token único de cobro y registrarlo en Editar Perfil de usuario.</li> <li>✓ Una vez registre un nuevo token se dirija a "Ver estadísticas de publicación" para continuar con el Cobro.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Cobrar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se dirija a la sección de cobro (check-out de Mercado Pago).</li> <li>✓ Se realice el pago visualizado en el "Total" automáticamente.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b></p> <p>(1).Figura 1. 96 Pantalla N° 39: Pantalla de Billetera Autor.</p> <p>(2).Figura 1. 97 Pantalla N° 40: Pantalla de Publicación Ver Resúmenes Generados.</p>	

<b>Historia de Usuario N.º 30</b> <b>“Generar Resumen de Publicación”</b>	
<b>Prioridad:</b>	Media
<b>Estimación:</b>	7 días
<b>Dependiente de:</b>	20
<p><b>Como</b> autor o sistema <b>quiero</b> generar un resumen <b>para</b> que se puedan visualizar los detalles de cada publicación con el correspondiente monto de cantidad de visualizaciones únicas y visualizaciones totales.</p>	
<b>Criterios de Aceptación</b>	
<p><b>Cuando</b> seleccione la opción “Actualizar resúmenes” [autor].</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se generen los resúmenes correspondientes a mis publicaciones con contrato desde el último resumen generado hasta la actualidad (fecha actual), que abarquen un rango mensual.</li> <li>✓ Se genere un resumen por publicación.</li> <li>✓ Los resúmenes tengan los siguientes datos sujetos al rango de fechas que representa el resumen: pago correspondiente al autor, cantidad de visualizaciones totales, cantidad de visualizaciones únicas, calificación de la publicación, likes recibidos, usuarios que la agregaron a favoritos, fecha desde, fecha hasta.</li> <li>✓ Se muestre todos los resúmenes pendientes de cobro con los detalles correspondientes (fecha desde-fecha hasta, cantidad de visualizaciones generales y únicas, monto de las donaciones, monto de las suscripciones y subtotal).</li> <li>✓ Se muestre el monto total de los resúmenes pendientes a cobrar.</li> </ul>
<p><b>Cuando</b> el autor no seleccione la opción de “Actualizar resúmenes” durante un mes, el sistema lo actualiza automáticamente [sistema].</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se muestre al autor todos los resúmenes pendientes de cobro con los detalles correspondientes (fecha desde-fecha hasta, cantidad de visualizaciones generales y únicas, monto de las donaciones, monto de las suscripciones y subtotal).</li> <li>✓ Se muestre al autor el monto total de los resúmenes pendientes a cobrar.</li> </ul>

Referencias
<p><b>Pantallas:</b></p> <p>(1).Figura 1. 97 Pantalla N° 40: Pantalla de Publicación Ver Resúmenes Generados.</p>

Historia de Usuario N.º 31 "Aceptar/Rechazar Contrato"	
<b>Prioridad:</b>	Media
<b>Estimación:</b>	5 días
<b>Dependiente de:</b>	20
<p><b>Como</b> administrador <b>quiero</b> poder aceptar o rechazar el contrato de un autor <b>para</b> que comience a cobrar por la cantidad de visualizaciones que genere el contenido del autor a partir de la fecha de creación del contrato, si cumple cierta cantidad de visualizaciones, y si no poder rechazar el contrato.</p>	
Criterios de Aceptación	
<p><b>Cuando</b> ingrese a la sección "Gestión de Contenido".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Aparezcan todos los contratos pendientes de aceptación.</li> <li>✓ Pueda ver la información importante de cada publicación: portada, nombre de la publicación, descripción, autor, categoría, etiquetas, valoración y visualizaciones.</li> <li>✓ Se pueda actualizar en ese momento las visualizaciones para tener la información precisa.</li> <li>✓ Pueda aceptarlo si cumple por lo menos con la cantidad mínima de visualizaciones requeridas de sus publicaciones.</li> <li>✓ Pueda rechazarlo por algún problema o motivo específico.</li> </ul>
<p><b>Cuando</b> seleccione la opción de "Actualizar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Aparezca la cantidad de visualizaciones que tiene hasta ese momento la publicación.</li> </ul>
<p><b>Cuando</b> seleccione la</p>	<p><b>Espero que:</b></p>

opción de "Aceptar Contrato".	<ul style="list-style-type: none"> <li>✓ Se apruebe el contrato y comience a cobrar por la actividad de la publicación.</li> <li>✓ Le llegue una notificación al autor con la confirmación del contrato.</li> </ul>
<b>Cuando</b> seleccione la opción de "Rechazar Contrato".	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Pueda ingresar los motivos del rechazo del contrato.</li> <li>✓ Pueda confirmar la decisión del rechazo del contrato.</li> </ul>
<b>Cuando</b> seleccione la opción "Confirmar".	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Le llegue la notificación al autor del rechazo del contrato con los motivos adjuntos.</li> </ul>
<b>Referencias</b>	
<b>Pantallas:</b>	
(1).Figura 1. 98 Pantalla N° 41: Pantalla de Aprobar/Rechazar Contrato.	

### 3.2.9. Módulo de Reportes

<b>Historia de Usuario N.º 32</b> <b>"Generar Reportes de Usuarios"</b>	
<b>Prioridad:</b>	Alta
<b>Estimación:</b>	6 días
<b>Dependiente de:</b>	01, 20, 30
<b>Como</b> administrador <b>quiero</b> poder generar, y visualizar reportes <b>para</b> tener información específica de aspectos relacionados a los usuarios.	
<b>Criterios de Aceptación</b>	
<b>Cuando</b> ingrese a la sección "Reportes".	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Pueda seleccionar la sección "Usuarios".</li> <li>✓ Pueda ingresar la "fecha desde", y "fecha hasta" del reporte a generar.</li> </ul>
<b>Cuando</b> seleccione la opción "Usuarios" e ingrese la "fecha desde" y "fecha hasta".	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Pueda visualizar información importante de los usuarios: Cantidad de usuarios por edad, cantidad de usuarios por género, cantidad total de usuarios, cantidad de lectores y autores, cantidad de</li> </ul>

	<p>seguidores, cantidad nuevos usuarios respecto de la fecha ingresada, cantidad de usuarios que dejaron la plataforma en el rango ingresado, "fecha desde" ingresada, "fecha hasta" ingresada, y "fecha de generación" del reporte.</p> <ul style="list-style-type: none"> <li>✓ Los reportes se generen dinámicamente de acuerdo con el rango de fechas ingresado (se ve online) y se puedan exportar en formato Excel o PDF.</li> </ul>
<b>Referencias</b>	
<p><b>Pantallas:</b></p> <p>(1).Figura 1. 99 Pantalla N° 42: Pantalla de Reportes (a).</p> <p>(2).Figura 1. 100 Pantalla N° 43: Pantalla de Reportes (b).</p>	

<b>Historia de Usuario N.º 33</b> <b>"Generar Reportes de Publicaciones"</b>	
<b>Prioridad:</b>	Alta
<b>Estimación:</b>	7 días
<b>Dependiente de:</b>	01, 20, 30
<p><b>Como</b> administrador <b>quiero</b> poder generar, y visualizar reportes <b>para</b> tener información específica de aspectos relacionados a las Publicaciones.</p>	
<b>Criterios de Aceptación</b>	
<p><b>Cuando</b> ingrese a la sección "Reportes".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda seleccionar la sección "Publicaciones".</li> <li>✓ Pueda ingresar la "fecha desde", y "fecha hasta" del reporte a generar.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Publicaciones" e ingrese la "fecha desde" y "fecha hasta".</p>	<ul style="list-style-type: none"> <li>✓ Se pueda ver información con respecto a publicaciones: Publicaciones totales, colecciones totales, cantidad de publicaciones por categoría, cantidad de visualizaciones, cantidad de likes, nuevas publicaciones hasta la fecha ingresada, cantidad de contratos, cantidad de nuevos contratos en el período seleccionado, "fecha desde" ingresada, "fecha hasta" ingresada, y "fecha de generación" del reporte.</li> <li>✓ Los reportes se generen dinámicamente de acuerdo</li> </ul>



	con el rango de fechas ingresado (se ve online) y se puedan exportar en formato Excel o PDF.
<b>Referencias</b>	
<b>Pantallas:</b> <b>(1).</b> Figura 1. 99 Pantalla N° 42: Pantalla de Reportes (a). <b>(2).</b> Figura 1. 100 Pantalla N° 43: Pantalla de Reportes (b).	

<b>Historia de Usuario N.º 34</b> <b>“Generar Reportes de Ingresos y Egresos”</b>	
<b>Prioridad:</b>	Alta
<b>Estimación:</b>	7 días
<b>Dependiente de:</b>	01, 20, 30
<p><b>Como</b> administrador <b>quiero</b> poder generar, y visualizar reportes <b>para</b> tener información específica de aspectos relacionados con los ingresos y egresos económicos de la plataforma.</p>	
<b>Criterios de Aceptación</b>	
<b>Cuando</b> ingrese a la sección “Reportes”.	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Pueda seleccionar la sección “Ingresos y egresos”.</li> <li>✓ Pueda ingresar la “fecha desde” y “fecha hasta” del reporte a generar.</li> </ul>
<b>Cuando</b> seleccione la opción “Ingresos y egresos” e ingrese la “fecha desde” y “fecha hasta”.	<ul style="list-style-type: none"> <li>✓ Se pueda ver información con respecto a los ingresos y egresos: Monto pagado a los autores, montos de suscripción, montos de donación, monto total en ese rango ingresado, “fecha desde” ingresada, “fecha hasta” ingresada, y “fecha de generación” del reporte.</li> <li>✓ Los reportes se generen dinámicamente de acuerdo con el rango de fechas ingresado (se ve online) y se puedan exportar en formato Excel o PDF.</li> </ul>
<b>Referencias</b>	
<b>Pantallas:</b> (1).Figura 1. 99 Pantalla N° 42: Pantalla de Reportes (a). (2).Figura 1. 100 Pantalla N° 43: Pantalla de Reportes (b).	

### 3.2.10. Módulo de Configuración

<b>Historia de Usuario N.º 35</b> <b>“Configurar Reglas de Negocio”</b>	
<b>Prioridad:</b>	Alta
<b>Estimación:</b>	15 días
<b>Dependiente de:</b>	---

**Como** administrador **quiero** poder configurar reglas de negocio **para** tenerlas parametrizadas y que se puedan ir actualizando dinámicamente, sin tener que reiniciar el sistema.

**Criterios de Aceptación**

<b>Cuando</b> ingrese a la sección de "Administración".	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Muestre un listado de todas las reglas de negocio.</li> <li>✓ Por cada campo de reglas, pueda editarlas.</li> <li>✓ No se puedan añadir ni eliminar reglas porque podrían interferir con la lógica del negocio.</li> <li>✓ Tener la opción de guardar para confirmar los cambios realizados.</li> </ul>
---	---

<b>Cuando</b> seleccione el botón "Guardar".	<b>Espero que:</b> <ul style="list-style-type: none"> <li>✓ Se validan los valores ingresados en los campos para impedir el ingreso de campos inválidos que dejen al sistema inutilizable o con valores erróneos de reglas de negocio.</li> <li>✓ Se guarden los cambios realizados.</li> <li>✓ Me muestre los campos con los cambios actualizados.</li> <li>✓ Se refleje el cambio en el sistema.</li> </ul>
--	---

**Referencias**

**Pantallas:**  
 (1).Figura 1. 101 Pantalla N° 44: Gestionar Reglas de Negocios.

**Historia de Usuario N.º 36  
 "Gestionar Categorías"**

<b>Prioridad:</b>	Alta
<b>Estimación:</b>	11 días
<b>Dependiente de:</b>	02

**Como** administrador **quiero** poder gestionar las categorías de las obras **para** poder darle un orden a las publicaciones y así hacer más sencilla la búsqueda de estas.

**Criterios de Aceptación**

<p><b>Cuando</b> seleccione la opción "Gestionar categorías".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se abra un formulario con los valores de configuración de categorías cargados en los campos, y que éstos puedan editarse.</li> <li>✓ Se puedan eliminar categorías en el caso de que no estén siendo utilizadas por ninguna publicación o colección.</li> <li>✓ Se pueda añadir el nombre de una nueva categoría.</li> <li>✓ Se puedan guardar los cambios realizados.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Editar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se pueda cambiar el nombre de las categorías.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Añadir".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Permite añadir el nombre de una nueva categoría.</li> </ul>
<p><b>Cuando</b> selecciono la opción "Eliminar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se valide que la categoría no se esté utilizando por ninguna publicación ni colección.</li> <li>✓ Se elimine la categoría.</li> </ul>
<p><b>Cuando</b> selecciono la opción "Guardar".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se guarde la edición de categorías.</li> <li>✓ Me muestre un mensaje de éxito.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b></p> <p>(1).Figura 1. 102 Pantalla N° 45: Pantalla de Administrador de Gestionar Categorías.</p>	

<p><b>Historia de Usuario N.º 37</b> <b>"ABM Roles"</b></p>	
<p><b>Prioridad:</b></p>	Alta
<p><b>Estimación:</b></p>	08 días
<p><b>Dependiente de:</b></p>	02

<p><b>Como</b> administrador <b>quiero</b> poder gestionar los Roles <b>para</b> poder asignar, modificar, y eliminar los permisos a los usuarios del sistema.</p>	
Criterios de Aceptación	
<p><b>Cuando</b> seleccione la opción de "Gestionar usuarios y roles".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Me muestre la opción de: "Roles" y "Usuarios" para indicar la sección que quiero ver.</li> </ul>
<p><b>Cuando</b> seleccione la opción de "Roles".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Me muestre una tabla con los datos de los roles existentes: id del rol, nombre del rol, cantidad de permisos que tiene asignado ese rol, fecha de creación y fecha de actualización del rol.</li> <li>✓ Tenga la opción de "Agregar Nuevo Rol".</li> </ul>
<p><b>Cuando</b> seleccione la opción "Agregar Nuevo Rol".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Pueda ingresar el nombre del rol que se quiere crear, aparezca la lista con los permisos disponibles y se pueda asignar los correspondientes al rol que se está estableciendo.</li> <li>✓ Esté la opción de seleccionar un rol por defecto.</li> <li>✓ Me salga la opción "Guardar" para guardar el nuevo rol creado.</li> </ul>
<p><b>Cuando</b> seleccione un rol existente.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Me permita editar los siguientes datos: "nombre del rol", "Permisos Asignados", "Lista de Permisos Disponibles" y el rol por defecto.</li> <li>✓ Aparezca la opción de "Eliminar" el rol seleccionado.</li> <li>✓ Se visualice la opción "Guardar" para guardar los cambios realizados.</li> </ul>
<p><b>Cuando</b> seleccione la opción "Eliminar Rol".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se muestre un mensaje de confirmación para eliminar definitivamente ese rol.</li> <li>✓ En caso de que el rol no tenga ningún usuario asignado o cuando no es un rol por defecto me permita eliminarlo.</li> <li>✓ En el caso de que el rol tiene un usuario asignado, o es un Rol por defecto, entonces dicho rol no se pueda eliminar.</li> <li>✓ Los Permisos que tenía asignados el rol eliminado, sean desasignados del mismo.</li> </ul>

<p><b>Cuando</b> seleccione la opción de "Usuarios".</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Me muestre una tabla con todos los usuarios disponibles en la plataforma y la siguiente información: id del usuario, nombre del usuario, rol que tiene asignado ese usuario, fecha en que el usuario se registró en la plataforma y si el usuario se ha validado o no.</li> <li>✓ Pueda filtrar los usuarios ingresando el nombre del usuario y/o seleccionar un rol.</li> <li>✓ Pueda seleccionar la opción de "Filtrar" para que realice esa búsqueda contemplando el filtro utilizado.</li> <li>✓ Pueda seleccionar la opción "Limpiar" para borrar los datos del filtro que ingresé.</li> <li>✓ Pueda ingresar la cantidad de ítems por página que quiero que se muestren.</li> </ul>
<p><b>Cuando</b> seleccione un usuario en particular.</p>	<p><b>Espero que:</b></p> <ul style="list-style-type: none"> <li>✓ Se visualicen los datos del usuario: nombre, email, fecha de creación, validación de usuario y rol asignado,</li> <li>✓ Se muestre un dropdown (lista desplegable) para poder seleccionar el rol y asignárselo al usuario.</li> <li>✓ Esté habilitada la opción de "Eliminar", "Asignar" y "Guardar" los cambios realizados para el usuario.</li> </ul>
<p><b>Referencias</b></p>	
<p><b>Pantallas:</b></p> <ol style="list-style-type: none"> <li>(1).Figura 1. 103 Pantalla N° 46: Gestionar Roles y Visualizar Tabla de Roles Disponibles.</li> <li>(2).Figura 1. 104 Pantalla N° 47: Crear Nuevo Rol.</li> <li>(3).Figura 1. 105 Pantalla N° 48: Editar Rol Existente.</li> <li>(4).Figura 1. 106 Pantalla N° 49: Gestionar Usuarios y sus Roles Asignados.</li> <li>(5).Figura 1. 107 Pantalla N° 50: Cambiar Rol a un Usuario Existente.</li> </ol>	

## 4. Pantallas

### 4.1 Mockups

Para efectuar un mejor análisis del flujo e interacción del usuario con la aplicación se realizan *mockups* (mediante *wireframes*) preliminares a las pantallas.

Estos únicamente representan la existencia de algunos componentes vitales y el flujo determinado por la interacción. Sin embargo, no son representativos del diseño, estilo o interfaz de la aplicación final. Son meras estructuras que facilitan el entendimiento global de la aplicación al nivel de interfaz gráfica. Son prototipos de pantallas que corresponden a UX (*User Experience*). (**VER ANEXO N°4 MOCKUPS.**)

### 4.2 Módulos

#### 4.2.1. Módulo de Registro y Actividad de Usuario

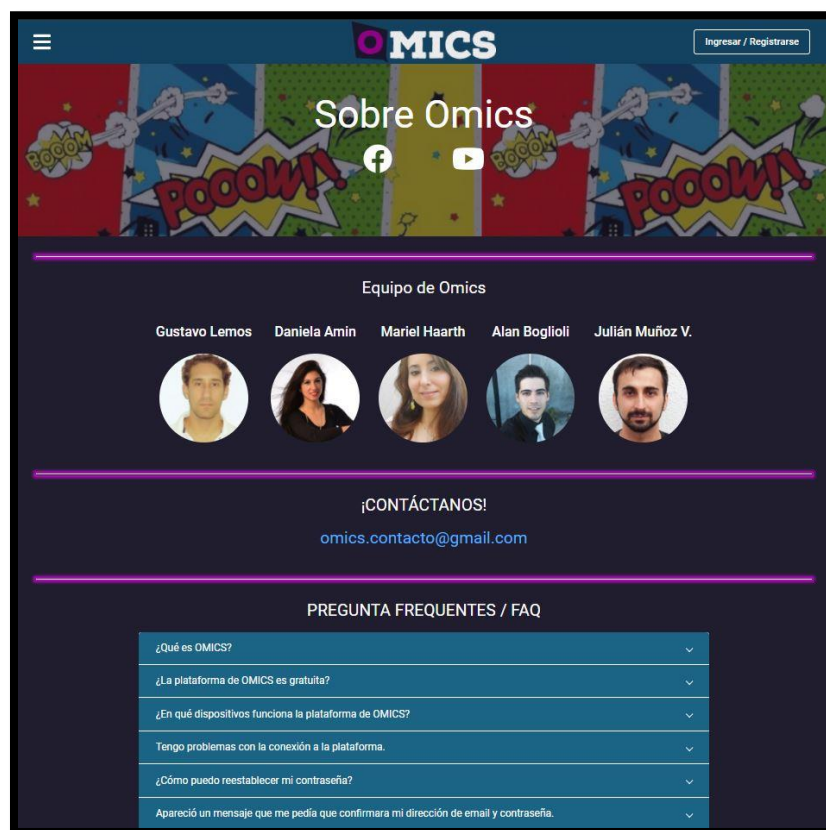


Figura 1. 58 Pantalla N° 1: Pantalla de Bienvenida a Omics.

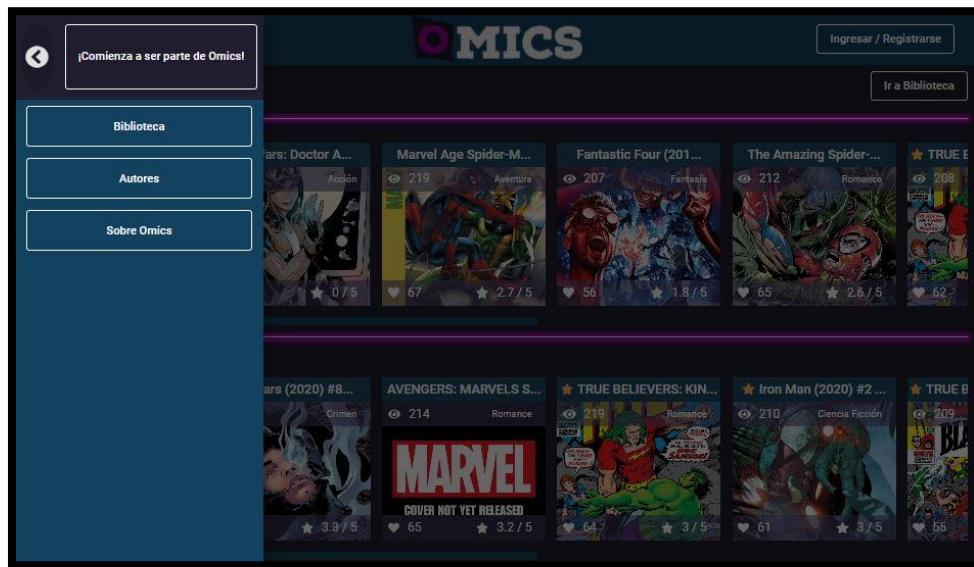


Figura 1. 59 Pantalla N° 2: Pantalla de Inicio Con Barra Lateral. Sin loguearse.

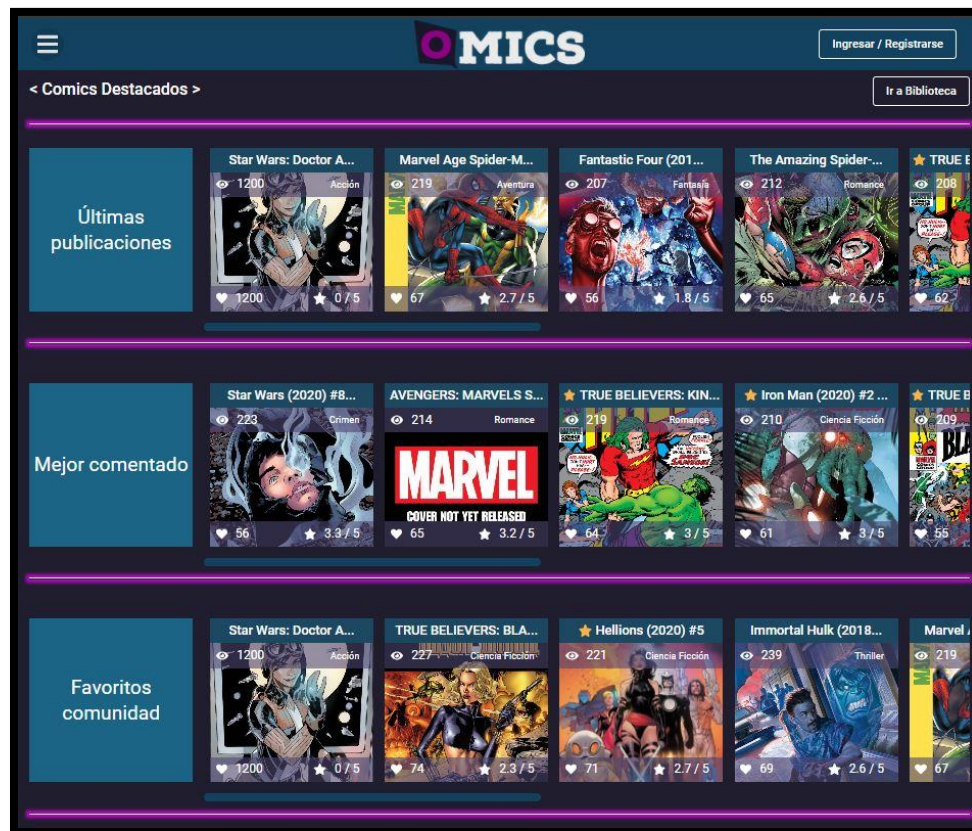


Figura 1. 60 Pantalla N° 3: Pantalla de Inicio Sin Barra Lateral. Sin loguearse.



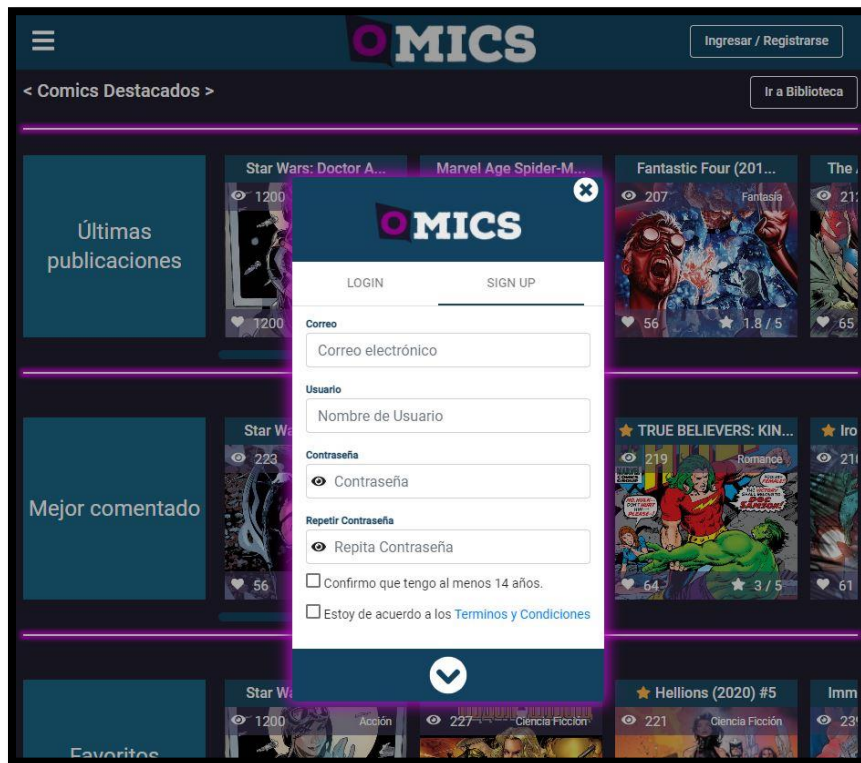


Figura 1. 61 Pantalla N° 4: Pantalla para Registrarse.

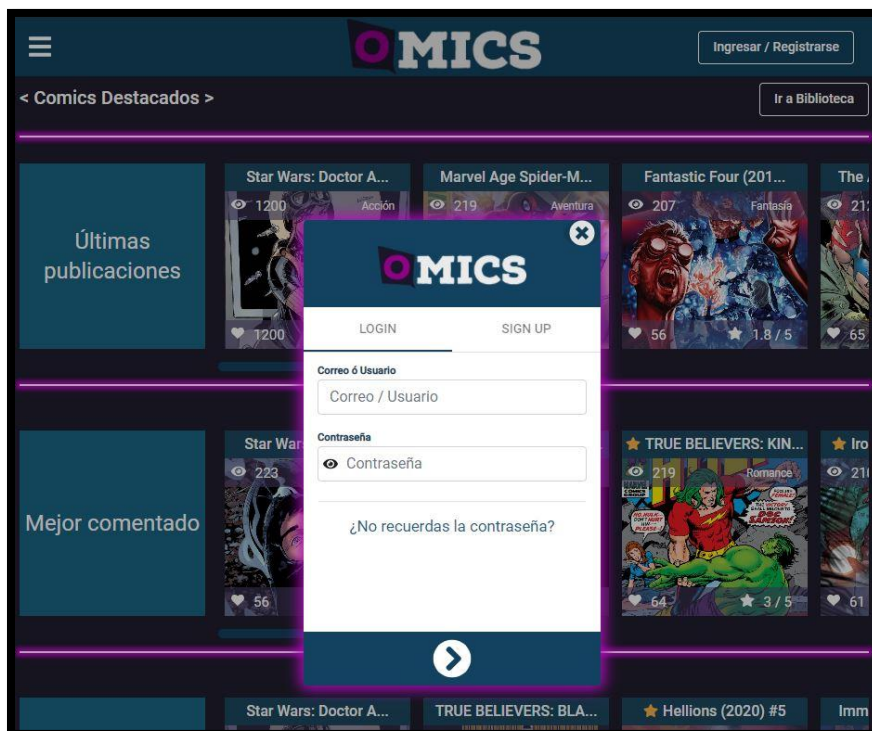


Figura 1. 62 Pantalla N° 5: Pantalla de Login.

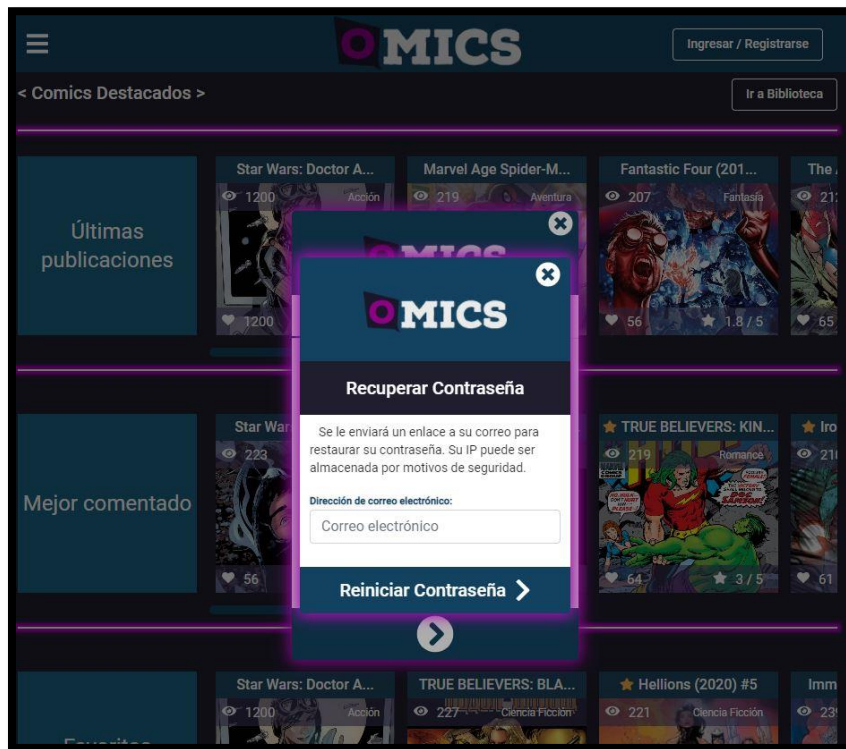


Figura 1. 63 Pantalla N° 6: Pantalla de Enviar Recuperar Contraseña.



Figura 1. 64 Pantalla N° 7: Pantalla de Nueva Contraseña.

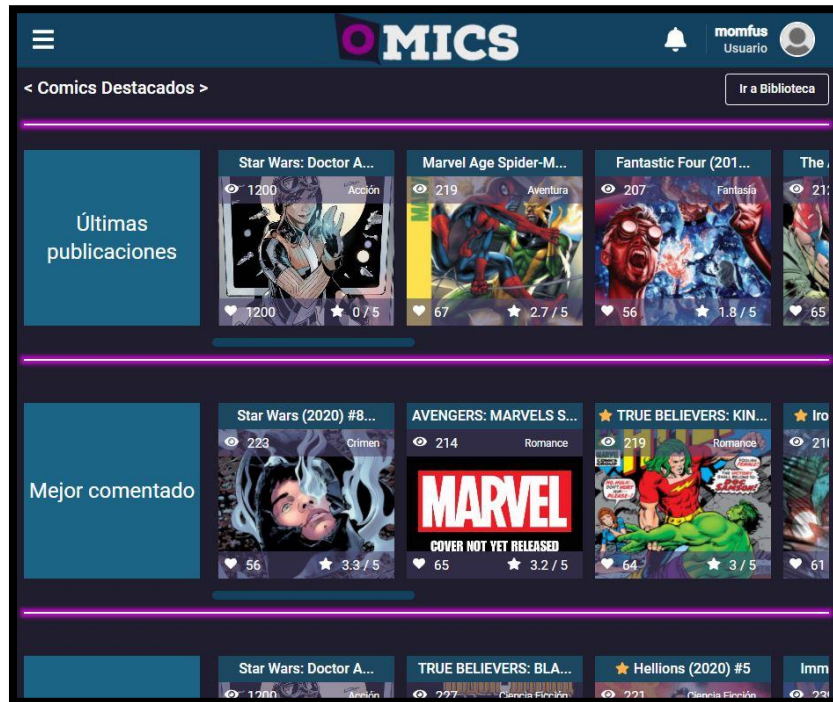


Figura 1. 65 Pantalla N° 8: Pantalla de Usuarios Logueados.

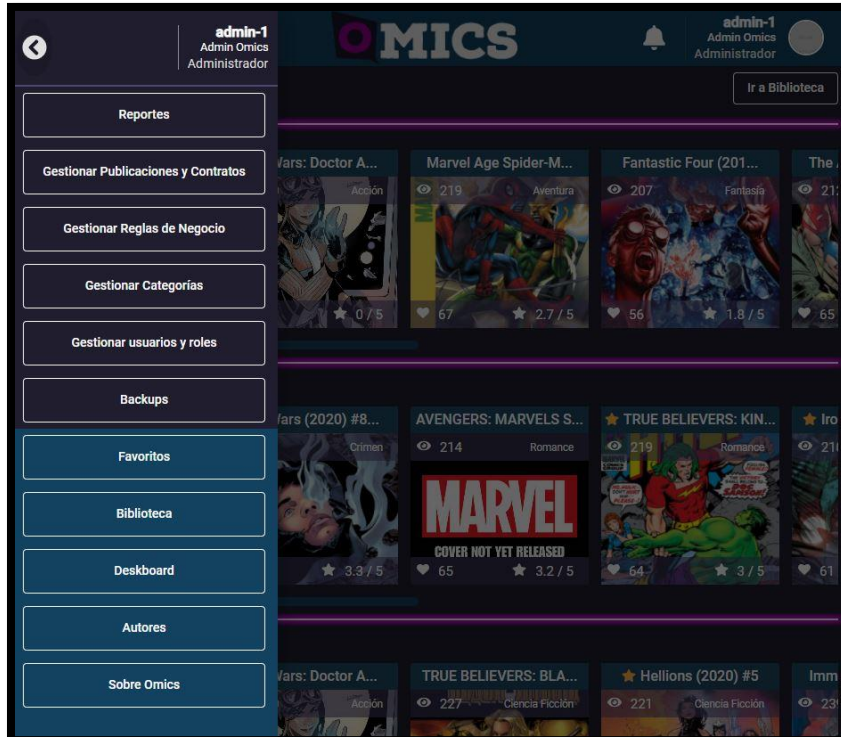


Figura 1. 66 Pantalla N° 9: Pantalla de Permisos de Usuario, con menú desplegable (lo de fondo violeta es los de acceso de administradores, y lo de azul para usuarios lectores y/o autores).

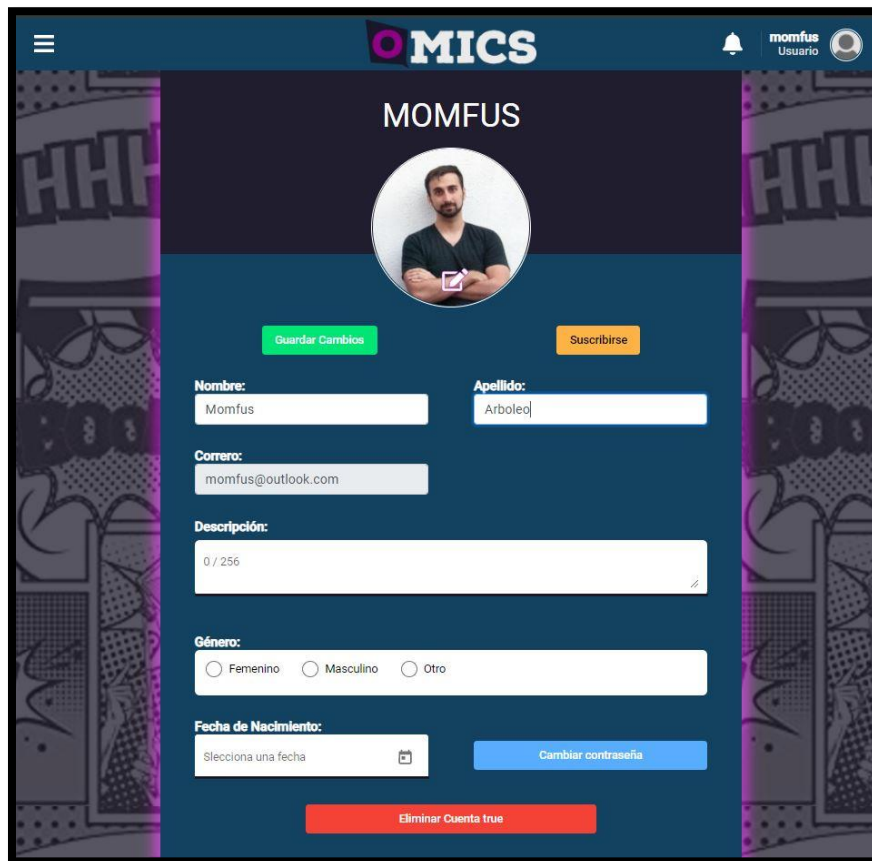


Figura 1. 67 Pantalla N° 10: Pantalla de Editar Perfil.

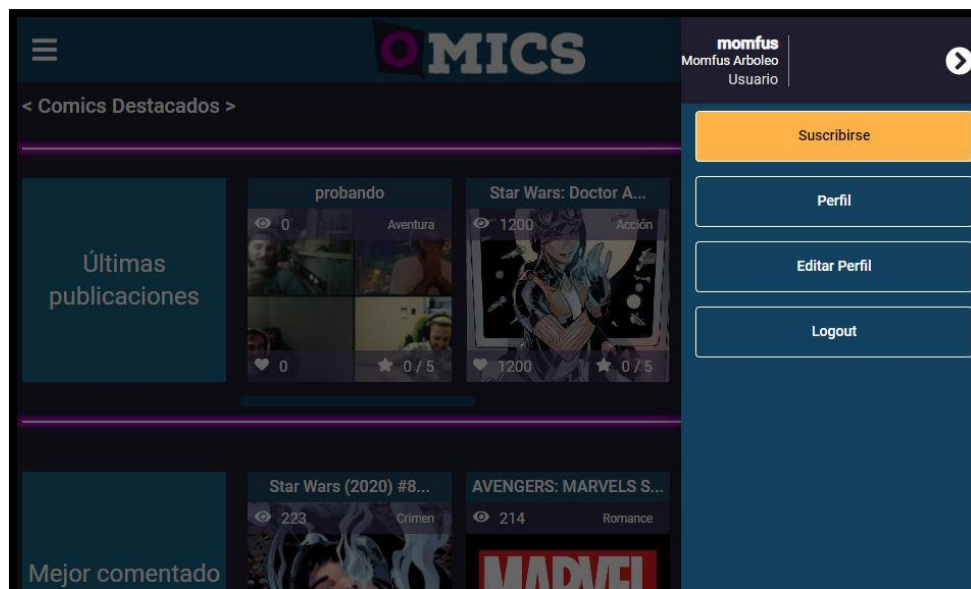


Figura 1. 68 Pantalla N° 11: Pantalla de Expandir Opciones de Perfil.



Figura 1. 69 Pantalla N° 12: Pantalla de Ver Perfil.

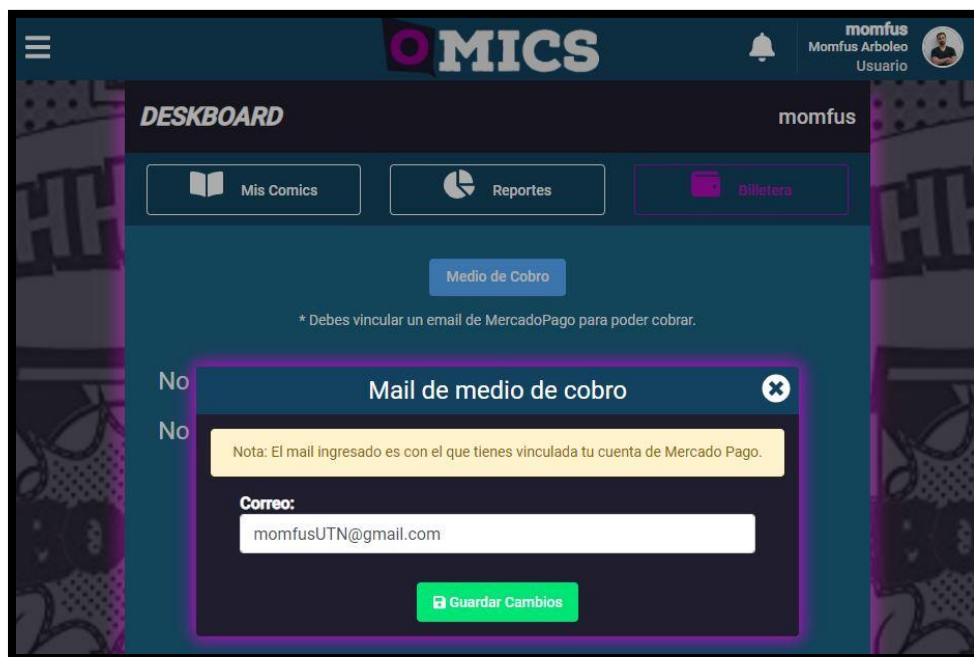


Figura 1. 70 Pantalla N° 13: Pantalla de Gestionar Medio de Pago.

### 4.2.2. Módulo de Catálogo y Búsqueda

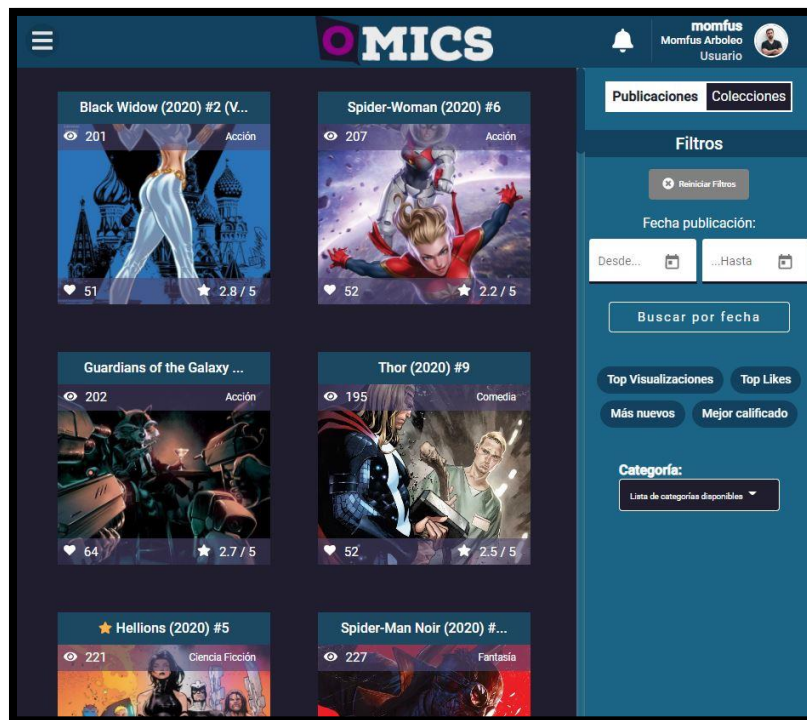


Figura 1. 71 Pantalla N° 14: Pantalla de Biblioteca General de Obras.

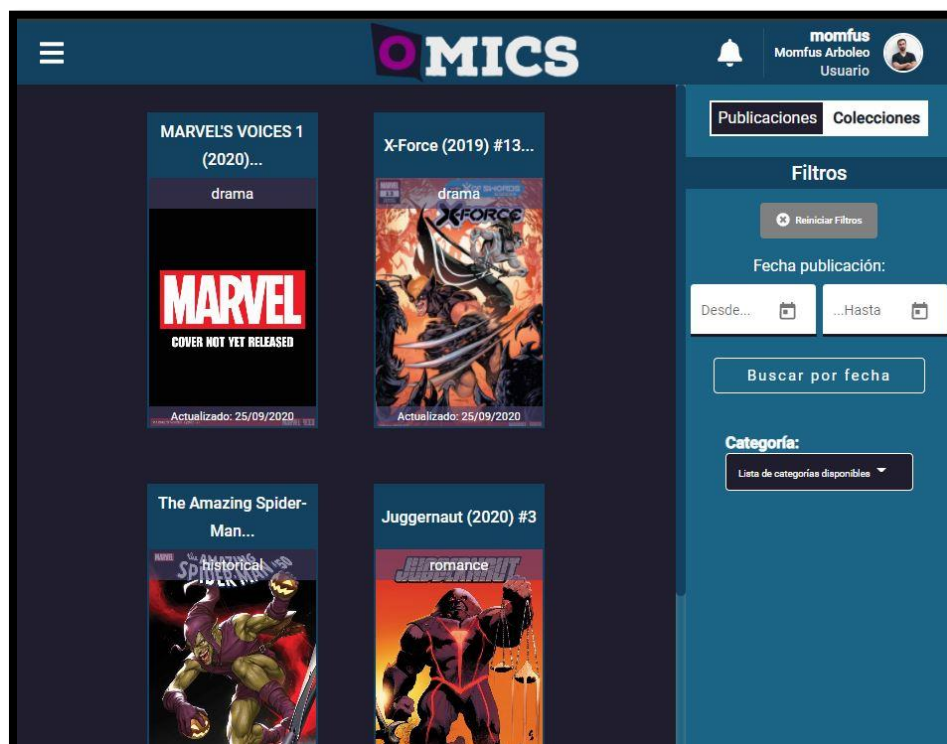


Figura 1. 72 Pantalla N° 15: Pantalla de Colecciones.

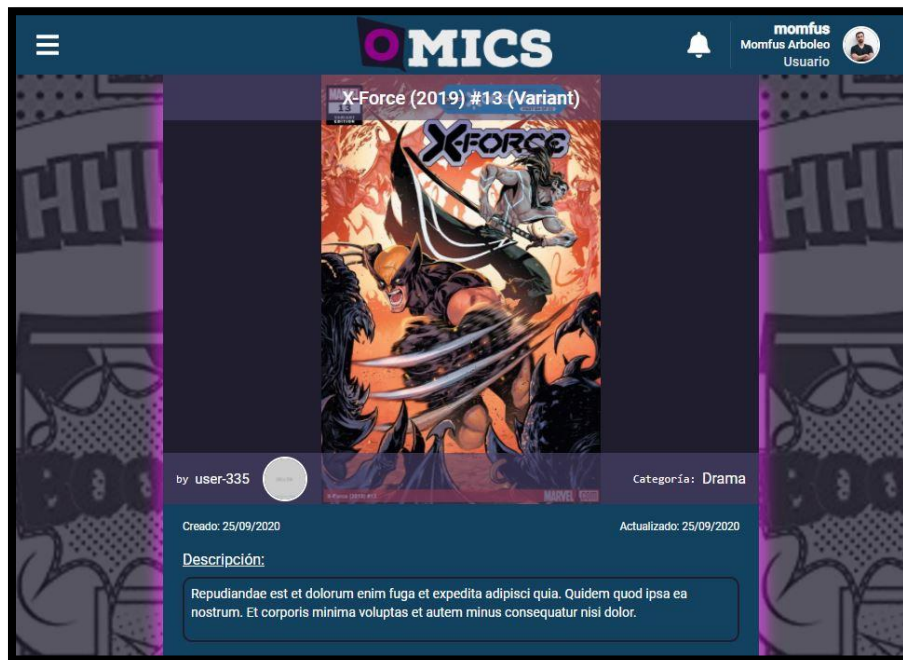


Figura 1. 73 Pantalla N° 16: Pantalla de Ver Detalle de Colección.

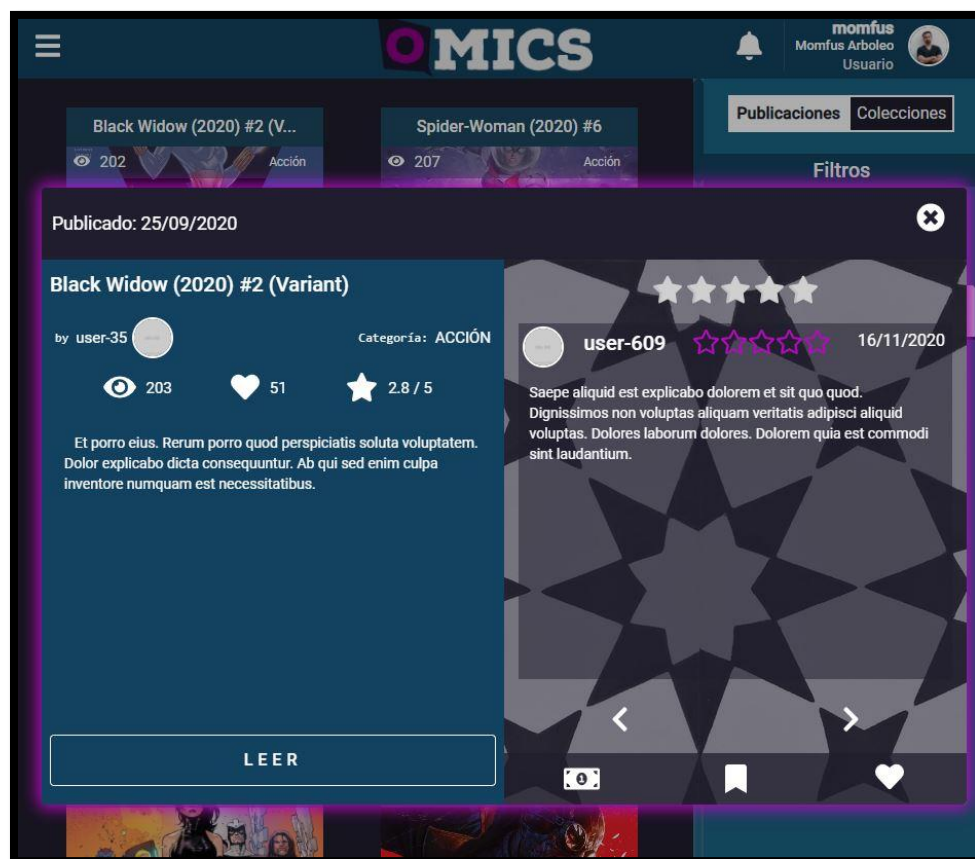


Figura 1. 74 Pantalla N° 17: Pantalla de Ver Detalle de Publicación.

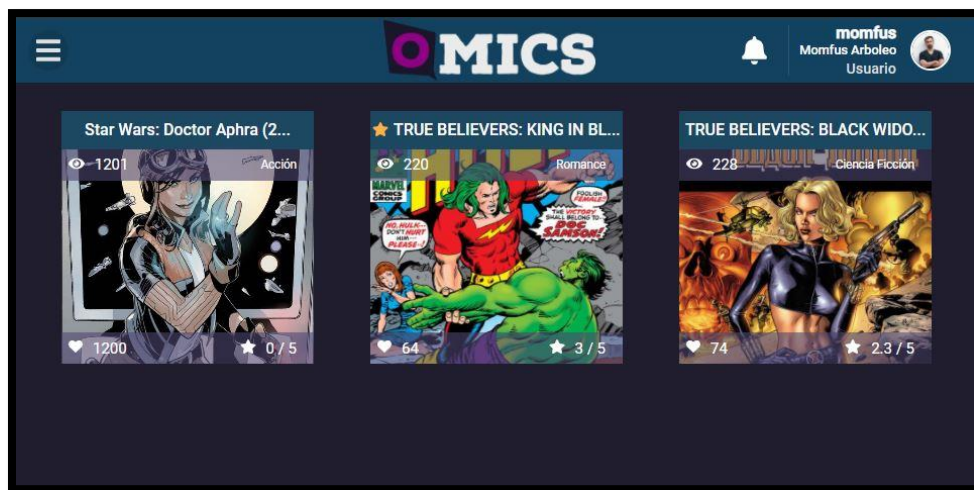


Figura 1. 75 Pantalla N° 18: Pantalla de Ver Favoritos Publicaciones.

### 4.2.3. Módulo de Visualización y Seguimiento

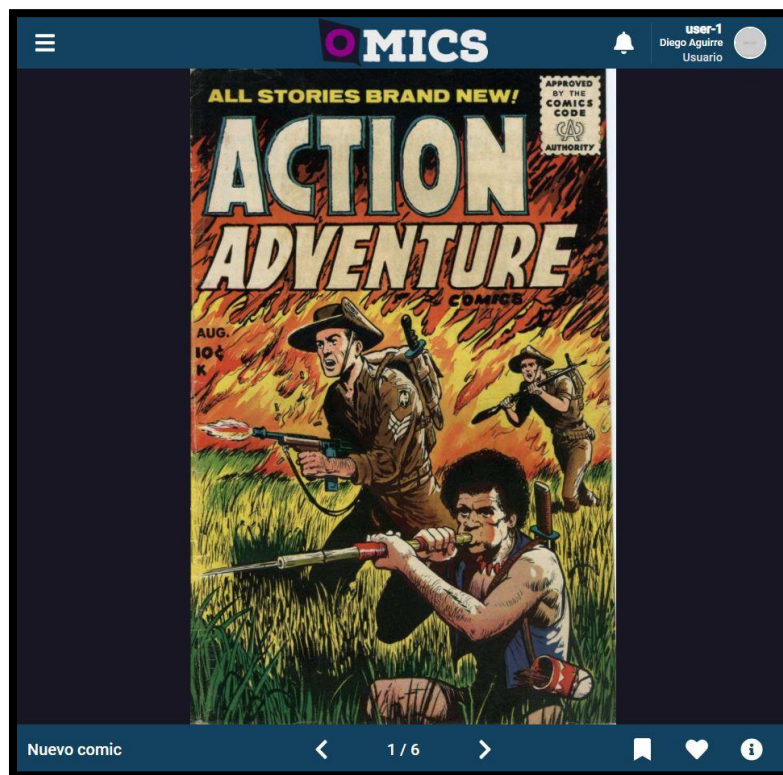


Figura 1. 76 Pantalla N° 19: Pantalla de Lector de Cómic.



#### 4.2.4. Módulo de Reseñas

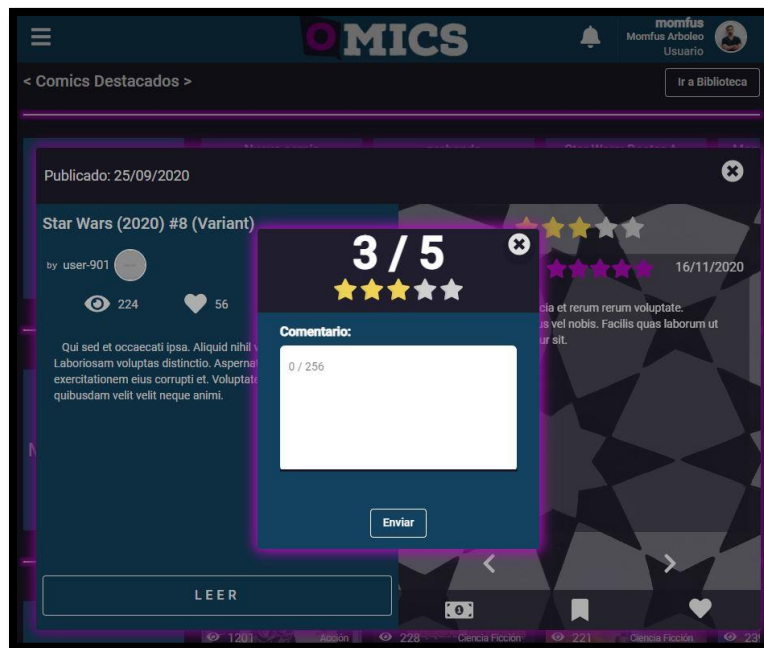


Figura 1. 77 Pantalla N° 20: Pantalla de Realizar Reseña.

#### 4.2.5. Módulo de Suscripción y Donación

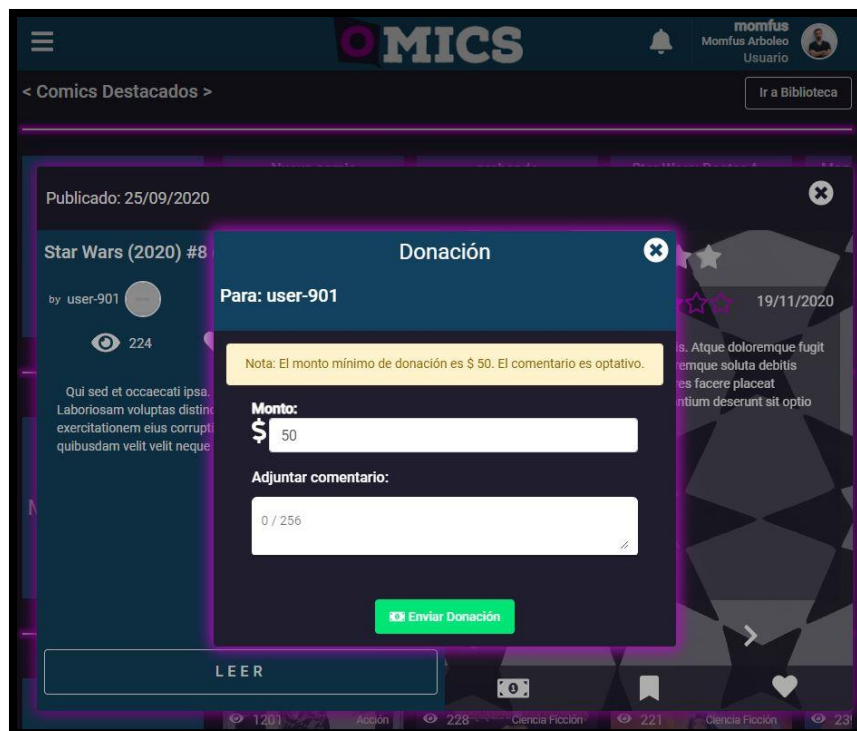


Figura 1. 78 Pantalla N° 21: Pantalla de Donar a Autor.

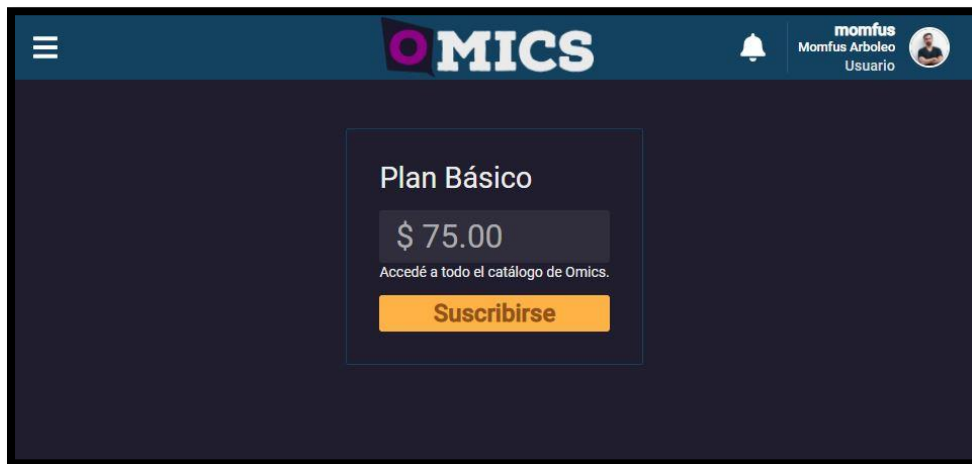


Figura 1. 79 Pantalla N° 22: Pantalla de Suscripción a la plataforma.



Figura 1. 80 Pantalla N° 23: Pantalla de Cancelar Suscripción.

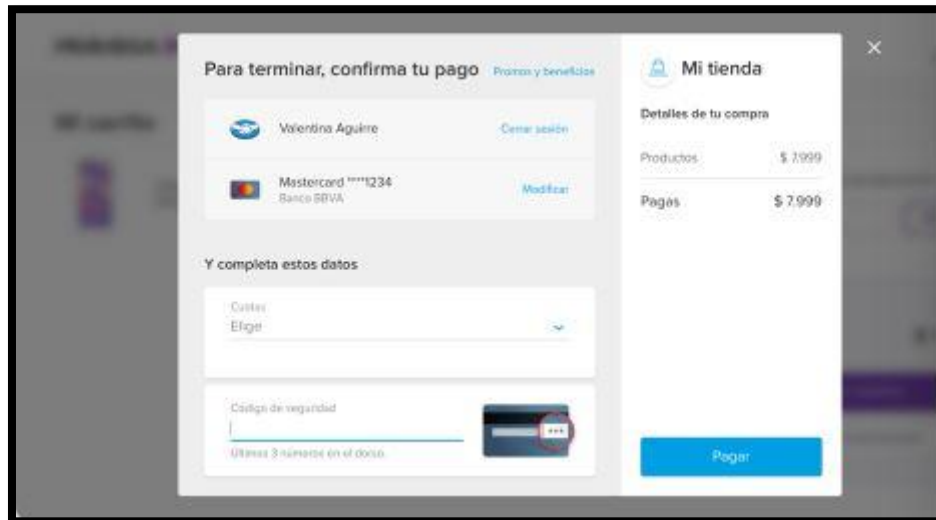


Figura 1. 81 Pantalla N° 24: Pantalla de Introducir Pago.

#### 4.2.6. Módulo de Publicación de Obras

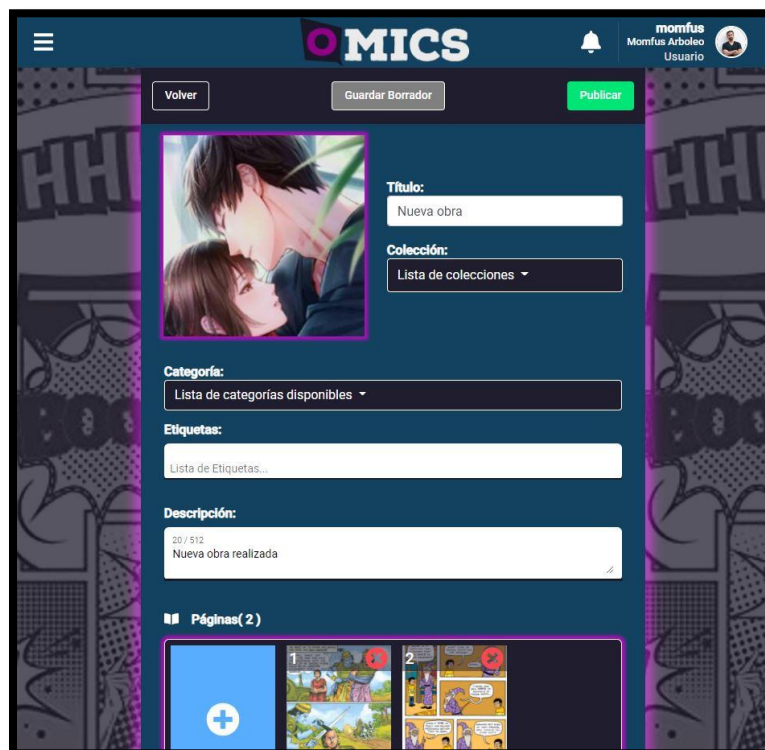


Figura 1. 82 Pantalla N° 25: Pantalla de Solicitar Publicar Obra.

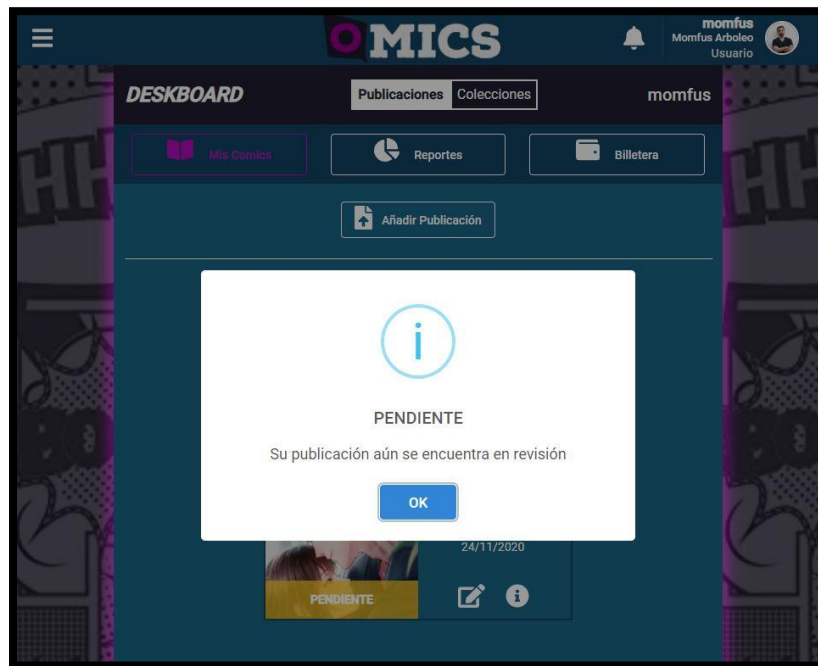


Figura 1. 83 Pantalla N° 26: Pantalla de Publicación Espera Aprobación.

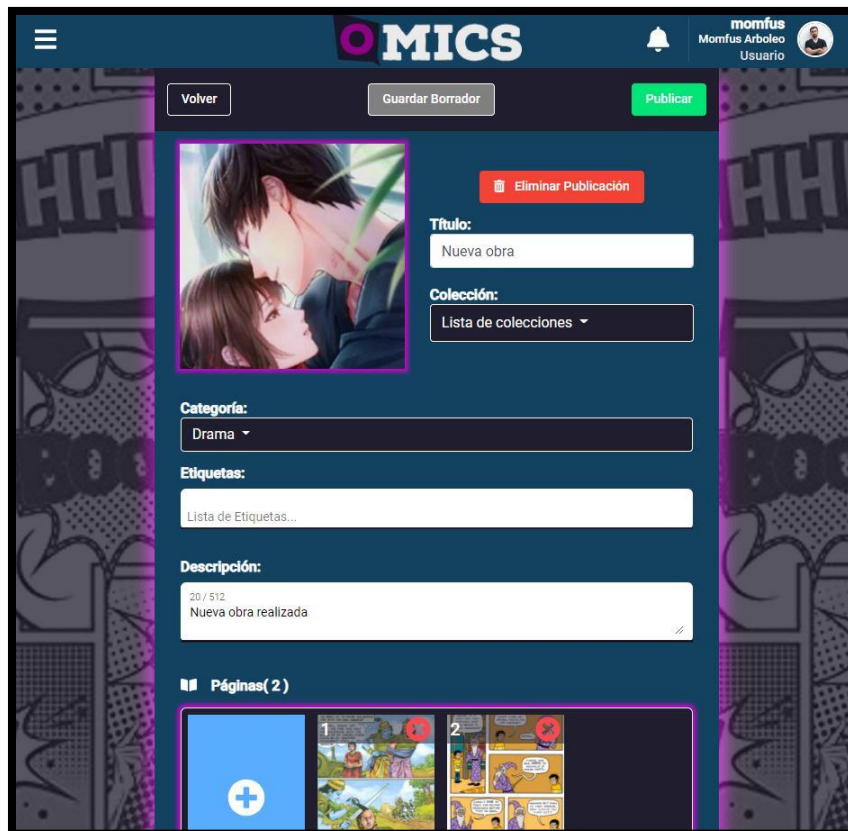


Figura 1. 84 Pantalla N° 27: Pantalla de Editar Publicación.

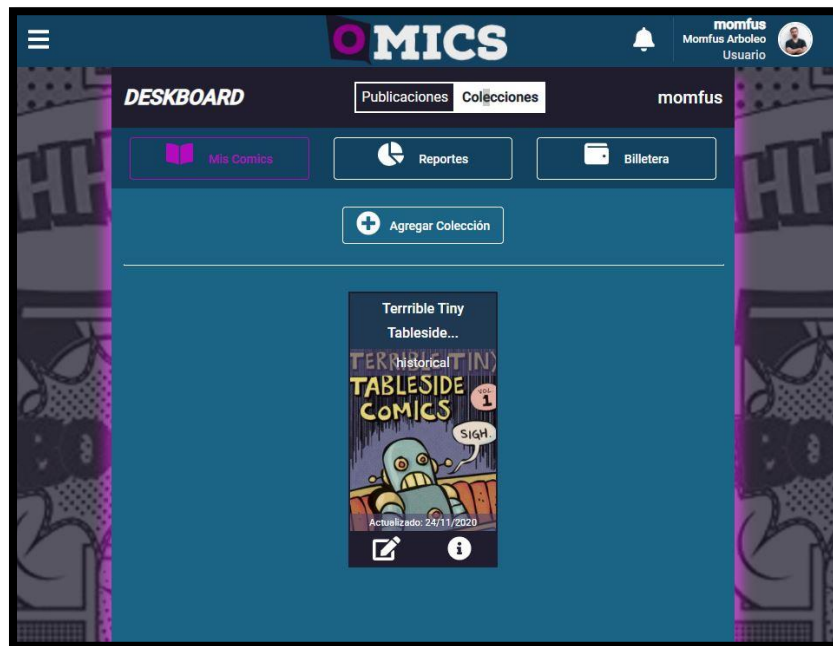


Figura 1. 85 Pantalla N° 28: Pantalla de Gestionar Colecciones.

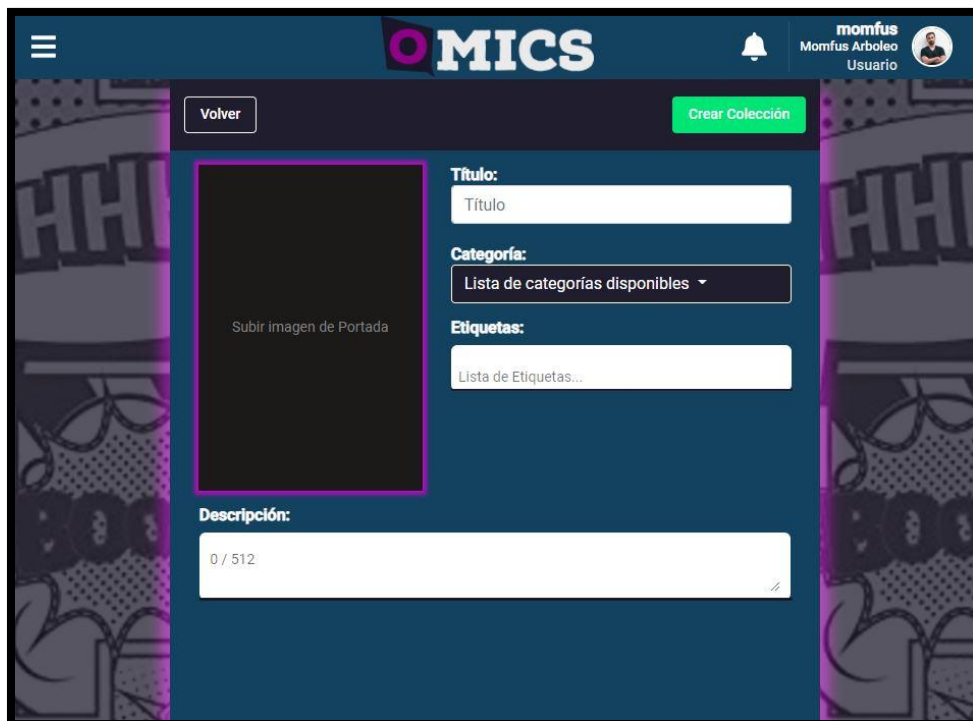


Figura 1. 86 Pantalla N° 29: Pantalla de Crear Colecciones.



Figura 1. 87 Pantalla N° 30: Pantalla de Editar Colecciones.

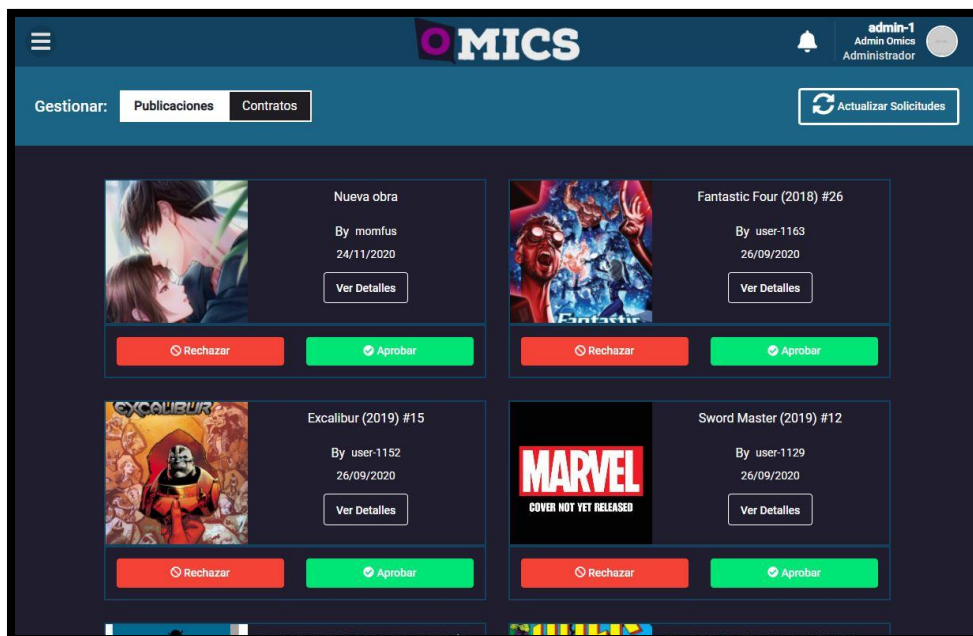


Figura 1. 88 Pantalla N° 31: Pantalla de Aceptar/Rechazar Publicaciones.

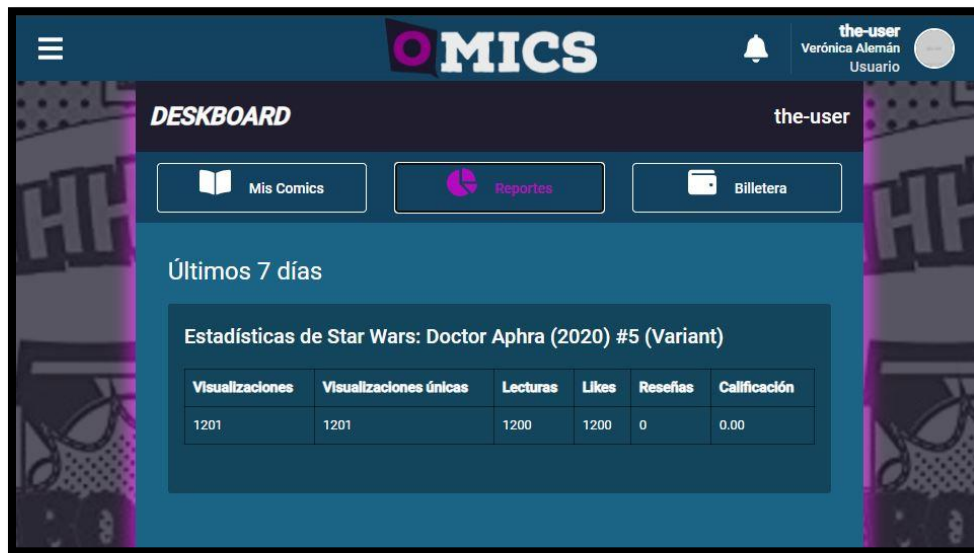


Figura 1. 89 Pantalla N° 32: Pantalla de Ver Estadísticas de Publicación.



Figura 1. 90 Pantalla N° 33: Pantalla de primera publicación aprobada.

### 4.2.7. Módulo de Notificaciones



Figura 1. 91 Pantalla N° 34: Pantalla de Notificaciones.

### 4.2.8. Módulo de Contrato

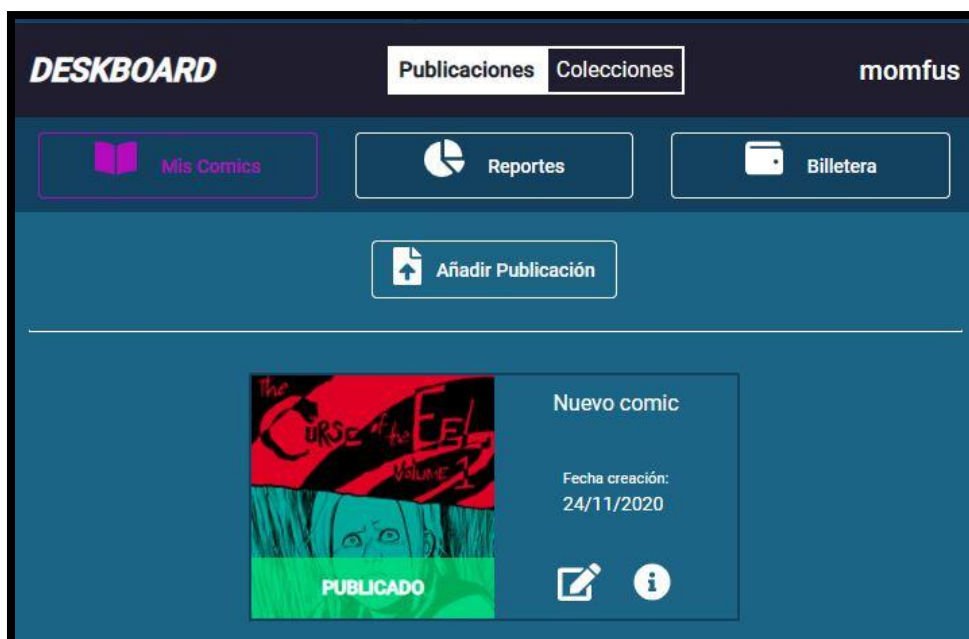




Figura 1. 92 Pantalla N° 35: Pantalla de Publicación Sin Contrato.

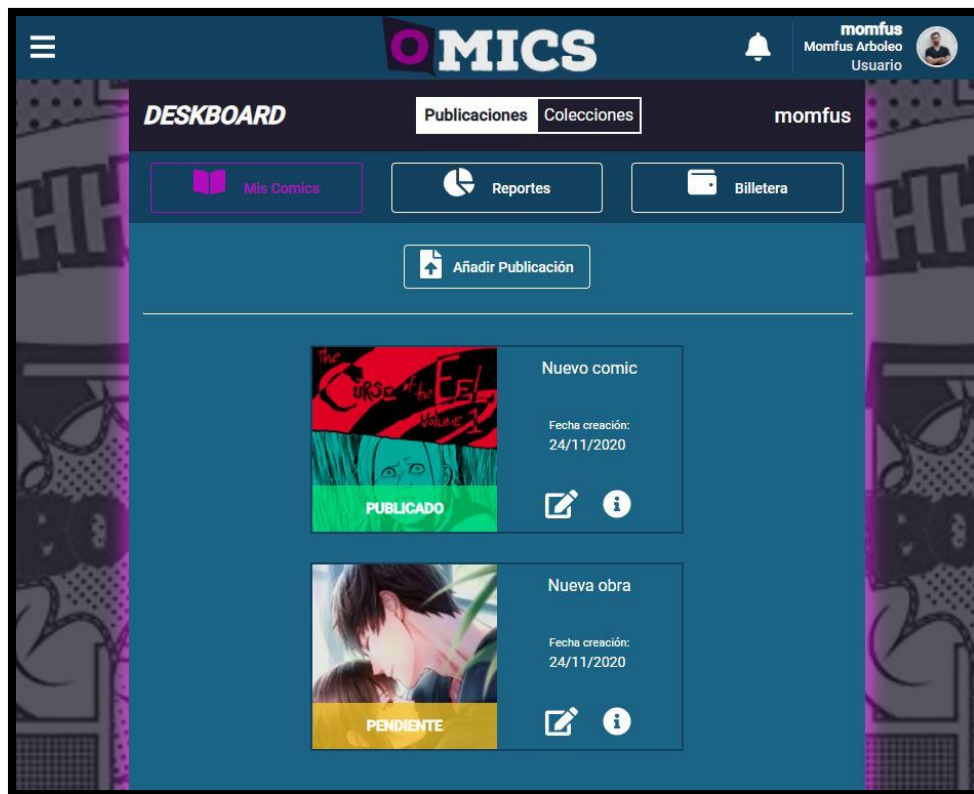


Figura 1. 93 Pantalla N° 36: Pantalla de Mis Comics.

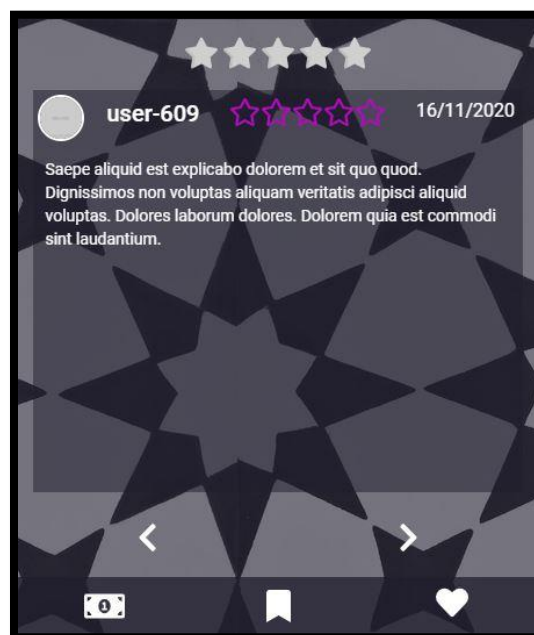


Figura 1. 94 Pantalla N° 37: Pantalla de Comentarios.



Figura 1. 95 Pantalla N° 38: Pantalla de Publicación Con Contrato.

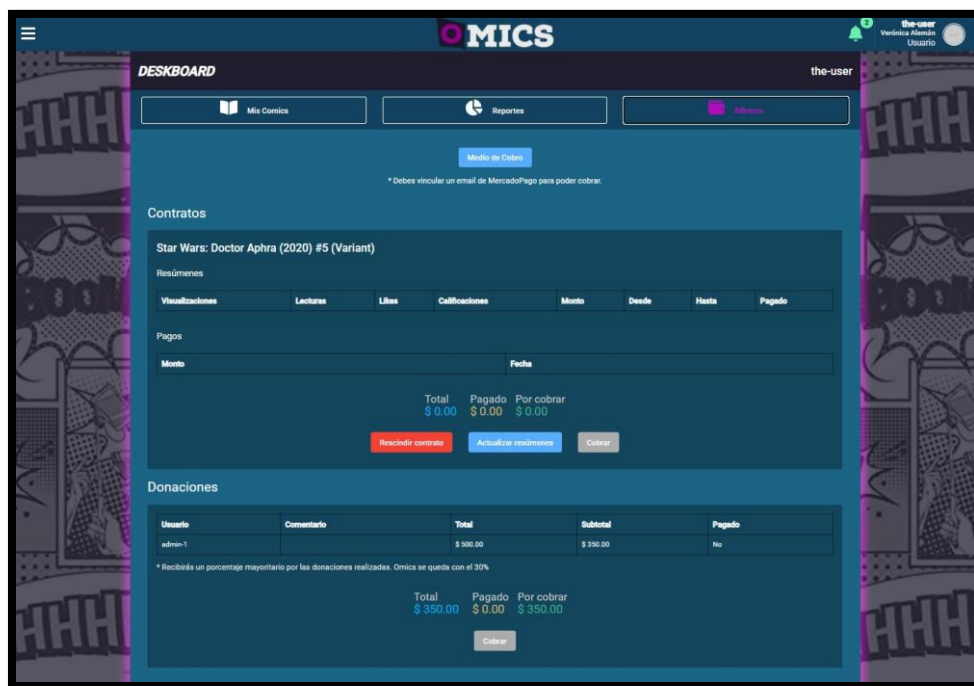


Figura 1. 96 Pantalla N° 39: Pantalla de Billetera Autor.

**Contratos**

**Excalibur (2019) #14**

Resúmenes

Visualizaciones	Lecturas	Likes	Calificaciones	Monto	Desde	Hasta	Pagado
637	145	100	2.00	\$ 976.00	26/09/2020	27/09/2020	Sí
251	259	296	5.00	\$ 351.00	27/09/2020	28/09/2020	Sí
437	490	813	5.00	\$ 467.00	28/09/2020	29/09/2020	Sí
920	542	159	2.00	\$ 49.00	29/09/2020	30/09/2020	Sí
786	134	231	2.00	\$ 108.00	30/09/2020	01/10/2020	Sí
280	494	756	5.00	\$ 90.00	01/10/2020	02/10/2020	Sí
478	573	194	4.00	\$ 689.00	02/10/2020	03/10/2020	No

Pagos

Monto	Fecha
\$ 2041.00	24/11/2020

Total \$ 2730.00    Pagado \$ 2041.00    Por cobrar \$ 689.00


Rescindir contrato    Actualizar resúmenes    Cobrar

Figura 1. 97 Pantalla N° 40: Pantalla de Publicación Ver Resúmenes Generados.

**OMICS**


admin-1 Admin Omics Administrador

Gestionar: Publicaciones    Contratos    Actualizar Solicitudes



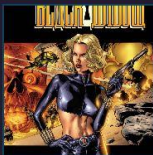
Immortal Hulk (2018) #40  
25/09/2020  
Ver Detalles

Rechazar    Aprobar



U.S.Agent (2020) #1  
25/09/2020  
Ver Detalles

Rechazar    Aprobar



TRUE BELIEVERS: BLACK WIDOW - YE...  
25/09/2020  
Ver Detalles

Rechazar    Aprobar

Figura 1. 98 Pantalla N° 41: Pantalla de Aprobar/Rechazar Contrato.

### 4.2.9. Módulo de Reportes

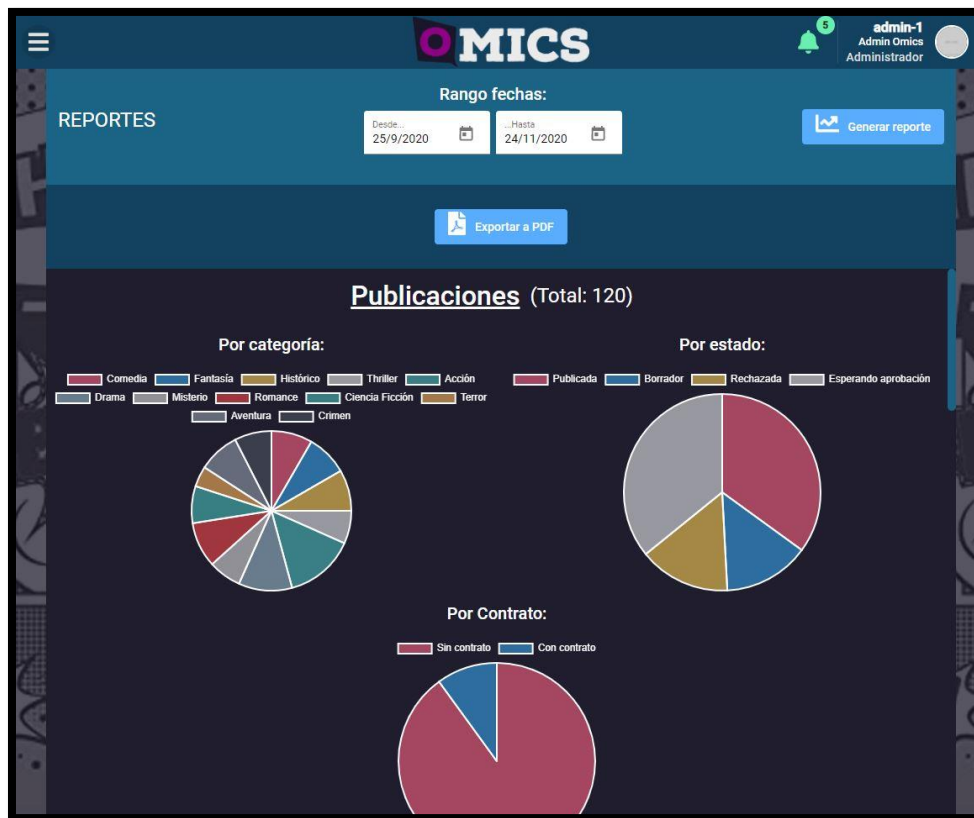


Figura 1. 99 Pantalla N° 42: Pantalla de Reportes (a).

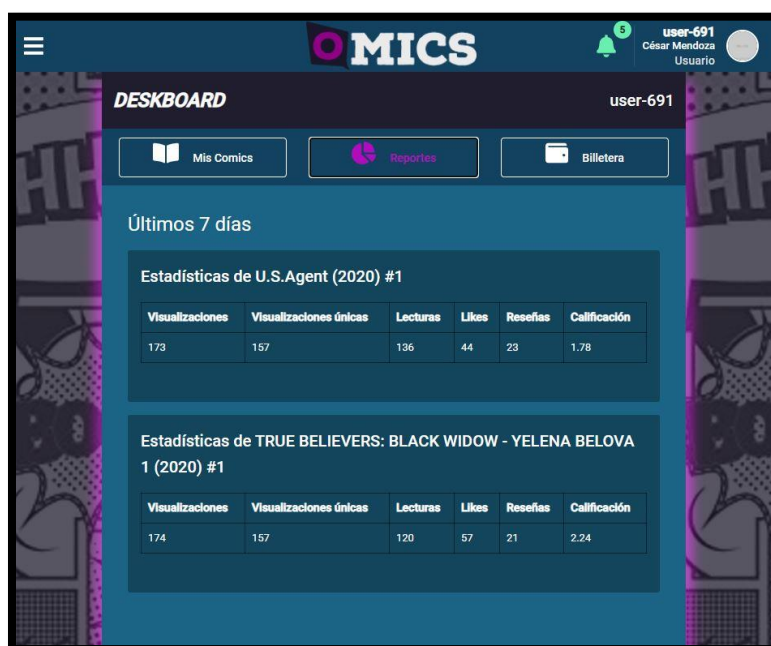


Figura 1. 100 Pantalla N° 43: Pantalla de Reportes (b).

### 4.2.10. Módulo de Configuración



Figura 1. 101 Pantalla N° 44: Gestionar Reglas de Negocios.



Figura 1. 102 Pantalla N° 45: Pantalla de Administrador de Gestionar Categorías.



Figura 1. 103 Pantalla N° 46: Gestionar Roles y Visualizar Tabla de Roles Disponibles.

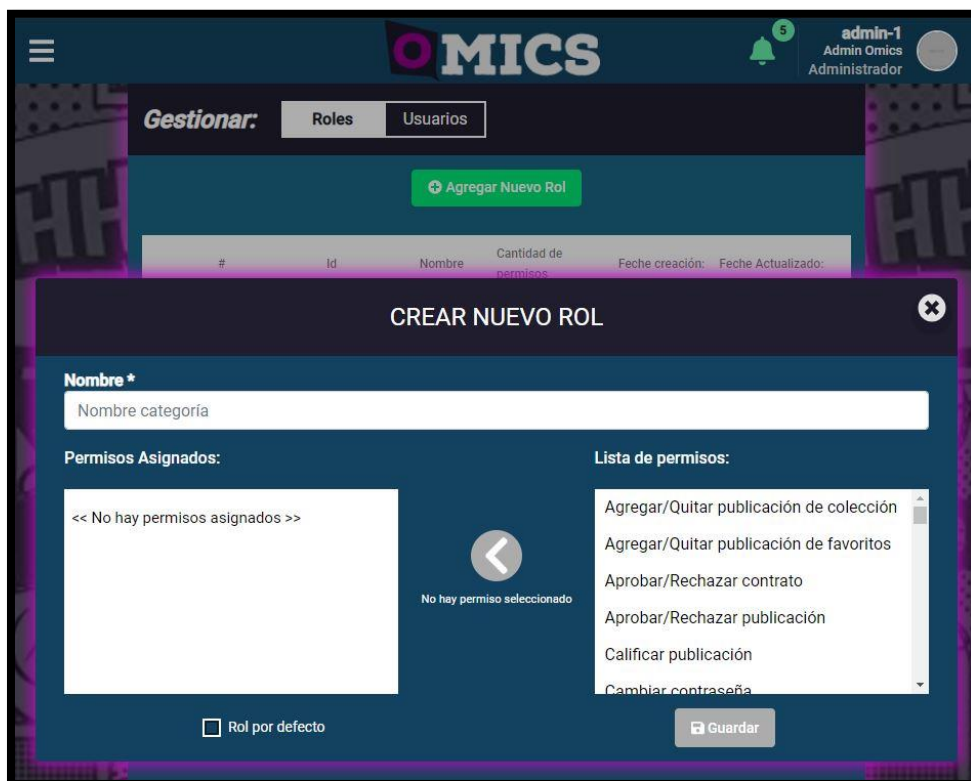


Figura 1. 104 Pantalla N° 47: Crear Nuevo Rol.

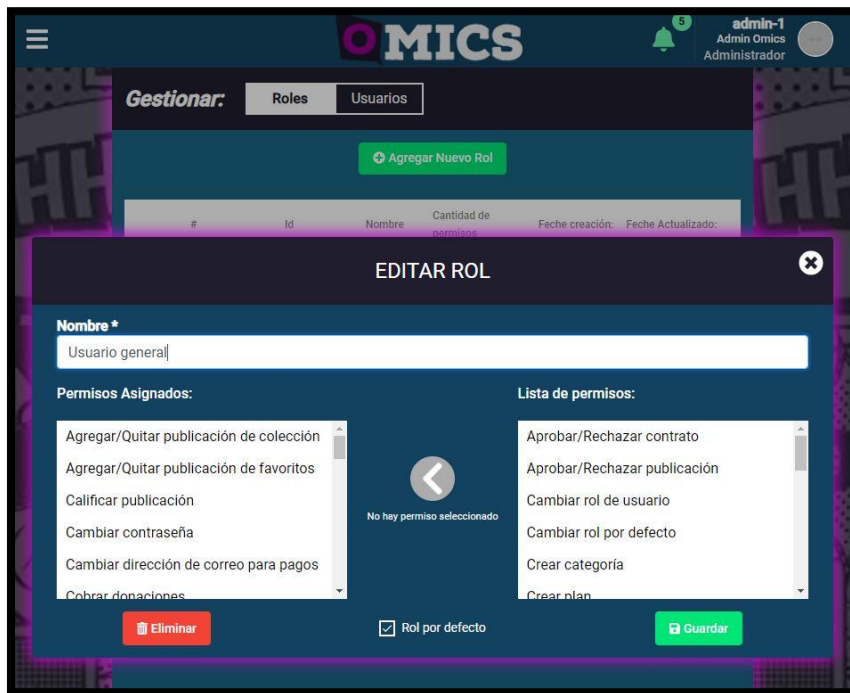


Figura 1. 105 Pantalla N° 48: Editar Rol Existente.

**Gestionar:** Roles Usuarios

Filtrar usuarios:

Buscar por nombre  Seleccionar un rol

Limpiar Filtrar

Items por página: 10 1 - 10 of 121

Username	Name	Rol	Usuario desde	¿Validado?
momfus	Momfus Arboleo	Usuario	24/11/2020	Sí
admin-1	Admin Omics	Administrador	24/11/2020	Sí
content-manager-1	Content Manager Omics	Gestor de contenido	24/11/2020	Sí
the-user	Verónica Alemán	Usuario	21/11/2020	Sí
user-1198	Carlota Romero	Gestor de contenido	26/09/2020	Sí
user-1197	Marcos Cortés	Usuario	26/09/2020	Sí
user-1196	Santiago Orozco	Usuario	26/09/2020	Sí
user-1195	Enrique Jáquez	Usuario	26/09/2020	Sí
user-1194	Guillermo Zamudio	Usuario	26/09/2020	Sí
user-1193	Samuel Leal	Usuario	26/09/2020	Sí

Figura 1. 106 Pantalla N° 49: Gestionar Usuarios y sus Roles Asignados.



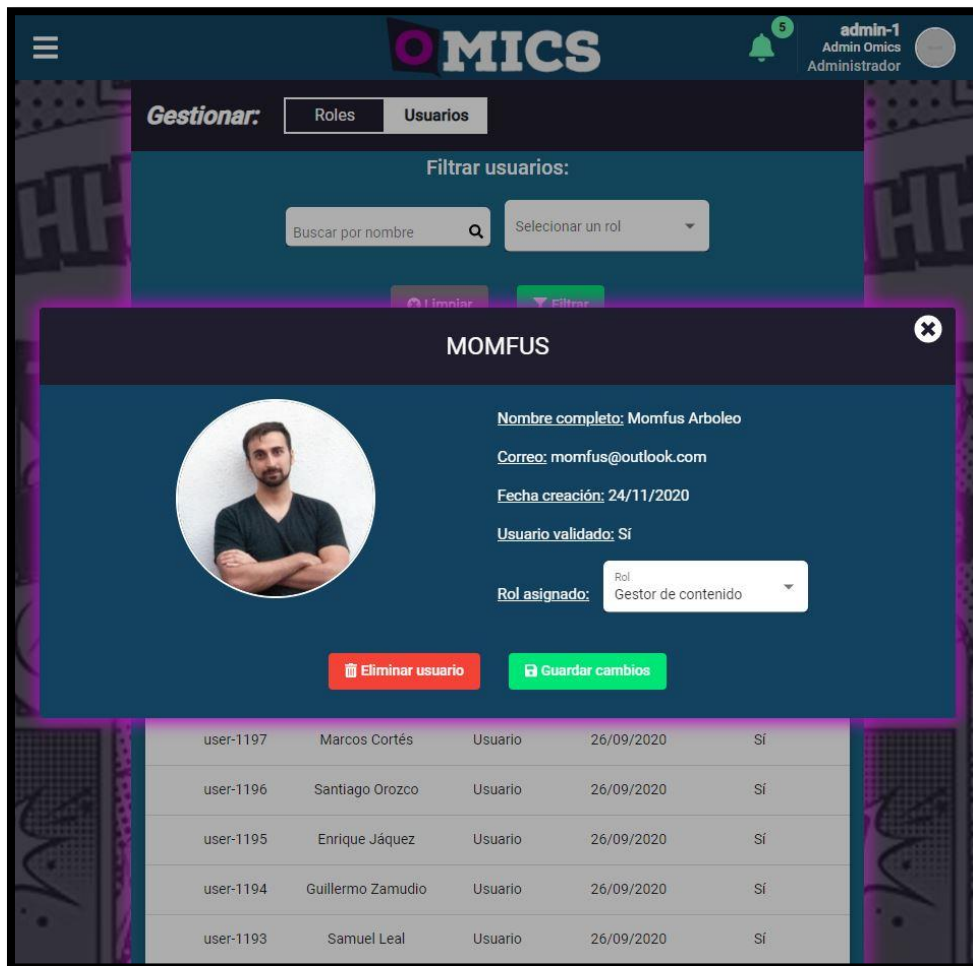


Figura 1. 107 Pantalla N° 50: Cambiar Rol a un Usuario Existente.

### 4.3. Correspondencia entre las Salidas del Sistema, las Historias de Usuario, y las Pantallas:

A continuación, se ve reflejado en la tabla la correspondencia que se ha realizado entre los casos de uso del proyecto *Omics*, las salidas del sistema, historias de usuario y pantallas que conllevan una relación entre sí. De esta manera se logra comprender mejor cada módulo que se ha definido.

Módulo de Registro y Actividad de Usuario	CASOS DE USO		
	<ol style="list-style-type: none"> <li>1. CU 01 - Registrar Cuenta de Usuario.</li> <li>2. CU 02 - Iniciar Sesión.</li> <li>3. CU 03 - Recuperar Contraseña.</li> <li>4. CU 04 - Cambiar Contraseña.</li> <li>5. CU 05 - Editar Cuenta de Usuario.</li> <li>6. CU 06 - Ver Perfil de Usuario.</li> <li>7. CU 07 - Desactivar Cuenta de Usuario.</li> </ol>		
	SALIDAS DEL SISTEMA	HISTORIAS DE USUARIO	PANTALLAS Y DIAGRAMAS
Salida N°1: Acerca de Nosotros.  Salida N°2: Registrarse en la Plataforma.	Historia de Usuario N°1: Registrar Cuenta de Usuario.	Figura 1. 58 Pantalla N° 1: Pantalla de Bienvenida a Omics.  Figura 1. 59 Pantalla N° 2: Pantalla de Inicio Con Barra Lateral. Sin loguearse.  Figura 1. 60 Pantalla N° 3: Pantalla de Inicio Sin Barra Lateral. Sin loguearse.  Figura 1. 61 Pantalla N° 4: Pantalla para Registrarse.	

	Salida N°3: Inicio de Sesión.	Historia de Usuario N°2: Iniciar Sesión.	Figura 1. 62 Pantalla N° 5: Pantalla de Login.
	Salida N°4: Recuperar la Contraseña.	Historia de Usuario N°3: Recuperar Contraseña.	Figura 1. 63 Pantalla N° 6: Pantalla de Enviar Recuperar Contraseña.
	Salida N°5: Cambiar la Contraseña.	Historia de Usuario N°4: Cambiar Contraseña.	Figura 1. 64 Pantalla N° 7: Pantalla de Nueva Contraseña. Figura 1. 65 Pantalla N° 8: Pantalla de Usuarios Logueados. Figura 1. 66 Pantalla N° 9: Pantalla de Permisos de Usuario, con menú desplegable (lo de fondo violeta es los de acceso de administradores, y lo de azul para usuarios lectores y/o autores).
	Salida N°6: Editar Cuenta.	Historia de Usuario N°5: Editar Cuenta de Usuario.	Figura 1. 67 Pantalla N° 10: Pantalla de Editar
	Salida N°7: Ver Perfil.	Historia de Usuario N°6: Ver Perfil de Usuario.	Figura 1. 68 Pantalla N° 11: Pantalla de Expandir Opciones de Perfil. Figura 1. 69 Pantalla N° 12: Pantalla de Ver Perfil.

			Figura 1. 70 Pantalla N° 13: Pantalla de Gestionar Medio de Pago.
	Salida N°8: Desactivar Cuenta.	Historia de Usuario N°7: Desactivar Cuenta de Usuario.	Figura 1. 67 Pantalla N° 10: Pantalla de Editar
<b>Módulo de Catálogo y Búsqueda</b>	<b>CASOS DE USO</b>		
	<p>8. CU 08 - Buscar y Filtrar Contenido.            9. CU 09 - Ver Detalle de Contenido.            10. CU 10 - Ver Favoritos.</p>		
	<b>SALIDAS DEL SISTEMA</b>	<b>HISTORIAS DE USUARIO</b>	<b>PANTALLAS Y DIAGRAMAS</b>
	Salida N°9: Mis Filtros y Contenidos.	Historia de Usuario N°8: Buscar y Filtrar Contenido.	Figura 1. 71 Pantalla N° 14: Pantalla de Biblioteca General de Obras.
	Salida N°10: Ver el Contenido de la Obra.	Historia de Usuario N°9: Ver Detalle de Contenido.	<p>Figura 1. 72 Pantalla N° 15: Pantalla de Colecciones.</p> <p>Figura 1. 73 Pantalla N° 16: Pantalla de Ver Detalle de Colección.</p> <p>Figura 1. 74 Pantalla N° 17: Pantalla de Ver Detalle de Publicación.</p>
	Salida N°11: Ver Contenido.	Historia de Usuario N°10: Ver Favoritos.	Figura 1. 75 Pantalla N° 18: Pantalla de Ver Favoritos Publicaciones.
	<b>CASOS DE USO</b>		

<b>Módulo de Visualización y Seguimiento</b>	11. CU 11 - Leer Publicación. 12. CU 12 - Añadir y Eliminar Contenido de Favoritos. 13. CU 13 - Seguir/Dejar de Seguir a Usuario.		
	<b>SALIDAS DEL SISTEMA</b>	<b>HISTORIAS DE USUARIO</b>	<b>PANTALLAS Y DIAGRAMAS</b>
	Salida N°12: Ver las Publicaciones.	Historia de Usuario N°11: Leer Publicación.	Figura 1. 76 Pantalla N° 19: Pantalla de Lector de Cómic.
	Salida N°13: Agregar y Eliminar de Favoritos Contenido.	Historia de Usuario N°12: Añadir y Eliminar Contenido de Favoritos.	Figura 1. 76 Pantalla N° 19: Pantalla de Lector de Cómic.
	Salida N°14: Ver Información del Usuario.	Historia de Usuario N°13: Seguir/Dejar de Seguir a Usuario.	Figura 1. 76 Pantalla N° 19: Pantalla de Lector de Cómic.
<b>Módulo de Reseñas</b>	<b>CASOS DE USO</b>		
	14. CU 14 - Calificar Publicación. 15. CU 15 - Dar Like a Publicación.		
	<b>SALIDAS DEL SISTEMA</b>	<b>HISTORIAS DE USUARIO</b>	<b>PANTALLAS Y DIAGRAMAS</b>
	Salida N°15: Calificar Publicaciones.	Historia de Usuario N°14: Calificar Publicación.	Figura 1. 77 Pantalla N° 20: Pantalla de Realizar Reseña.
	Salida N°16: Dar Like a Publicación.	Historia de Usuario N°15: Dar Like a Publicación.	Figura 1. 77 Pantalla N° 20: Pantalla de Realizar Reseña.
	<b>CASOS DE USO</b>		
	16. CU 16 - Donar a Autor. 17. CU 17 - Realizar Suscripción. 18. CU 18 - Cancelar Suscripción. 19. CU 19 - Pagar Suscripción.		

	<b>SALIDAS DEL SISTEMA</b>	<b>HISTORIAS DE USUARIO</b>	<b>PANTALLAS Y DIAGRAMAS</b>
<b>Módulo de Suscripción y Donación</b>	Salida N°17: Realizar Donación a Autor.	Historia de Usuario N°16: Donar a Autor.	Figura 1. 78 Pantalla N° 21: Pantalla de Donar a Autor.
	Salida N°18: Suscribirse a la Plataforma.	Historia de Usuario N°17: Realizar Suscripción.	Figura 1. 79 Pantalla N° 22: Pantalla de Suscripción a la plataforma.  Diagrama 1. 17 Diagrama de Estado para Suscripción.
	Salida N°19: Cancelar la Suscripción.	Historia de Usuario N°18: Cancelar Suscripción.	Figura 1. 80 Pantalla N° 23: Pantalla de Cancelar Suscripción.  Diagrama 1. 17 Diagrama de Estado para Suscripción.
	Salida N°20: Pagar Suscripción.	Historia de Usuario N°19: Pagar Suscripción.	Figura 1. 81 Pantalla N° 24: Pantalla de Introducir Pago.  Diagrama 1. 17 Diagrama de Estado para Suscripción.
	<b>CASOS DE USO</b>		
	20. CU 20 - Publicar Obra. 21. CU 21 - Modificar Publicación. 22. CU 22 - Gestionar Colecciones. 23. CU 23 - Aceptar/Rechazar Publicación. 24. CU 24 - Ver Estadísticas de Publicaciones.		

	<b>SALIDAS DEL SISTEMA</b>	<b>HISTORIAS DE USUARIO</b>	<b>PANTALLAS Y DIAGRAMAS</b>
<b>Módulo de Publicación de Obras</b>	Salida N°21: Publicar mi Obra.	Historia de Usuario N°20: Publicar Obra.	Figura 1. 82 Pantalla N° 25: Pantalla de Solicitar Publicar Obra.  Figura 1. 83 Pantalla N° 26: Pantalla de Publicación Espera Aprobación.  Diagrama 1. 15 Diagrama de Estado para Publicación.
	Salida N°22: Editar Obra.	Historia de Usuario N°21: Modificar Publicación.	Figura 1. 84 Pantalla N° 27: Pantalla de Editar Publicación.
	Salida N°23: Editar Colecciones.	Historia de Usuario N°22: Gestionar Colecciones.	Figura 1. 85 Pantalla N° 28: Pantalla de Gestionar Colecciones.  Figura 1. 86 Pantalla N° 29: Pantalla de Crear Colecciones.  Figura 1. 87 Pantalla N° 30: Pantalla de Editar Colecciones.
	Salida N°24: Agregar o Rechazar Publicación.	Historia de Usuario N°23: Aceptar/Rechazar Publicación.	Figura 1. 88 Pantalla N° 31: Pantalla de Aceptar/Rechazar Publicaciones.  Figura 1. 89 Pantalla N° 32: Pantalla de

			Ver Estadísticas de Publicación.  Figura 1. 90 Pantalla N° 33: Pantalla de primera publicación aprobada.  Diagrama 1. 15 Diagrama de Estado para Publicación.
	Salida N°25: Estadísticas de Publicaciones.	Historia de Usuario N°24: Ver Estadística de Publicaciones.	Figura 1. 89 Pantalla N° 32: Pantalla de Ver Estadísticas de Publicación.
<b>Módulo de Notificaciones</b>	<b>CASOS DE USO</b>		
	25. CU 25 - Notificar Actividad de Autores y Contenido. 26. CU 26 - Notificar Resumen de Pago.		
	<b>SALIDAS DEL SISTEMA</b>	<b>HISTORIAS DE USUARIO</b>	<b>PANTALLAS Y DIAGRAMAS</b>
	Salida N°26: Recibir Notificaciones de Autor.	Historia de Usuario N°25: Notificar Actividad de Autores y Contenido.	Figura 1. 91 Pantalla N° 34: Pantalla de Notificaciones.
	Salida N°27: Recibir Notificación de Resumen de Pago.	Historia de Usuario N°26: Notificar Resumen de Pago.	Figura 1. 91 Pantalla N° 34: Pantalla de Notificaciones.
	<b>CASOS DE USO</b>		
	27. CU 27 - Requerir Contrato. 28. CU 28 - Rescindir Contrato. 29. CU 29 - Cobrar por Resumen de Publicación. 30. CU 30 - Generar Resumen de Publicación. 31. CU 31 - Aceptar/Rechazar Contrato.		



	<b>SALIDAS DEL SISTEMA</b>	<b>HISTORIAS DE USUARIO</b>	<b>PANTALLAS Y DIAGRAMAS</b>
<b>Módulo de Contrato</b>	Salida N°28: Solicitar Contrato.	Historia de Usuario N°27: Requerir Contrato.	Figura 1. 92 Pantalla N° 35: Pantalla de Publicación Sin Contrato.  Figura 1. 93 Pantalla N° 36: Pantalla de Mis Comics.  Figura 1. 94 Pantalla N° 37: Pantalla de Comentarios.  Diagrama 1. 16 Diagrama de Estado para Contrato.
	Salida N°29: Rescindir Contrato.	Historia de Usuario N°28: Rescindir Contrato.	Figura 1. 95 Pantalla N° 38: Pantalla de Publicación Con Contrato.  Diagrama 1. 16 Diagrama de Estado para Contrato.
	Salida N°30: Cobrar Publicaciones.	Historia de Usuario N°29: Cobrar por Resumen de Publicación.	Figura 1. 96 Pantalla N° 39: Pantalla de Billetera Autor.  Figura 1. 97 Pantalla N° 40: Pantalla de Publicación Ver Resúmenes Generados.

	Salida N°31: Resumen Publicaciones.	Historia de Usuario N°30: Generar Resumen de Publicación.	Figura 1. 97 Pantalla N° 40: Pantalla de Publicación Ver Resúmenes Generados.
	Salida N°32: Verificar Contrato.	Historia de Usuario N°31: Aceptar/Rechazar Contrato.	Figura 1. 98 Pantalla N° 41: Pantalla de Aprobar/Rechazar Contrato.  Diagrama 1. 16 Diagrama de Estado para Contrato.
<b>Módulo de Reportes</b>	<b>CASOS DE USO</b>		
	<p>32. CU 32 - Generar Reportes de Usuarios.</p> <p>33. CU 33 - Generar Reportes de Publicaciones.</p> <p>34. CU 34 - Generar Reportes de Ingresos y Egresos.</p>		
	<b>SALIDAS DEL SISTEMA</b>	<b>HISTORIAS DE USUARIO</b>	<b>PANTALLAS Y DIAGRAMAS</b>
	Salida N°33: Ver Reportes de Usuarios.	Historia de Usuario N°32: Generar Reportes de Usuarios.	Figura 1. 99 Pantalla N° 42: Pantalla de Reportes (a).  Figura 1. 100 Pantalla N° 43: Pantalla de Reportes (b).
	Salida N°34: Ver Reporte de Publicaciones.	Historia de Usuario N°33: Generar Reportes de Publicaciones.	Figura 1. 99 Pantalla N° 42: Pantalla de Reportes (a).

			Figura 1. 100 Pantalla N° 43: Pantalla de Reportes (b).
	Salida N°35: Ver Reporte de Ingresos y Egresos.	Historia de Usuario N°34: Generar Reportes de Ingresos y Egresos.	Figura 1. 99 Pantalla N° 42: Pantalla de Reportes (a).  Figura 1. 100 Pantalla N° 43: Pantalla de Reportes (b).
<b>Módulo de Configuración</b>	<b>CASOS DE USO</b>		
	35. CU 35 - Configurar Reglas de Negocio. 36. CU 36 - Gestionar Categorías. 37. CU 37 - ABM Roles.		
	<b>SALIDAS DEL SISTEMA</b>	<b>HISTORIAS DE USUARIO</b>	<b>PANTALLAS Y DIAGRAMAS</b>
	Salida N°36: Administrar Reglas de Negocio.	Historia de Usuario N°35: Configurar Reglas de Negocio.	Figura 1. 101 Pantalla N° 44: Gestionar Reglas de Negocios.
	Salida N°37: Administrar Categoría de Publicaciones.	Historia de Usuario N°36: Gestionar Categorías.	Figura 1. 102 Pantalla N° 45: Pantalla de Administrador de Gestionar Categorías.
Salida N°38: Administrar Roles de usuario.	Historia de Usuario N°37: ABM Roles.	Figura 1. 103 Pantalla N° 46: Gestionar Roles y Visualizar Tabla de Roles Disponibles.	

			<p>Figura 1. 104 Pantalla N° 47: Crear Nuevo Rol.</p> <p>Figura 1. 105 Pantalla N° 48: Editar Rol Existente.</p> <p>Figura 1. 106 Pantalla N° 49: Gestionar Usuarios y sus Roles Asignados.</p> <p>Figura 1. 107 Pantalla N° 50: Cambiar Rol a un Usuario Existente.</p>
--	--	--	--

*Tabla 1. 2 Correspondencia entre las salidas del sistema, las historias de usuarios y las pantallas.*

## 5. Modelo de Datos

### 5.1. Diagrama de Clases

El Diagrama de Clases se realiza en inglés para mantener una correspondencia directa con el modelo de datos en la base de datos, y en el desarrollo del sistema. Es importante que se reflejen correctamente estos conceptos, y su entendimiento, ya que representan las entidades principales del sistema. ([VER ANEXO N° 3: DIAGRAMA DE CLASES.](#))

Algunas entidades tienen tres propiedades referentes al tiempo (**fecha**):

- ❖ **created\_at**: Fecha de creación de la entidad (o registro a nivel de base de datos).
- ❖ **update\_at**: Fecha de última modificación. Inicialmente NULL (recién creada).
- ❖ **deleted\_at**: fecha de eliminación. Si es NULL significa que la instancia está aún vigente, si es distinta a NULL significa que esta fue eliminada (*soft-delete* o baja lógica).

En algunos casos, como el de **Plan**, únicamente posee *created\_at* y *deleted\_at*. Esto significa que únicamente puede ser creada y eliminada lógicamente. Así, los planes vigentes tendrán *deleted\_at* = NULL, y los dados de baja algún valor de fecha.

Se tienen dos entidades principales:

#### ❖ **User**

Representa al usuario en sí, este se caracteriza por poder:

- Registrarse en la plataforma, generando una nueva instancia de **User** cuyos campos **username**, **password** y **e-mail** son obligatorios. Tiene relación con **Gender** (género).
- Cuando se registra, se genera la nueva instancia de **User** con *validated* = false. Y una nueva instancia de **UserValidation** con un *code* único generado aleatoriamente, el cual sirve para que valide su e-mail al enviarle un enlace único, el cual incluye este código único y que, al ser clickeado, valida a **User** y elimina la instancia de **UserValidation** ya que no se vuelve a requerir.
- Al ingresar, iniciando sesión, con sus credenciales (usuario y contraseña). La contraseña del usuario es un *hash* criptográfico, no texto plano.
- Un **UserToken** que determina que el usuario ha ingresado a la plataforma (*login*). Lo utilizará para que el sistema autorice todas sus acciones.

- Si olvida la contraseña, se genera una instancia de **PasswordRecovery** con un *code* único generado aleatoriamente. Se envía un e-mail con una URL conteniendo este código. Debe abrirla para cambiar la contraseña. Una vez cambiada, se elimina PasswordRecovery.
- Seguir a otro **User**.
- Posee un **Role** asignado que determina el nivel de acceso que tiene a distintas funcionalidades del sistema. Tiene múltiples **Permission**, uno por cada **Module**, indicando el acceso del rol a cada módulo. Estos permisos se definen en la forma: CREATE, READ, UPDATE y DELETE. Es decir, todos los accesos de CRUD. Cada **Module** está predefinido.
- Agregar una publicación a sus marcadores o favoritos, mediante **PublicationFavorite**. De esta forma se permite acceder en el futuro más rápido y recibir notificaciones de cambios en la publicación.
- Suscribirse a un **Plan** para acceder a contenido premium, mediante **Suscription**. La suscripción requiere que se realice un pago todos los meses. Este pago queda registrado en **SuscriptionPayment**.
- Dar "Me gusta" a una **Publication**, mediante **Like**.
- Calificar una **Publication** mediante **Review**, donde se asigna determinada cantidad de estrellas y un comentario.
- Visualizar una **Publication**, quedando registro en **View**.
- Terminar de leer una obra, quedando registro en **Read**.
- Realizar donaciones a autores, esto es **Donation** en relación con **User** y **Publication**. La donación se puede realizar desde la publicación o el perfil del autor.
- Solicitar un contrato para una obra: **Contract**.
- Tiene relación con un **UserType**. Al momento de registrarse, ingreso con *code* "READER" indicando que es un lector. Al momento de realizar su primera publicación se agrega el **UserType** "AUTHOR". Así, puede tener más de un **UserType** asignado.

#### ❖ **Publication**

Representa una obra, la cual contiene lo siguiente:

- Tiene atributos básicos que funcionan como estadísticas, precalculados, en **PublicationStatistics**:
  - **views**: Cantidad total de visualizaciones. Proviene de **View**.
  - **unique\_views**: Contador de visualizaciones únicas, es decir, discriminando por usuario, ya que un mismo usuario puede haber leído la misma Publicación múltiples veces. Proviene de **View**.

- **stars:** Promedio de estrellas, representa la calificación de la obra. Proviene de **Review**.
  - **likes:** Cantidad de "Me gusta" que tiene la Publicación. Proviene de **Like**.
  - **followers:** Cantidad de usuarios siguiendo la Publicación a través de "Favoritos". Proviene de **PublicationFavorite**.
- Tiene relación con **Image** como imagen de portada.
  - Tiene múltiples relaciones con **Image**, una por cada imagen que compone a la obra.
  - Tiene **Access** definido, que determina el nivel de acceso (free, premium) requerido para acceder a la obra. *Premium* establece que sólo usuarios con **Suscription** para el mes actual tienen acceso al contenido, y a su vez que sólo las obras premium pueden recibir retribución económica mediante **Contract**.
  - Tiene **Category** y **Tag** asignados. Las categorías están preestablecidas en el sistema y las etiquetas pueden ser creadas por el mismo autor.
  - Tiene **PublicationStatus**, el cual puede cambiar con el tiempo mediante **PublicationStatusHistory**.

Estos estados son:

- *Draft:* Borrador, antes de realizar la publicación como tal.
  - *waiting-approval:* Una vez publicada la Publicación, en espera de aprobación por un Administrador "Gestor de contenido".
  - *Published:* Publicación aceptada por un gestor de contenido, aparece en el catálogo general.
  - *Rejected:* La Publicación es rechazada por el Administrador "Gestor de contenido".
- Periódicamente el sistema genera **Summary**, un resumen de estadísticas para una **Publication** en un período determinado. En esta también se detalla el monto de dinero asignado a ese resumen de acuerdo con la cantidad de visualizaciones generadas por la **Publication** y los ingresos de la organización en ese período, representado en **Invoice**; entre otros criterios determinados por el algoritmo de cálculo de pago a autores utilizado, el cual será una estrategia dentro del sistema. El mismo autor puede generar un **Summary** para sus publicaciones de forma manual, al hacer click en un botón del panel de gestión. **Summary** es de gran

importancia para realizar el pago a autores de acuerdo con lo generado por sus obras.

Además de las dos entidades anteriores, existe una tercera de importancia relativamente alta. Esta es la que establece el pacto entre la organización, y el autor para que este último distribuya contenido en la plataforma a cambio de una remuneración.

#### ❖ **Contract**

Representa un contrato real entre un autor (**User**) y la organización para una de sus obras (**Publication**). Es de especial interés por la intervención de distintos roles en su mantenimiento. Una publicación puede tener un único contrato vigente al mismo tiempo, aunque se almacena el histórico de contratos de la publicación.

Tiene:

- Un **ContractStatus** asignado, que contempla sus variaciones mediante **ContractStatusHistory**. Este estado se corresponde con *requested* (cuando el contrato es requerido por el autor para una de sus obras), *approved* o *rejected* (una vez que se analiza por el personal de la organización y este la rechaza o aprueba, según sea el caso).
- **ContractKind**, o más bien, tipo de contrato. Esto contempla contratos especiales o de exclusividad con determinadas publicaciones. Por ejemplo, algunas obras adheridas a la plataforma para generar comunidad de autores con los que se realiza pactos por fuera del sistema.

En cuanto al pago a autores, este depende de los resúmenes generados: **Summary**. Los autores cuyas **Publication** tienen **Contract**, y cumplen con un mínimo de visualizaciones preestablecidas en **Summary** (*unique\_views*), tiene la posibilidad de generar un **Payment**, el cual representa el pago al autor por parte de la plataforma. Este **Payment** se realiza sumando los montos de los últimos **Summary** (*payment\_amount*) no pagados (sin ninguna relación con **Payment**).

## Propiedades Importantes

A continuación, se detallan las clases y algunos atributos considerados como más importantes:

#### ❖ **User**



- **is\_author:** Es una propiedad precalculada. No existe una diferencia real entre lo que es un usuario lector y un usuario autor, más que la cantidad de obras publicadas que tenga un usuario. Es decir, un usuario con más de una *Publication* se puede considerar autor. Sin embargo, se utiliza esta propiedad dinámica (computada) para obtener un acceso inmediato al valor que indica si es o no autor. Al considerar esta propiedad, se facilita al desarrollador la consulta por medio de "user.is\_author", y, además, en un futuro puede abarcar otros criterios para considerar a un usuario como autor, simplemente cambiando la lógica que establece esta propiedad.
- **password:** Vale la pena aclarar, nuevamente, que la contraseña del usuario no se almacena en texto plano, sino que se utiliza un *hash* criptográfico.

#### ❖ Publication

- **size:** Sumatoria del tamaño de las imágenes.
- **review\_only\_subscribers:** Un autor puede definir si cualquier usuario puede calificar su obra o solamente aquellos que están suscriptos. Por defecto: false.

#### ❖ PublicationStatistics

- **views:** Cantidad de visualizaciones de una publicación. Simplemente cuenta las entradas (instancias) de **View**.
- **unique\_views:** Cantidad de visualizaciones discriminando por usuario. Se obtiene al sumar las entradas (instancias) de **View**, sumando +1 por usuario, es decir, se cuentan una única vez las instancias repetidas para el mismo usuario y publicación.
- **stars:** Promedio de estrellas. Se obtiene al sacar el promedio de todas las entradas de **Review.stars**.
- **likes:** Cantidad de instancias de **Like**, para la publicación en cuestión.
- **followers:** Cantidad de seguidores, esto es, cantidad de entradas en **PublicationFollow** en relación con la publicación.

#### ❖ Summary

Algunas propiedades son las mismas que en *PublicationStatistics*, pero discriminadas o calculadas en función de un período distinto, por ejemplo, mensualmente, para reconocer cuánto debe pagarse al autor.

- **payment\_amount**: Monto, en dinero, que corresponde al usuario acorde a la cantidad de visualizaciones generadas por alguna de sus publicaciones en particular. Se calcula a partir de **Income**, instancia que almacena los ingresos de la organización de acuerdo con las suscripciones en el mes.
- Así, calcular este valor, se podría obtener así:

$$\text{porcentaje\_visualizaciones} = \text{mi\_publicacion.unique\_views} / \text{total\_visualizaciones}$$
$$\text{Summary.payment\_amount} = \text{porcentaje\_visualizaciones} * \text{Income.payment\_amount}$$

Se calcula el porcentaje de visualizaciones generadas por el usuario en función del total de visualizaciones que ha recibido la plataforma (todas las publicaciones). De acuerdo con el monto destinado a pagar a autores (un porcentaje del ingreso total) se obtiene el monto total que se pagará al autor, considerando el rango de fechas.

- **from**: "Fecha desde".
- **to**: "Fecha hasta". Considera un rango.

#### ❖ **Income**

Ingresos de la organización en un período dado.

- **total**: Ingresos totales. Sumatoria de todos los **SubscriptionPayment**, es decir, los pagos de suscripciones realizados en el mes.
- **payment\_amount**: Porcentaje del ingreso total derivado a pagar a autores por sus publicaciones.

#### ❖ **Configuration**

Es una clase particular, ya que almacena información de configuración y parámetros utilizados por algunos algoritmos y funcionalidades.

- **min\_payment\_amount**: Monto mínimo para que un autor pueda cobrar por una de sus publicaciones, en **Summary**. La acumulación de resúmenes sin cobrar ( $\sum \text{Summary.payment\_amount}$ ) debe ser mayor o igual que **Configuration.min\_payment\_amount** para poder cobrar todos esos resúmenes en un único **Payment**. Imponer un mínimo es importante para asegurar que sólo cobren aquellos autores que se están haciendo populares en la plataforma.

Además, realizar transferencias de dinero significativas, apaleando el interés impuesto por los servicios de pago utilizados.

- **min\_donation:** Monto mínimo de donación. Se utilizan servicios externos que gestionan el flujo de dinero y medios de pago, como Mercado Pago.

Algunos autores pueden obtener un resumen cuyo monto de pago sea de \$10 únicamente, y debido a que estas plataformas de cobro externas cobran un interés por transacción, sería poco beneficioso para ambas partes realizar una transferencia de solamente \$10.

## 5.2. Diagrama de Estados

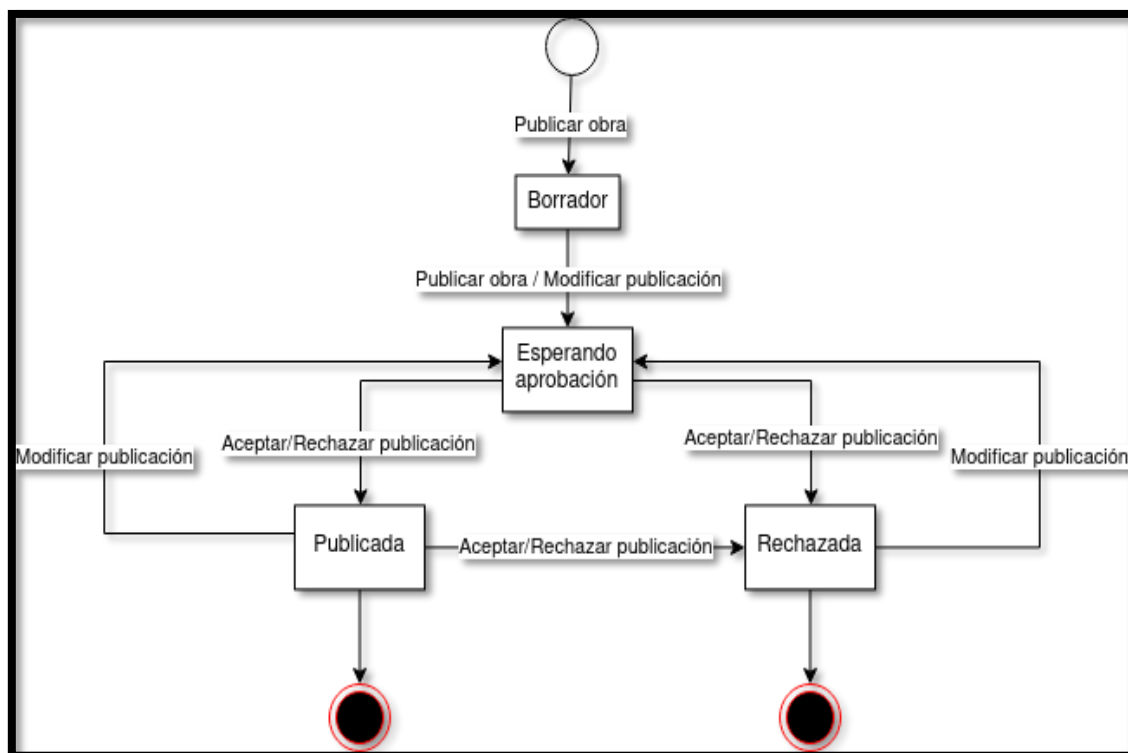


Diagrama 1. 15 Diagrama de Estado para Publicación.

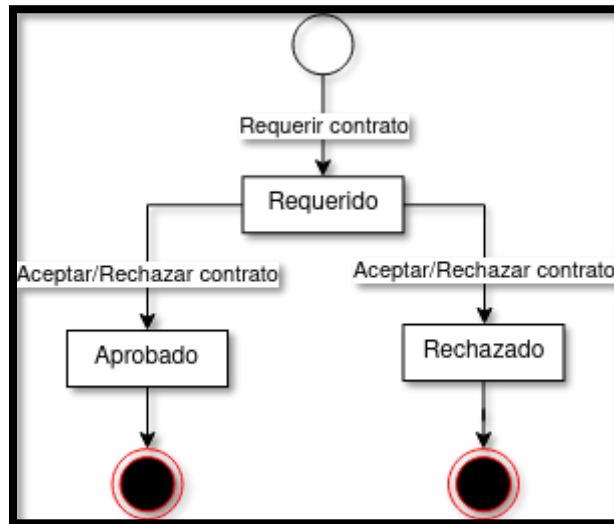


Diagrama 1. 16 Diagrama de Estado para Contrato.

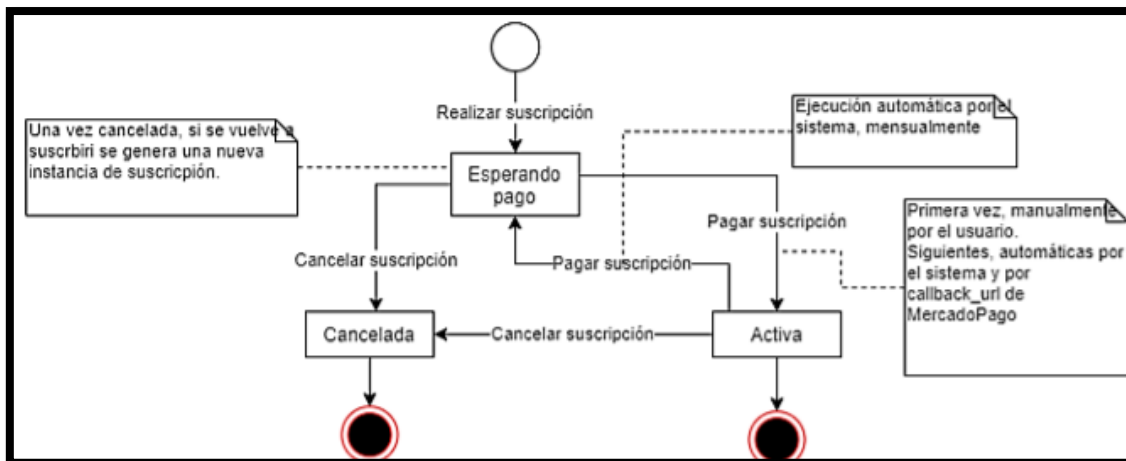


Diagrama 1. 17 Diagrama de Estado para Suscripción.

## Desarrollo e Implementación

### 1. Programación y Documentación

#### 1.1. Tecnologías Utilizadas

Las tecnologías que se utilizan en el desarrollo de la plataforma *Omics* se ven reflejadas en la siguiente tabla:

TECNOLOGÍAS	
Herramientas	Descripción
<b>Front-end</b>	
<b>TypeScript</b>	Lenguaje de programación. Similar a JavaScript pero fuertemente tipado. Transpila directamente a JavaScript por lo que beneficia el desarrollo en cuanto a seguridad de los datos y comunicación de los desarrolladores.  <a href="https://www.typescriptlang.org/">HTTPS://WWW.TYPESCRIPTLANG.ORG/</a>
<b>Angular</b>	Framework web enfocado en el desarrollo reactivo. Cubre la creación de componentes (lógica, datos, renderizado).  <a href="https://angular.io/">HTTPS://ANGULAR.IO/</a>
<b>Back-end</b>	
<b>PostgreSQL</b>	Motor de base de datos principal. Basado en SQL por lo que provee esquemas de datos seguros, con referencia e intersección entre tablas.  <a href="https://www.postgresql.org/">HTTPS://WWW.POSTGRESQL.ORG/</a>
<b>Redis</b>	Servicio de caché distribuida.  Almacén de estructura de datos en memoria. Se utiliza como base de datos temporal, como caché y, además, puede utilizar como <i>message broker</i> . Soporta estructura de datos como cadenas de texto, hashes, listas, sets y sets ordenados, además de rangos de datos y operaciones atómicas sobre los mismos.

	<p>Tiene integrada la replicación automática para utilizar múltiples instancias del servicio con datos distribuidos en clústeres.</p> <p><a href="https://redis.io/">HTTPS://REDIS.IO/</a></p>
<p><b>RabbitMQ</b></p>	<p><i>Message broker.</i> Se encarga de distribuir eventos y mensajes generados por el sistema a todos los módulos.</p> <p>Los módulos (bounded contexts) de la plataforma se integran principalmente de forma eventual. Esto es, cada módulo publica eventos para notificar los cambios en los datos y los demás pueden escuchar o suscribirse a dichos eventos para actualizar sus datos pertinentes.</p> <p>RabbitMQ permite este modelo de publicación y suscripción para eventos en forma de mensajes. Además de permitir enrutar dichos eventos a determinados tópicos, formando así una red de colas de eventos listos para consumirse por la plataforma.</p> <p><a href="https://www.rabbitmq.com/">HTTPS://WWW.RABBITMQ.COM/</a></p>
<p><b>Docker</b></p>	<p>Tecnología de contenedores que utiliza una abstracción directa sobre el kernel de Linux para ejecutar estos contenedores como si se tratase de sistemas nativos. Una mejora frente a la virtualización tradicional.</p> <p>Aporta un gran beneficio en cuanto a la integración y uso de servicios externos, como motores de base de datos, caché, message brokers, etc. La instanciación de servicios consta únicamente de levantar contenedores a partir de imágenes predefinidas provistas por los desarrolladores oficiales de cada servicio. Esto abre las puertas para escalar a futuro en caso de que el sistema crezca exponencialmente en cuanto a cantidad de usuarios y requerimientos de rendimiento.</p> <p>La arquitectura principal del sistema se puede definir completamente con algunos archivos de configuración de Docker para levantar todos los servicios, incluidos el back-end y front-end, con un único comando de consola.</p> <p><a href="https://www.docker.com/">HTTPS://WWW.DOCKER.COM/</a></p>

<p><b>Rust</b></p>	<p>Lenguaje de programación de última generación, fuertemente tipado, con reglas estrictas para evadir cualquier tipo de error generado por el mal uso de la memoria.</p> <p>No posee garbage collector. Se basa únicamente tres reglas que el programador debe respetar y que derivará en sistemas de gran performance:</p> <p>Sólo puede existir un único dueño de determinado dato o variable.</p> <p>Pueden existir múltiples referencias inmutables a una variable al mismo tiempo.</p> <p>Puede existir una única referencia mutable a una variable al mismo tiempo.</p> <p>Gracias a esas reglas, no es necesario garbage collector. Y así se obtienen sistemas de gran performance al nivel de sistemas escritos en lenguaje C pero de alta calidad y con cero errores de memoria o administración de hilos.</p> <p>Rust es un lenguaje que permite programar sistemas nativos (compilación a binario estáticamente enlazado) de gran rendimiento, con restricciones de mutabilidad en los datos.</p> <p>Se ha elegido este lenguaje de programación emergente ya que provee grandes beneficios en cuanto a seguridad gracias a sus restricciones de mutabilidad, y a su vez brinda rendimiento cercano al lenguaje máquina ya que no hace uso de garbage collector ni de virtual machine. Tiene grandes ventajas en cuanto a lógica del sistema mitigando errores humanos y carga del sistema.</p> <p><a href="https://www.rust-lang.org/">HTTPS://WWW.RUST-LANG.ORG/</a></p>
<p><b>Integración y Soporte</b></p>	
<p><b>GIT + GitHub</b></p>	<p>GIT como herramienta para colaborar sobre la misma base de código en repositorios y tener seguimiento de cada cambio realizado sobre el mismo.</p> <p>Permite la integración de cambios sobre el código fuente y realizar revisiones de código periódicamente.</p>

	<p>GitHub es el servicio de repositorios más grande del mundo. Permite la interacción entre desarrolladores en una interfaz web amigable.</p> <p>Aloja el repositorio central del proyecto.</p> <p><a href="https://github.com/">HTTPS://GITHUB.COM/</a></p>
<b>GitHub Actions</b>	<p>GitHub, además, provee un servicio de integración continua: GitHub Actions. Gracias a este se pueden realizar builds periódicas y automáticas del sistema, tanto del backend como del front-end, y desplegar dichas builds en los servidores correspondientes sin intervención de personas.</p> <p>Se utilizar, también, para correr todos los tests antes de realizar el despliegue para validar el correcto funcionamiento del sistema en términos de lógica y validación de datos.</p> <p><a href="https://github.com/features/actions">HTTPS://GITHUB.COM/FEATURES/ACTIONS</a></p>
<b>Insomnia</b>	<p>Documentación de la API. Genera una interfaz web para realizar pruebas.</p> <p><a href="https://insomnia.rest/">HTTPS://INSOMNIA.REST/</a></p>
<b>Servidores</b>	
<b>VPS (Virtual Private Server)</b>	<p>DigitalOcean es uno de los mejores proveedores de servidores privados virtuales en cuestión de precio-calidad.</p> <p>Principalmente permite la contratación de VPS basados en GNU/Linux. Estos pueden escalar (requerir más recursos, como procesadores, memoria RAM y espacio de disco) fácilmente.</p> <p><a href="https://www.digitalocean.com/">HTTPS://WWW.DIGITALOCEAN.COM/</a></p>
<b>Amazon S3</b>	<p>Alojamiento de archivos estáticos.</p> <p>Para el proyecto se elige esta opción para almacenar las imágenes de las publicaciones.</p> <p><a href="https://aws.amazon.com/es/s3/">HTTPS://AWS.AMAZON.COM/ES/S3/</a></p>
<b>Servicios Externos</b>	
<b>API de Mercado Pago</b>	<p>Integración para la realización de pagos mediante Mercado Pago.</p>



	<p>Este provee una API amigable al desarrollador para gestionar todo tipo de pagos. A su vez, hace que no sea necesario manejar datos sensibles del usuario como números de tarjeta de crédito/débito, dirección, entre otros, en el sistema, sino que se delegan a este servicio externo.</p> <p><a href="https://www.mercadopago.com.ar/developers">HTTPS://WWW.MERCADOPAGO.COM.AR/DEVELOPERS</a></p>
--	---

Tabla 1. 3 Tecnologías utilizadas en el desarrollo

## 1.2. Plantilla de Documentación del código

Se define la plantilla a utilizar para la documentación del código fuente de la funcionalidad seleccionada:

PLANTILLA DE DOCUMENTACIÓN	
<b>Nombre del Sistema</b>	Nombre del sistema elegido para el proyecto.
<b>Nombre del módulo del cual proviene el Caso de Uso</b>	Nombre del Módulo al que pertenece el Caso de Uso que se va a referenciar.
<b>Nombre del Caso de Uso</b>	Nombre del Caso de Uso relacionado al módulo.
<b>Actores</b>	Nombre del actor que se relaciona con el caso de uso.
<b>Casos de Prueba utilizados</b>	Nombre del caso de Prueba utilizados en este Caso de Uso.
<b>Fecha Inicio programación</b>	Se coloca la fecha en la que se inicia la programación.
<b>Fecha Fin programación</b>	Se coloca la fecha en la que se finaliza la programación.
<b>Clases involucradas</b>	Nombrar las clases que participan y se relacionan con el caso de uso.
<b>Nombre y Apellido del programador</b>	Escribir el nombre y apellido del programador que ha realizado el código del caso de uso.
<b>Nombre y Apellido del usuario que aprobó</b>	Escribir el nombre y apellido del usuario que aprobó el código realizado del caso de uso.
<b>Código Fuente – Front end</b>	
Porción del código del Front-end desarrollado del caso de uso.	
<b>Código Fuente – Back end</b>	

Porción del código del Back-end desarrollado del caso de uso.

*Tabla 1. 4 Plantilla base de documentación de código.*

### 1.3. Documentación del Código

Se muestra en la siguiente planilla la documentación del Front-end y Back-end del Caso de Uso "Publicar Obra" que es uno de los principales en el funcionamiento de la plataforma.

PLANTILLA DE DOCUMENTACIÓN	
<b>Nombre del Sistema</b>	Omics
<b>Nombre del módulo del cual proviene el Caso de Uso</b>	Módulo de Publicación de Obras
<b>Nombre del Caso de Uso</b>	Publicar Obra
<b>Actores</b>	Autor
<b>Casos de Prueba utilizados</b>	<ul style="list-style-type: none"> <li>❖ <b>Caso de Prueba de Lógica de los Módulos Principales 02:</b> Autor adjunta la portada al publicar una obra.</li> <li>❖ <b>Caso de Prueba de integración entre los Módulos del Sistema 03:</b> Publicar y visualizar el contenido de la obra.</li> <li>❖ <b>Caso de Prueba de Seguridad por Niveles de Usuario 03:</b> Autor puede visualizar reportes y detalles de sus publicaciones.</li> <li>❖ <b>Caso de Prueba de Carga 03:</b> Peticiones por segundo de la API REST al endpoints de catálogo.</li> <li>❖ <b>Caso de Prueba de Carga 04:</b> Peticiones por segundo de la API REST al endpoints de catálogo.</li> </ul>
<b>Fecha Inicio programación</b>	21/07/2020
<b>Fecha Fin programación</b>	22/09/2020
<b>Clases involucradas</b>	<ul style="list-style-type: none"> <li>❖ De dominio:               <ul style="list-style-type: none"> <li>➤ Publication</li> <li>➤ Category</li> <li>➤ Image</li> <li>➤ Author (proyección de User)</li> </ul> </li> </ul>

<b>Nombre y Apellido del programador</b>	Alan Boglioli y Julián Muñoz Velázquez
<b>Nombre y Apellido del usuario que aprobó</b>	Alan Boglioli
<b>Código Fuente – Front end</b>	
<b>HTML</b>	
<pre> &lt;div class="perfil-editar-body"&gt;   &lt;div class="side"&gt;&lt;/div&gt;   &lt;div class="center"&gt;     &lt;form autocomplete="on" [formGroup]="formProfile" *ngIf="this.userData" &gt;       &lt;span style="display: flex; flex-direction: column; height: 100%;"&gt;         &lt;div class="header"&gt;           &lt;div class="header-items"&gt;             &lt;h1 class="profile-name"&gt;{{ this.userData.username   uppercase }}&lt;/h1&gt;           &lt;/div&gt;           &lt;!-- #region Imagen avatar perfil--&gt;           &lt;div class="avatar-edit"             (click)="uploadImageAvatar()"&gt;             &lt;fa-icon [icon]="faEditField" size="2x"&gt;&lt;/fa-icon&gt;             &lt;img *ngIf="!isProfileAvatarImageExists"               src="assets/images/generic/avatar196x196.jpg"               alt="omics-avatar-profile"               class="avatar"&gt;             &lt;img *ngIf="isProfileAvatarImageExists"               [src]="profileAvatarImage"               alt="omics-avatar-profile"               class="avatar"&gt;           &lt;/div&gt;           &lt;!-- #endregion --&gt;         &lt;/div&gt;       &lt;/span&gt;     &lt;/form&gt;   &lt;/div&gt; &lt;/div&gt;                 </pre>	

```
</div>
<div class="body">
  <div class="button-actions">
    <button mat-flat-button
      color="accent"
      class="button-white-borders"
      (click)="onGuardarCambios()">
      Guardar Cambios
    </button>

    <!-- #region Botón suscripción -->
    <span *ngIf="!readerIsSubscribed">
      <button mat-flat-button
        color="primary"
        class="button-white-borders subscription-button"
        (click)="onSuscribirse()">
        Suscribirse
      </button>
    </span>
    <span *ngIf="readerIsSubscribed">
      <button mat-flat-button
        color="primary"
        class="button-white-borders subscription-button"
        (click)="onSuscribirse()">
        Desuscribirse
      </button>
    </span>
    <!-- @endregion -->
```

```
</div>
<!-- Apellido y nombre -->
<div class="box-row">
  <!-- Block: campo nombre -->
  <div class="block-field">
    <div class="input-field-row">
      <div class="title-and-error">
        <p class="field-name">Nombre: </p>
        <!-- mensaje de error -->
        <p *ngIf="nombreNovalido"
          class="form-text text-danger">
          Ingrese un nombre de al menos 2 caracteres
        </p>
      </div>
    </div>
  </div>
  <!-- Input text -->
  <input class="form-control"
    type="text"
    placeholder="Nombre"
    formControlName="name"
    [class.is-invalid]="nombreNovalido">
  </div>
  <!-- Block: campo apellido -->
  <div class="block-field">
    <div class="input-field-row">
      <div class="title-and-error">
        <p class="field-name">Apellido: </p>
        <!-- mensaje de error -->
```

```
<p *ngIf="apellidoNovalido"
  class="form-text text-danger">
  Ingrese un apellido de al menos 2 caracteres
</p>
</div>
</div>
<!-- Input text -->
<input class="form-control"
  type="text"
  placeholder="Apellido"
  formControlName="lastname"
  [class.is-invalid]="apellidoNovalido">
</div>
</div>
<!-- Email -->
<div class="box-row">
  <!-- Block: campo email -->
  <div class="block-field">
    <div class="input-field-row">
      <div class="title-and-error">
        <p class="field-name">Correro: </p>
        <!-- mensaje de error -->
        <p *ngIf="correoNovalido"
          class="form-text text-danger">
          Ingrese un correo válido
        </p>
      </div>
    </div>
  </div>
</div>
```

```

<!-- Input text -->
<input class="form-control"
      type="email"
      placeholder="Correo electrónico"
      formControlName="email"
      [class.is-invalid]="correoNovalido">
</div>
<!-- Block: espacio auxiliar -->
<div class="block-field">
</div>
</div>
<!-- Biografía -->
<div class="box-row">
  <div class="block-field">
    <div class="input-field-row">
      <div class="title-and-error">
        <p class="field-name">Descripción: </p>
        <!-- mensaje de error -->
        <p *ngIf="biografiaNovalido"
          class="form-text text-danger">
          Requiere que sea menos de 256 caracteres.
        </p>
      </div>
    </div>
  </div>
  <mat-form-field appearance="fill" class="text-area">
    <mat-label>{{ biografiaLenght }} / 256</mat-label>
    <textarea matInput
      formControlName="biography"> </textarea>

```

```

        </mat-form-field>
    </div>
</div>
<!-- Genero -->
<div class="box-row">
    <div class="block-field">
        <div class="input-field-row">
            <div class="title-and-error">
                <p class="field-name">Género: </p>
            </div>
        </div>
        <mat-radio-group formControlName="gender">
            <mat-radio-button *ngFor="let genero of generosList"
                color="primary"
                [value]="genero.value">
                {{ genero.name }}
            </mat-radio-button>
        </mat-radio-group>
    </div>
</div>
<!-- Fecha Nacimiento -->
<div class="box-row">
    <div class="block-field">
        <div class="input-field-row">
            <div class="title-and-error">
                <p class="field-name">Fecha de Nacimiento: </p>
            </div>
        </div>
    </div>
</div>

```



```

<mat-form-field appearance="fill" class="matDate">
  <mat-label>Selecciona una fecha</mat-label>
  <input matInput
    [max]="maxDateBirthday"
    [matDatepicker]="picker"
    FormControlName="birthdate"
    (dateChange)="convertDateToRFC3339($event.value)">
  <mat-datepicker-toggle matSuffix [for]="picker"></mat-datepicker-
toggle>
  <mat-datepicker #picker></mat-datepicker>
</mat-form-field>
</div>
<!-- Block: cambiar contraseña -->
<div class="block-field" style="justify-content: center;">
  <button mat-flat-button
    color="primary"
    class="button-white-borders"
    (click)="onChangePassword()">
    Cambiar contraseña
  </button>
</div>
</div>
</div>
<!-- Eliminar cuenta -->
<div style="display: flex; justify-content: center; margin-bottom: 1em;">
  <button mat-flat-button color="warn"
    class="button-white-borders"
    style="width: 50%;"
    (click)="onEliminarCuenta()">

```

```

        Eliminar Cuenta
    </button>
</div>
</span>
</form>
</div>
<div class="side"></div>
</div>
<!-- Swal Message -->
<swal #swalDataInvalid
    title="Datos incompletos o erroneos"
    text="Complete correctamente los datos necesarios"
    icon="error"
    [focusConfirm]="true">
</swal>
<!-- Confirm Delete Account -->
<swal #swalDeleteAccount icon="info"
    title="¿Estas seguro que deseas eliminar tu cuenta?"
    [showConfirmButton]="false"
    [showCloseButton]="true"
    (close)="formDeleteAccount.reset()">
    <form *swalPortal [formGroup]="formDeleteAccount">
        <!-- Mensaje de alerta -->
        <div class="alert alert-danger">
            <small>Tus obras, datos y publicación dejarán de ser visibles en la
plataforma.</small>
        </div>
        <div class="alert alert-info">
            <small>Ingresar contraseña para confirmar.</small>

```

```

</div>

<!-- Contraseña ingresar para eliminar -->
<div class="block-field" style="margin: 0em;">
  <div class="input-field-row">
    <div class="title-and-error">
      <p class="field-name">Contraseña: </p>
    </div>
  </div>
</div>

<!-- Input text -->
<div class="input-with-icon">
  <input class="form-control"
    [type]="passwordProperties.type"
    placeholder="Contraseña"
    formControlName="password">
  <fa-icon class="button"
    (click)="showPassword( passwordProperties.visible )"
    [icon]="(passwordProperties.visible) ? faEyeSlash : faEyeOpen"
    size="1x">
  </fa-icon>
</div>
</div>

<!-- Botón confirmar eliminar cuenta -->
<div style="display: flex; justify-content: center; margin: 1em;">
  <button mat-flat-button color="warn"
    class="button-white-borders"
    style="width: 50%;"
    [disabled]="formDeleteAccount.invalid"
    (click)="eliminarCuentaConfirm()">

```

*Confirmar Eliminar Cuenta*

*</button>*

*</div>*

*</form>*

*</swal>*

### **TYPESCRIPT / JAVACRIPT**

```
import { Component, OnInit, ViewChild } from '@angular/core';
import { Router, ActivatedRoute } from '@angular/router';
import { forkJoin, Observable } from 'rxjs';
import { FormGroup, FormBuilder, Validators, FormArray, FormControl } from
'@angular/forms';
import { MatCheckbox } from '@angular/material/checkbox';
import { COMMA, ENTER } from '@angular/cdk/keycodes';
import { faPlusCircle, faTimesCircle, faBookOpen, faTrashAlt } from
'@fortawesome/free-solid-svg-icons';
import { AuthService } from '../././domain/services/auth.service';
import { FileService } from '../././domain/services/file.service';
import { DropdownDataObrasService } from '../././services/dropdown-data-
obras.service';
import { NgxSpinnerService } from 'ngx-spinner';
import { IPublication, IPage } from '../././domain/models/publication';
import { IDropdownItem } from '../././models/dropdown-item.interface';
import { MatChipInputEvent } from '@angular/material/chips';
import { SwalComponent } from '@sweetalert2/ngx-sweetalert2';
import { PublicationService, IUpdatePagesCommand, IGetByIdResponse,
IUpdateCommand, IReadResponse } from '../././domain/services/publication.service';
import { ICreateCommand, CollectionService } from
'../././domain/services/collection.service';
import { SweetAlertGenericMessageService } from '../././services/sweet-alert-generic-
message.service';
```

```
import { ICollection } from '../../domain/models/collection';
import Swal from 'sweetalert2';
export interface IPageForm {
  number: number;
  image: string;
  thumbaillImage: string;
  url: string;
}
export interface IPageAndNumber {
  number: number;
  url: string;
}
@Component({
  selector: 'app-publication-new-edit',
  templateUrl: './publication-new-edit.component.html',
  styleUrls: ['./publication-new-edit.component.scss']
})
export class PublicationNewEditComponent implements OnInit {
  @ViewChild('formDataInvalid') private swalFormDataInvalid: SwalComponent;
  @ViewChild('formDataValid') private swalFormDataValid: SwalComponent;
  @ViewChild('formSketchValid') private swalFormSketchValid: SwalComponent;
  @ViewChild('formEditPublishValid') private swalFormEditPublishValid:
  SwalComponent;

  // FontAwesome Icon
  public faPlus = faPlusCircle;
  public faCloseCircle = faTimesCircle;
  public faBoookOpen = faBookOpen;
  public faDelete = faTrashAlt;

  // Usados para Forms
```

```
public formPublication: FormGroup;
public collectionList: IDropdownItem[];
public portadaImage = {
  thumbnail: null,
  url: null
};
public categoryList: IDropdownItem[];
public tagsList: string[] = [];
// Otros
public isToEdit: boolean;
private isToSketch: boolean;
private publicationToEditId: string;
public chipTagsKeysCodes: number[] = [ENTER, COMMA]; // Usado para los tags
constructor(
  private router: Router,
  private activateRoute: ActivatedRoute,
  private authService: AuthService,
  private fb: FormBuilder,
  private spinnerService: NgxSpinnerService,
  private dropdownDataObrasService: DropdownDataObrasService,
  private fileServ: FileService,
  private publicationService: PublicationService,
  private collectionService: CollectionService,
  private sweetAlertGenericService: SweetAlertGenericMessageService,
) {}
ngOnInit(): void {
  this.isToSketch = false;
}
```

```
    this.authService.authStart();
    this.buildForms();
    this.setSubscriptionData();
  }

  public backToDeskboard(): void {
    this.router.navigate(['/deskboard']);
  }

  private buildForms(): void {
    this.formPublication = this.fb.group({
      name: ['', [ Validators.required, Validators.minLength(5) ] ],
      cover: ['', Validators.required ],
      collectionArray: this.fb.array([]),
      synopsis: [ '', [ Validators.required, Validators.minLength(5),
Validators.maxLength(512) ] ],
      category_id: [ '', Validators.required ],
      tags: [ null ],
      pagesList: this.fb.array([])
    });
  }

  public setSubscriptionData(): void {
    this.spinnerService.show();

    const observableList =
[ this.dropdownDataObrasService.getAllCollectionDropdownDataById( this.authService.getIdUser() ),
  this.dropdownDataObrasService.getAllCategoryDropdown()
];

    forkJoin( observableList ).subscribe( ([ dataCollection, dataCategory ] ) => {
      this.collectionList = dataCollection;
      this.categoryList = dataCategory;
    });
  }
}
```

```
this.activateRoute.params.subscribe(  
  (params: any) => {  
    this.publicationToEditId = params.id;  
    // Si no existe el id, es una nueva publicación, sino se busca con el ID la  
publicación  
    if ( this.publicationToEditId === undefined ) {  
      this.isToEdit = false;  
    } else {  
      this.isToEdit = true;  
      this.getPublicationToEdit(this.publicationToEditId);  
    }  
    this.spinnerService.hide();  
  });  
});  
}  
  
private getPublicationToEdit( publicationId: string): void {  
  this.spinnerService.show();  
  this.publicationService.getById(publicationId).subscribe(  
    (res: IGetByIdResponse) => {  
      this.spinnerService.hide();  
      const publicationEdit: IPublication = res.publication;  
      this.publicationService.getCollections(publicationId).subscribe(  
        (collectionRes) => {  
          this.setFormPublicationByObject(publicationEdit, collectionRes.items);  
        },  
        (err: Error) => {  
          console.error(err);  
          this.spinnerService.hide();  
        }  
      )  
    }  
  )  
}
```



```
);  
},  
(err: Error) => {  
  
    console.error(err);  
    this.spinnerService.hide();  
}  
);  
}  
  
private setFormPublicationByObject( publicationObject: IPublication, collectionList:  
ICollection[] ): void {  
    this.spinnerService.show();  
    setTimeout(() => {  
        this.spinnerService.hide();  
    }, 2000); // 5 segundos  
    this.formPublication.reset({  
        name: publicationObject.name,  
        cover: publicationObject.cover,  
        synopsis: publicationObject.synopsis,  
        category_id: publicationObject.category_id,  
    });  
    // Asignar las colecciones  
    if ( collectionList.length > 0 ) {  
        collectionList.forEach( (collection: ICollection) => {  
            this.collectionArrayCheck.push(  
                new FormControl(collection.id)  
            );  
        });  
    }  
}
```

```
// Manejo de portada
this.portadalmage.thumbnail = publicationObject.cover;
this.portadalmage.url = publicationObject.cover;

// Asignar tags
publicationObject.tags.forEach( (tag: string) => {
    this.tagsList.push(tag);
});
this.formPublication.get('tags').setValue(this.tagsList);

// Asignar pagesList
publicationObject.pages.forEach( (page: IPage) => {
    const pageAux = this.newPage( page );
    this.pagesList.push( pageAux );
});
}

// Generales
public uploadImagePortada(): void {
    // Crear elemento input de tipo 'file' para poder manejarlo desde el botón que lo llama
    const inputFileElement = document.createElement('input');
    inputFileElement.type = 'file'; // Nota: Solo uno a la vez, para varios:
inputFileElement.multiple = multiple
    inputFileElement.accept = '.png, .jpg, .jpeg';
    inputFileElement.click();

    // Definir la función del llamado al hacer click (cuando realiza un cambio)
    inputFileElement.onChange = ( event: any ) => {
        const fdImage: FormData = new FormData();
        const imagePortada = event.target.files[0];
        this.spinnerService.show();

        // #region Cargar para previsualizar en pantalla
```

```
const reader = new FileReader();

let tempCoverThumbail;

reader.onload = (eventReader: any) => {
  tempCoverThumbail = eventReader.target.result;
};

reader.readAsDataURL(imagePortada);

//#endregion

fdImage.append('image', imagePortada, imagePortada.name);

this.fileServ.upload(fdImage).subscribe(
  (res: any) => {
    this.portadaImage.thumbail = tempCoverThumbail;
    this.portadaImage.url = res.files[0].url;
    this.formPublication.get('cover').setValue(this.portadaImage.url);
    this.spinnerService.hide();
  }, (err: Error) => {
    console.error(err);
    this.spinnerService.hide();
  }
);

}

public guardarBorrador(): void {
  this.isToSketch = true;
  this.submitPublicationForm();
}

public onPublicationDelete(): void {
  Swal.fire({
    title: 'Eliminar publicación',
```

```
text: '¿Estas seguro ',
icon: 'warning',
showCancelButton: true,
focusCancel: true,
confirmButtonColor: '#FC4850',
cancelButtonColor: '#3085d6',
confirmButtonText: 'Sí, eliminar',
cancelButtonText: 'No'
}).then((result) => {
  if (result.isConfirmed) {
    this.deletePublicationConfirm();
  }
});
}

private deletePublicationConfirm(): void {
  this.spinnerService.show();
  this.publicationService.delete( this.publicationToEditId ).subscribe(
    (res: any) => {
      Swal.fire(
        'Eliminado con éxito',
        ` ${ this.formPublication.get('name').value } se ha eliminado. `,
        'success'
      ).then((result) => {
        if (result.isConfirmed) {
          this.backToDeskboard();
        }
      });
    }
  );
  this.spinnerService.hide();
}
```

```
    },  
    (err: Error) => {  
        this.sweetAlertGenericService.showAlertError('Ha ocurrido un problema al  
eliminar la publicación', 'Error Servidor');  
        console.error(err);  
        this.spinnerService.hide();  
    }  
);  
}  
  
//#region Realizar publicación  
public submitPublicationForm(): void {  
    this.formPublication.get('tags').setValue(this.tagsList);  
    // Reducir descripción los espacios vacios que pueda tener al final  
    const description = this.formPublication.get('synopsis');  
    this.formPublication.get('synopsis').setValue(description.value.trim());  
    if ( this.formPublication.invalid ) {  
        if ( this.formPublication.get('cover').invalid ) {  
            this.sweetAlertGenericService.showAlertError('Se requiere una portada para crear  
la publicación');  
        } else {  
            this.swalFormDataInvalid.fire();  
        }  
    }  
    return Object.values( this.formPublication.controls ).forEach( control => {  
        // Si es un objeto  
        if ( control instanceof FormGroup ) {  
            Object.values( control.controls ).forEach( subControl =>  
subControl.markAsTouched() );  
        } else {  
            control.markAsTouched(); // Marcar todos como tocadas
```

```
    }  
  });  
} else {  
  if ( this.pagesTotal > 0 || this.isToSketch ) {  
    this.spinnerService.show();  
    if ( this.isToEdit ) {  
      (this.pagesTotal > 0) ? this.editPublication() :  
this.editCollectionToPublication(this.publicationToEditId);  
    } else {  
      (this.pagesTotal > 0) ? this.newPublication() : this.uploadPublicationNewPages();  
    }  
  } else {  
    this.sweetAlertGenericService.showAlertError( 'No hay páginas cargadas a la  
colección.' );  
  }  
}  
  
private newPublication(): void {  
  const arrayPageObervables: Observable<any>[] = [];  
  let pagesUrl: IUpdatePagesCommand = {  
    pages: [] = []  
  };  
  this.pagesList.controls.forEach( (pageForm: FormGroup) => {  
    const page = pageForm.get('image').value;  
    arrayPageObervables.push( this.fileServ.upload( page ) );  
  });  
  forkJoin( arrayPageObervables ).subscribe( ( dataPage ) => {  
    pagesUrl = this.getUrlFromFileService(dataPage);  
    this.uploadPublicationNewPages(pagesUrl);  
  });  
}
```

```
}, (error: Error) => {
    console.error(error);
    this.spinnerService.hide();
});
}

private editPublication(): void {
    // #region Crear un arreglo de páginas con el total que existen y asignarles un
número

    const pagesOrderByNumberList: IPageAndNumber[] = [];

    for ( let i = 0; i < this.pagesTotal; i++ ) {
        pagesOrderByNumberList.push({
            number: i + 1,
            url: null
        });
    }

    // #endregion

    // #region Obtener primero todos los que no se modificaron (se asignan nomas con
las del número que corresponde)

    this.pagesList.controls.forEach( (pageForm: FormGroup) => {
        const isNewPage = pageForm.get('image').value; // Si es nulo, es una página no
actualizada...sino es nueva.

        if ( !isNewPage ) {
            const oldPage = pageForm.value as IPageForm;

            pagesOrderByNumberList.map( page => {
                if ( page.number === oldPage.number ) {
                    page.url = oldPage.url;
                }
            });
        }
    }
}
```

```
});  
  
//#endregion  
  
//#region Obtener todas las páginas nuevas  
const arrayNewPageObervables: Observable<any>[] = [];  
  
const pagesUrl: IUpdatePagesCommand = {  
  pages: [] = []  
};  
  
this.pagesList.controls.forEach( (pageForm: FormGroup) => {  
  const page = pageForm.get('image').value;  
  // No es nulo el campo 'image' => es una página cargada nueva  
  if ( page ) {  
    arrayNewPageObervables.push( this.fileServ.upload( page ) );  
  }  
});  
  
// Si no se ha cargado una nueva imagen, ir directo a la carga de las páginas sino  
esperar a que se asignen la url necesaria  
if ( arrayNewPageObervables.length === 0 ) {  
  // << Sin paginas nuevas >>  
  pagesOrderByNumberList.forEach( (page: IPageAndNumber) => {  
    pagesUrl.pages.push({  
      images: [page.url]  
    });  
  });  
};  
  
this.uploadPublicationEditPages(pagesUrl);  
} else {  
  // << Con páginas nuevas >>  
  let auxPagesNewUrl: IUpdatePagesCommand = {  
    pages: [] = []  
  };  
};
```



```

// #region Obtener los url de las nuevas páginas
forkJoin( arrayNewPageObservables ).subscribe( ( dataPage ) => {
    auxPagesNewUrl = this.getUrlFromFileService(dataPage);

    // #region Asignar los url obtenidos, en orden, en los espacios sin url creados en
    el arreglo anteriormente

    auxPagesNewUrl.pages.forEach( (pageNew: any) => {
        pagesOrderByNumberList.find( page => !page.url ).url = pageNew.images[0];
    });

    // #endregion

    // --> Enviar para colocarlos en una publicación ya existente
    pagesOrderByNumberList.forEach( (page: IPageAndNumber) => {
        pagesUrl.pages.push({
            images: [page.url]
        });
    });

    this.uploadPublicationEditPages(pagesUrl);
}, (error: Error) => {
    console.error(error);
    this.spinnerService.hide();
});
}

//#endregion
}

private uploadPublicationNewPages( pagesUrlToUpload?: IUpdatePagesCommand ):
void {

    const createSketch: ICreateCommand = {
        name: this.formPublication.get('name').value,
        cover: this.formPublication.get('cover').value,
        synopsis: this.formPublication.get('synopsis').value,
    }
}

```

```
category_id: this.formPublication.get('category_id').value,
tags: this.formPublication.get('tags').value,
};
this.publicationService.create( createSketch ).subscribe(
  (resSketch: any) => {
    const idSketch = resSketch.id;
    // Si solo se quería generar el nuevo id, se saltea el paso de asignar las páginas
    if ( !pagesUrlToUpload ) {
      this.assignCollectionToPublication(idSketch);
    } else {
      // Subir las paginas
      this.publicationService.updatePages( idSketch, pagesUrlToUpload ).subscribe(
        (resPagesUpload: any) => {
          this.assignCollectionToPublication(idSketch);
        },
        (error: Error) => {
          console.error(error);
          this.spinnerService.hide();
        }
      );
    }
  },
  (error: Error) => {
    console.error(error);
    this.spinnerService.hide();
  }
);
}
```

```
private uploadPublicationEditPages( pagesUrlToUpload: IUpdatePagesCommand ):
void {

    // Subir las paginas

    this.publicationService.updatePages( this.publicationToEditId,
pagesUrlToUpload ).subscribe(

        (resPagesUpload: any) => {

            this.editCollectionToPublication(this.publicationToEditId);

        },

        (error: Error) => {

            console.error(error);

            this.spinnerService.hide();

        }

    );

}

private assignCollectionToPublication( idPublication: string ): void {

    if ( this.collectionArrayCheck.controls.length === 0 ) {

        ( this.isToEdit || this.isToSketch ) ?

            this.editPublicationFinish(idPublication) :
this.newPublicationFinish(idPublication) ;

    } else {

        // Primero se crea una lista con todas los observables a usar para añadir la
colección a la publicación creada

        const collectionSubscriptionsList: any[] = [];

        this.collectionArrayCheck.controls.forEach( (collection: FormControl) => {

collectionSubscriptionsList.push( this.collectionService.addPublication( collection.value ,
idPublication ) );

        } );

        // Se realiza la subscripción de todas las colección al Id de publicación y si está
correcto, se publica finalmente
```

```

forkJoin( collectionSubscriptionsList ).subscribe(
  (data: any) => {
    ( this.isToEdit || this.isToSketch ) ?
      this.editPublicationFinish(idPublication) :
this.newPublicationFinish(idPublication) ;
  },
  (error: Error) => {
    console.error(error);
    this.spinnerService.hide();
  }
);
}
}

private editCollectionToPublication( idPubcalition: string ): void {
  /*
   Se consiguen todas las colecciones que tiene la publicación a editar, si no tiene se
   va a la función que asigna las mismas. Caso que tenga alguna se quitan las mismas
   de la publicación
   antes de ir a asignarselas las que se seleccionaron al editar.
  */
  let collectionToRemove: ICollection[] = [];
  this.publicationService.getCollections( idPubcalition ).subscribe(
    (resCollection) => {
      collectionToRemove = resCollection.items;
      if ( collectionToRemove.length === 0 ) {
        this.assignCollectionToPublication(idPubcalition);
      } else {
        const collectionRemoveSubscriptionList: any[] = [];
        collectionToRemove.forEach( (collection: ICollection) => {

```

```

collectionRemoveSubscriptionList.push( this.collectionService.removePublication( collec
tion.id, idPubcalition ) );

    });

    forkJoin( collectionRemoveSubscriptionList ).subscribe(

        (res: any) => {

            this.assignCollectionToPublication(idPubcalition);

        },

        (err: Error) => {

            console.error(err);

            this.spinnerService.hide();

        }

    );

}

}

);

}

private newPublicationFinish( idPublication: string ): void {

    // Realizar la publicación en sí con todos los datos necesarios.

    this.publicationService.publish( idPublication ).subscribe(

        (resPublish: any) => {

            // Decidir es por edición (en ese caso: nueva publicaición o borrador) o
publicación totalmente nueva

            if ( this.isToEdit ) {

                this.swalFormEditPublishValid.fire();

            } else {

                this.swalFormDataValid.fire();

            }

            this.spinnerService.hide();

        },

```

```
(error: any) => {  
    console.error(error);  
    this.spinnerService.hide();  
}  
);  
}  
  
private editPublicationFinish( idPublication: string ): void {  
    const publicationUpdateCMD: IUpdateCommand = {  
        name: this.formPublication.get('name').value,  
        synopsis: this.formPublication.get('synopsis').value,  
        category_id: this.formPublication.get('category_id').value,  
        tags: this.formPublication.get('tags').value,  
        cover: this.formPublication.get('cover').value,  
    };  
    // Editar la publicación en sí con todos los datos necesarios.  
    this.publicationService.update( idPublication, publicationUpdateCMD ).subscribe(  
        (resEditPublish: any) => {  
            if ( this.isToSketch ) {  
                this.swalFormSketchValid.fire();  
                this.isToSketch = false; // Por si luego desea publicar (puede volver a guardarlo  
como borrador)  
                this.spinnerService.hide();  
            } else {  
                this.newPublicationFinish(idPublication);  
            }  
        },  
        (error: any) => {  
            console.error(error);  
            this.spinnerService.hide();  
        }  
    );  
}
```

```
    }  
    );  
  }  
  
  // #endregion  
  
  // #region Dropdown Checkbox Collection  
  public onCheckboxChangeCollection( event: MatCheckbox ): void {  
    // Comprobar si debe agregarlo a lista o no  
    if ( event.checked ) {  
      this.collectionArrayCheck.push( new FormControl( event.value ) );  
    } else {  
      // Busca en todo el array el elemento que tiene el mismo valor que el que se saco  
      para quitarlo del array  
  
      let i = 0;  
      this.collectionArrayCheck.controls.forEach( (item: FormControl) => {  
        if ( item.value === event.value ) {  
          this.collectionArrayCheck.removeAt(i);  
          return;  
        }  
        i++;  
      });  
    }  
  }  
  
  public onRadioChangeCollection(): void {  
    this.collectionArrayCheck.clear();  
  }  
  
  public isNotCheckedAllCollection(): boolean {  
    return (this.collectionArrayCheck.length === 0);  
  }  
  
  public isCheckedCollectionItem( item: IDropdownItem ): boolean {
```

```
return( (this.collectionArrayCheck.value as Array<string> ).indexOf(item.valueId) >
-1 );
}
// #endregion
// #region Tags
public addTag( event: MatChipInputEvent): void {
    const input = event.input;
    const value = event.value.trim();
    if ((value || '')) {
        this.tagsList.push(value);
    }
    // Reset the input value
    if (input) {
        input.value = '';
    }
}

public removeTag( tag: string ): void {
    const index = this.tagsList.indexOf(tag);
    if (index >= 0) {
        this.tagsList.splice(index, 1);
    }
}
// #endregion
// #region Pages
public addPage(): void {
    const newPage = this.newPage();
    // Crear elemento input de tipo 'file' para poder manejarlo desde el botón que lo
    llama
    const inputFileElement = document.createElement('input');
```



```
inputFileElement.type = 'file'; // Nota: Solo uno a la vez, para varios:
inputFileElement.multiple = multiple

inputFileElement.accept = '.png, .jpg, .jpeg';
inputFileElement.click();

// Definir la función del llamado al hacer click (cuando realiza un cambio)
inputFileElement.onChange = ( event: any ) => {
  const fdImage: FormData = new FormData();
  const pagelImage = event.target.files[0];

  // #region Cargar para previsualizar en pantalla
  const reader = new FileReader();

  reader.onload = (eventReader: any ) => {
    newPage.get('thumbailImage').setValue(eventReader.target.result);
  };

  reader.readAsDataURL(pagelImage);

  //#endregion

  fdImage.append('image', pagelImage, pagelImage.name);
  newPage.get('image').setValue(fdImage);
  this.pagesList.push( newPage );
};
}

public removePage( index: number ): void {
  this.pagesList.removeAt( index );
  const listLength = this.pagesTotal;
  for ( let i = index; i < listLength; i++ ) {
    this.pagesList.value[i].number = i + 1;
  }
}

private newPage( pageObject?: IPAGE ): FormGroup {
  if ( pageObject ) {
```

```
const urlPage = pageObject.images[0].url;
return this.fb.group({
  number: (pageObject.number) + 1,
  image: null,
  thumbaillImage: urlPage,
  url: urlPage
});
}
else {
  return this.fb.group({
    number: this.pagesList.length + 1,
    image: [new FileReader(), Validators.required ],
    thumbaillImage: "",
    url: ""
  });
}
}

public getUrlFromFileService( dataFilePageUploaded: any[] ):
IUpdatePagesCommand {
  const auxStringUrlPage: IUpdatePagesCommand = {
    pages: [] = []
  };
  dataFilePageUploaded.forEach( pageUploaded => {
    auxStringUrlPage.pages.push( {
      images: [pageUploaded.files[0].url]
    });
  });
  return auxStringUrlPage;
}
```

```
//#endregion  
  
// #region Getters  
  
get nombreNovalido(): boolean {  
    return ( this.formPublication.get('name').invalid &&  
this.formPublication.get('name').touched );  
}  
  
get synopsisNovalido(): boolean {  
    return ( this.formPublication.get('synopsis').invalid &&  
this.formPublication.get('synopsis').touched );  
}  
  
get synopsisLenght(): number {  
    return this.formPublication.get('synopsis').value.length;  
}  
  
get pagesList(): FormArray {  
    return this.formPublication.get('pagesList') as FormArray;  
}  
  
get pagesTotal(): number {  
    return this.pagesList.length;  
}  
  
get collectionArrayCheck(): FormArray {  
    return this.formPublication.get('collectionArray') as FormArray;  
}  
  
get categoryNoValido(): boolean {  
    return ( this.formPublication.get('category_id').invalid &&  
this.formPublication.get('category_id').touched );  
}  
  
get categoryFormValueId(): string {  
    return this.formPublication.get('category_id').value;  
}
```

```

get categoryValueName(): string {
    return this.categoryList.filter( element => {
        return (element.valued === this.categoryFormValued);
    } )[0].name;
}

//#endregion
}

```

**Código Fuente – Back end**

**Create** (capa de aplicación)

```

use serde::{Deserialize, Serialize};
use common::event::EventPublisher;
use common::result::Result;
use crate::domain::author::{AuthorId, AuthorRepository};
use crate::domain::category::{CategoryId, CategoryRepository};
use crate::domain::publication::{
    Header, Image, Name, Page, Publication, PublicationRepository, Synopsis, Tag,
};
#[derive(Deserialize)]
pub struct PageDto {
    pub images: Vec<String>,
}
#[derive(Deserialize)]
pub struct CreateCommand {
    pub name: String,
    pub synopsis: String,
    pub category_id: String,
    pub tags: Vec<String>,
    pub cover: String,
}

```

```
pub pages: Option<Vec<PageDto>>,
}
#[derive(Serialize)]
pub struct CreateResponse {
    pub id: String,
}
pub struct Create<'a> {
    event_pub: &'a dyn EventPublisher,
    author_repo: &'a dyn AuthorRepository,
    category_repo: &'a dyn CategoryRepository,
    publication_repo: &'a dyn PublicationRepository,
}
impl<'a> Create<'a> {
    pub fn new(
        event_pub: &'a dyn EventPublisher,
        author_repo: &'a dyn AuthorRepository,
        category_repo: &'a dyn CategoryRepository,
        publication_repo: &'a dyn PublicationRepository,
    ) -> Self {
        Create {
            event_pub,
            author_repo,
            category_repo,
            publication_repo,
        }
    }
    pub async fn exec(&self, auth_id: String, cmd: CreateCommand) ->
    Result<CreateResponse> {
        let name = Name::new(cmd.name)?;
```

```
let synopsis = Synopsis::new(cmd.synopsis?);
let mut tags = Vec::new();
for tag in cmd.tags.into_iter() {
    tags.push(Tag::new(tag)?);
}
let cover = Image::new(cmd.cover?);
let category_id = CategoryId::new(cmd.category_id?);
self.category_repo.find_by_id(&category_id).await?;
let header = Header::new(name, synopsis, category_id, tags, cover?);
let author_id = AuthorId::new(auth_id?);
self.author_repo.find_by_id(&author_id).await?;
let mut publication = Publication::new(
    self.publication_repo.next_id().await?,
    author_id.clone(),
    header,
)?;
// Add pages
if let Some(page_dtos) = cmd.pages {
    let mut pages = Vec::new();
    for (page_n, page) in page_dtos.into_iter().enumerate() {
        let mut images = Vec::new();
        for image in page.images.into_iter() {
            images.push(Image::new(image)?);
        }
        let mut page = Page::new(page_n as u32?);
        page.set_images(images)?;
        pages.push(page);
    }
}
```

```

        publication.set_pages(pages)?;
    }

    self.publication_repo.save(&mut publication).await?;

    self.event_pub

        .publish_all(publication.events().to_vec()?)

        .await?;

    Ok(CreateResponse {
        id: publication.base().id().to_string(),
    })
}
}
}

```

**Publish** (capa de aplicación)

```

use common::error::Error;
use common::event::EventPublisher;
use common::request::CommandResponse;
use common::result::Result;
use crate::domain::author::{AuthorId, AuthorRepository};
use crate::domain::publication::{PublicationId, PublicationRepository};

pub struct Publish<'a> {
    event_pub: &'a dyn EventPublisher,
    author_repo: &'a dyn AuthorRepository,
    publication_repo: &'a dyn PublicationRepository,
}

impl<'a> Publish<'a> {
    pub fn new(
        event_pub: &'a dyn EventPublisher,
        author_repo: &'a dyn AuthorRepository,
        publication_repo: &'a dyn PublicationRepository,
    )

```

```

) -> Self {
  Publish {
    event_pub,
    author_repo,
    publication_repo,
  }
}

pub async fn exec(&self, auth_id: String, publication_id: String) ->
Result<CommandResponse> {
  let author_id = AuthorId::new(auth_id)?;
  self.author_repo.find_by_id(&author_id).await?;
  let publication_id = PublicationId::new(publication_id)?;
  let mut publication = self.publication_repo.find_by_id(&publication_id).await?;
  if publication.author_id() != &author_id {
    return Err(Error::not_owner("publication"));
  }
  publication.publish()?;
  self.publication_repo.save(&mut publication).await?;
  self.event_pub
    .publish_all(publication.events().to_vec()?)
    .await?;
  Ok(CommandResponse::default())
}
}

```

**Publication** (capa de dominio)

```

#[derive(Debug, Clone)]
pub struct Publication {
  base: AggregateRoot<PublicationId>,

```



```
events: Events<PublicationEvent>,
author_id: AuthorId,
header: Header,
pages: Vec<Page>,
contract: bool,
statistics: Statistics,
status_history: StatusHistory<Status>,
}
impl Publication {
pub fn new(id: PublicationId, author_id: AuthorId, header: Header) -> Result<Self>
{
let mut publication = Publication {
base: AggregateRoot::new(id),
events: Events::new(),
author_id,
header,
pages: Vec::new(),
contract: false,
statistics: Statistics::default(),
status_history: StatusHistory::new(Status::Draft),
};
publication.events.record_event(PublicationEvent::Created {
id: publication.base().id().to_string(),
author_id: publication.author_id().to_string(),
name: publication.header().name().to_string(),
synopsis: publication.header().synopsis().to_string(),
category_id: publication.header().category_id().to_string(),
tags: publication
.header()

```

```
        .tags()
        .iter()
        .map(|t| t.name().to_string())
        .collect(),
    cover: publication.header().cover().to_string(),
});
Ok(publication)
}
pub fn build(
    base: AggregateRoot<PublicationId>,
    author_id: AuthorId,
    header: Header,
    pages: Vec<Page>,
    contract: bool,
    statistics: Statistics,
    status_history: StatusHistory<Status>,
) -> Self {
    Publication {
        base,
        events: Events::new(),
        author_id,
        header,
        pages,
        contract,
        statistics,
        status_history,
    }
}
```

```
pub fn set_header(&mut self, header: Header) -> Result<()> {
    self.header = header;
    self.make_draft()?;
    self.base.update();
    self.events.record_event(PublicationEvent::HeaderUpdated {
        id: self.base().id().to_string(),
        name: self.header().name().to_string(),
        synopsis: self.header().synopsis().to_string(),
        category_id: self.header().category_id().to_string(),
        tags: self
            .header()
            .tags()
            .iter()
            .map(|t| t.name().to_string())
            .collect(),
        cover: self.header().cover().url().to_string(),
    });
    Ok(())
}

pub fn set_pages(&mut self, pages: Vec<Page>) -> Result<()> {
    self.pages = pages;
    self.make_draft()?;
    self.base.update();
    self.events.record_event(PublicationEvent::PagesUpdated {
        id: self.base().id().to_string(),
        pages_count: self.pages().len(),
    });
    Ok(())
}
```

```
}  
  
pub fn set_statistics(&mut self, statistics: Statistics) -> Result<> {  
    self.statistics = statistics;  
  
    self.events  
        .record_event(PublicationEvent::StatisticsUpdated {  
            id: self.base().id().to_string(),  
            views: self.statistics().views(),  
            unique_views: self.statistics().unique_views(),  
            readings: self.statistics().readings(),  
            likes: self.statistics().likes(),  
            reviews: self.statistics().reviews(),  
            stars: self.statistics().stars(),  
        });  
    Ok(())  
}  
  
pub fn view(&mut self, reader: &Reader, unique: bool) -> Result<View> {  
    if !self.is_published() {  
        return Err(Error::new("publication", "not_published"));  
    }  
  
    self.statistics.add_view(unique);  
  
    self.events.record_event(PublicationEvent::Viewed {  
        reader_id: reader.base().id().to_string(),  
        publication_id: self.base().id().to_string(),  
        unique,  
    });  
  
    self.events  
        .record_event(PublicationEvent::StatisticsUpdated {  
            id: self.base().id().to_string(),
```

```
views: self.statistics().views(),
unique_views: self.statistics().unique_views(),
readings: self.statistics().readings(),
likes: self.statistics().likes(),
reviews: self.statistics().reviews(),
stars: self.statistics().stars(),
});
Ok(View::new(
    ReaderPublicationId::new(reader.base().id().clone(), self.base().id().clone()),
    unique,
    )?)
}
pub fn read(&mut self, reader: &Reader) -> Result<Reading> {
    if !self.is_published() {
        return Err(Error::new("publication", "not_published"));
    }
    if self.has_contract() && !reader.is_subscribed() {
        return Err(Error::new("reader", "not_subscribed"));
    }
    self.statistics.add_reading();
    self.events.record_event(PublicationEvent::Read {
        reader_id: reader.base().id().to_string(),
        publication_id: self.base().id().to_string(),
    });
    self.events
        .record_event(PublicationEvent::StatisticsUpdated {
            id: self.base().id().to_string(),
            views: self.statistics().views(),
```

```
        unique_views: self.statistics().unique_views(),
        readings: self.statistics().readings(),
        likes: self.statistics().likes(),
        reviews: self.statistics().reviews(),
        stars: self.statistics().stars(),
    });

    Ok(Reading::new(ReaderPublicationId::new(
        reader.base().id().clone(),
        self.base().id().clone(),
    )?))
}

pub fn like(&mut self, reader: &Reader) -> Result<Like> {
    if !self.is_published() {
        return Err(Error::new("publication", "not_published"));
    }

    if self.has_contract() && !reader.is_subscribed() {
        return Err(Error::new("reader", "not_subscribed"));
    }

    self.statistics.add_like();

    self.events.record_event(PublicationEvent::Liked {
        reader_id: reader.base().id().to_string(),
        publication_id: self.base().id().to_string(),
    });

    self.events
        .record_event(PublicationEvent::StatisticsUpdated {
            id: self.base().id().to_string(),
            views: self.statistics().views(),
            unique_views: self.statistics().unique_views(),
```

```
        readings: self.statistics().readings(),
        likes: self.statistics().likes(),
        reviews: self.statistics().reviews(),
        stars: self.statistics().stars(),
    });
    Ok(Like::new(ReaderPublicationId::new(
        reader.base().id().clone(),
        self.base().id().clone(),
    )?))
}
pub fn unlike(&mut self, reader: &Reader) -> Result<> {
    if !self.is_published() {
        return Err(Error::new("publication", "not_published"));
    }
    if self.has_contract() && !reader.is_subscribed() {
        return Err(Error::new("reader", "not_subscribed"));
    }
    self.statistics.remove_like();
    self.events.record_event(PublicationEvent::Unliked {
        reader_id: reader.base().id().to_string(),
        publication_id: self.base().id().to_string(),
    });
    self.events
        .record_event(PublicationEvent::StatisticsUpdated {
            id: self.base().id().to_string(),
            views: self.statistics().views(),
            unique_views: self.statistics().unique_views(),
            readings: self.statistics().readings(),
```

```
        likes: self.statistics().likes(),
        reviews: self.statistics().reviews(),
        stars: self.statistics().stars(),
    });
    Ok(())
}

pub fn review(&mut self, reader: &Reader, stars: Stars, comment: Comment) ->
Result<Review> {
    if !self.is_published() {
        return Err(Error::new("publication", "not_published"));
    }
    if self.has_contract() && !reader.is_subscribed() {
        return Err(Error::new("reader", "not_subscribed"));
    }
    self.statistics.add_review(&stars);
    self.events.record_event(PublicationEvent::Reviewed {
        reader_id: reader.base().id().to_string(),
        publication_id: self.base().id().to_string(),
        stars: stars.value(),
        comment: comment.to_string(),
    });
    self.events
        .record_event(PublicationEvent::StatisticsUpdated {
            id: self.base().id().to_string(),
            views: self.statistics().views(),
            unique_views: self.statistics().unique_views(),
            readings: self.statistics().readings(),
            likes: self.statistics().likes(),
            reviews: self.statistics().reviews(),
```



```
        stars: self.statistics().stars(),
    });
    Ok(Review::new(
        ReaderPublicationId::new(reader.base().id().clone(), self.base().id().clone())?,
        stars,
        comment,
    )?)
}

pub fn delete_review(&mut self, reader: &Reader, stars: &Stars) -> Result<> {
    if !self.is_published() {
        return Err(Error::new("publication", "not_published"));
    }
    if self.has_contract() && !reader.is_subscribed() {
        return Err(Error::new("reader", "not_subscribed"));
    }
    self.statistics.remove_review(stars);
    self.events.record_event(PublicationEvent::ReviewDeleted {
        reader_id: reader.base().id().to_string(),
        publication_id: self.base().id().to_string(),
    });
    self.events
        .record_event(PublicationEvent::StatisticsUpdated {
            id: self.base().id().to_string(),
            views: self.statistics().views(),
            unique_views: self.statistics().unique_views(),
            readings: self.statistics().readings(),
            likes: self.statistics().likes(),
            reviews: self.statistics().reviews(),
```

```
        stars: self.statistics().stars(),
    });
    Ok(())
}

pub fn add_contract(&mut self) -> Result<()> {
    if !self.is_published() {
        return Err(Error::new("publication", "not_published"));
    }
    if self.has_contract() {
        return Err(Error::new("publication", "already_has_a_contract"));
    }
    self.contract = true;
    self.base.update();
    self.events.record_event(PublicationEvent::ContractAdded {
        id: self.base().id().to_string(),
    });
    Ok(())
}

pub fn remove_contract(&mut self) -> Result<()> {
    if !self.has_contract() {
        return Err(Error::new("publication", "does_not_have_a_contract"));
    }
    self.contract = false;
    self.base.update();
    self.events.record_event(PublicationEvent::ContractAdded {
        id: self.base().id().to_string(),
    });
    Ok(())
}
```

```
}  
  
pub fn make_draft(&mut self) -> Result<> {  
    if !matches!(self.status_history().current(), Status::Draft) {  
        let draft = self.status_history.current().draft()?;  
        self.status_history.add_status(draft);  
        self.base.update();  
        self.events.record_event(PublicationEvent::ChangedToDraft {  
            id: self.base().id().to_string(),  
        });  
    }  
    Ok(())  
}  
  
pub fn publish(&mut self) -> Result<> {  
    let waiting_approval = self.status_history.current().publish()?;  
    if self.pages.is_empty() {  
        return Err(Error::new("publication", "does_not_have_pages"));  
    }  
    for page in self.pages().iter() {  
        if page.images().is_empty() {  
            return Err(Error::new("publication", "empty_page")  
                .add_context("page", &page.number().to_string()));  
        }  
    }  
    self.status_history.add_status(waiting_approval);  
    self.base.update();  
    self.events.record_event(PublicationEvent::ApprovalWaited {  
        id: self.base().id().to_string(),  
    });  
}
```

```
Ok()  
}  
  
pub fn approve(&mut self, user_id: UserId, comment: Comment) -> Result<> {  
    let published = self.status_history.current().approve(user_id, comment)?;  
    self.status_history.add_status(published);  
    self.base.update();  
    self.events.record_event(PublicationEvent::Published {  
        id: self.base().id().to_string(),  
        author_id: self.author_id().to_string(),  
        name: self.header().name().to_string(),  
        synopsis: self.header().synopsis().to_string(),  
        category_id: self.header().category_id().to_string(),  
        tags: self  
            .header()  
            .tags()  
            .iter()  
            .map(|t| t.name().to_string())  
            .collect(),  
        cover: self.header().cover().url().to_string(),  
        pages_count: self.pages().len(),  
    });  
    Ok()  
}  
  
pub fn reject(&mut self, user_id: UserId, comment: Comment) -> Result<> {  
    let rejected = self.status_history().current().reject(user_id, comment)?;  
    self.status_history.add_status(rejected);  
    self.base.update();  
    self.events.record_event(PublicationEvent::Rejected {
```

```
        id: self.base().id().to_string(),
    });
    Ok(())
}

pub fn delete(&mut self) -> Result<()> {
    self.base.delete();
    self.events.record_event(PublicationEvent::Deleted {
        id: self.base().id().to_string(),
    });
    Ok(())
}
}
```

Tabla 1. 5 Plantilla de documentación de código de "Publicar Obra".

## 1.4. Arquitectura

Si bien ya se cuenta con un diseño sólido de la plataforma, es necesario establecer una capa de abstracción sobre el mismo que beneficie el desarrollo e implementación de este.

En primer lugar, es práctico dividir el sistema en capas que satisfagan las directrices de alta cohesión y bajo acoplamiento en todo el sistema. Su principal objetivo es aportar a la mantenibilidad de este a largo plazo y las posibilidades de escalar en caso de ser necesario, debido a un creciente número de usuarios.

Para lograr lo anterior se hace uso de distintas metodologías y patrones fuertemente establecidos en el campo del desarrollo de software actual: **Domain-Driven Design** y **Clean Architecture**.

### 1.4.1. Domain-Driven Design

Es un enfoque para el desarrollo de software con necesidades complejas mediante una profunda conexión entre la implementación y los conceptos del modelo y núcleo del negocio.

El DDD no es una tecnología ni una metodología, este provee una estructura de prácticas y terminologías para tomar decisiones de diseño que enfoquen y aceleren el manejo de dominios complejos en los proyectos de software (Fundación Wikipedia, s/f).

Sus ventajas son las siguientes:

- ❖ Comunicación efectiva entre expertos del dominio y expertos técnicos a través de Ubiquitous Language.
- ❖ Foco en el desarrollo de un área dividida del dominio (subdominio) a través de Bounded Context's.
- ❖ El software es más cercano al dominio, y por lo tanto es más cercano al cliente.
- ❖ Código bien organizado, permitiendo el testing de las distintas partes del dominio de manera aisladas.
- ❖ Lógica de negocio reside en un solo lugar, y dividida por contextos.
- ❖ Mantenibilidad a largo plazo.

Como se ha mencionado, DDD aporta una capa de abstracción entre el diseño realizado y la implementación que lo satisfaga.

### 1.4.2. Clean Architecture

Se elige esta arquitectura ampliamente difundida para la implementación de la plataforma ya que aporta, además de buenas prácticas, una estructura sólida para el desarrollo y la mantenibilidad del código a futuro.

El diseño realizado en la sección anterior se corresponde con el dominio y la capa de aplicación en esta arquitectura. Lo importante que se debe notar es que las capas interiores no tienen conocimiento de las exteriores y no tienen conocimiento de las mismas, lo que se traduce en un bajo acoplamiento, pero principalmente, en una correcta cohesión y responsabilidad de los objetos involucrados.

Los casos de usuario se ejecutan en la capa de aplicación que no es más que un conjunto de objetos encargados de coordinar los objetos de dominio. Es el dominio el que está influenciado por las reglas de negocio y la lógica del diseño planteado.

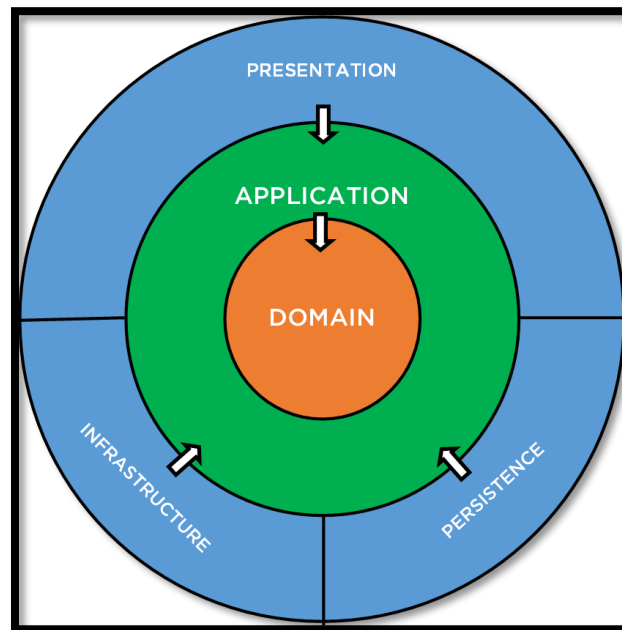


Figura 1. 108 Arquitectura limpia (Clean Architecture).

La capa de infraestructura posee las implementaciones reales de los servicios y repositorios (encargados de gestionar la persistencia de los objetos). Es en esta capa donde se implementan los repositorios encargados de interactuar con el motor de base de datos. Y el dominio únicamente establece la abstracción de cada repositorio en forma de *interface* a implementar por las clases concretas.

### 1.4.3. Interacción

Vale destacar que DDD no es más que una capa de abstracción que unifica el diseño real con una implementación consistente. Y basa la estructuración del sistema en *Bounded Contexts*.

Este concepto (Contexto Delimitado) nace con DDD y es el patrón central del mismo. La idea es que cuando se tienen modelos grandes estos se dividan en contextos delimitados poniendo de forma explícita la relación entre ellos.

La idea detrás de un Bounded Context es conseguir un *Lenguaje Ubiquo* que sea único para el modelo y que el mismo no se preste a confusión, permitiendo así una mejor comunicación entre los desarrollos de software y los expertos de dominio.

Es importante aclarar que no existe una relación directa y unívoca entre los módulos que se han definido en la sección de Diseño y los *Bounded Contexts* aquí presentados. Cada módulo puede estar representado por uno o más *bounded contexts*, ya que la misión de los mismos es facilitar la transaccionalidad de las entidades en su conjunto. Dicho de otra forma, agrupa las entidades por responsabilidad (como en los

módulos definidos) pero al mismo tiempo por transaccionalidad de ejecución. Un módulo, generalmente, estará representado por más de un *bounded context*.

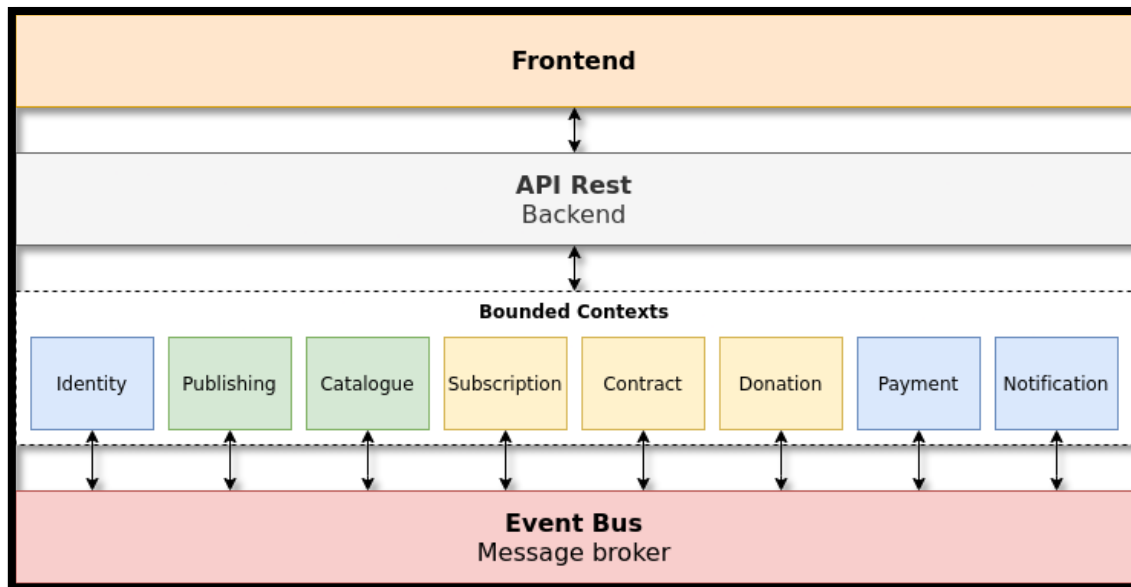


Figura 1. 109 Esquema DDD.

Como se observa en la arquitectura presente en el diagrama anterior, existen dos bounded contexts centrales que definen la razón de ser de la plataforma: **Publishing** y **Catalogue**. Ambos resaltados en color verde.

**Subscription**, **Contract** y **Donation** son BC (bounded context) críticos, que requieren atención especial debido a su naturaleza.

Los BCs (bounded context) resaltados en azul son de soporte para los demás. **Identity**, encargado de la gestión del usuario, la autenticación y autorización, **Payment**, encargado de la gestión de pagos mediante servicios externos y **Notificación**, encargado de las notificaciones.

Esta división es meramente física, por necesidades de transaccionalidad y comunicación asíncrona. La relación entre estos y los módulos es que la ejecución de un caso de uso, dentro de un módulo, puede suponer la ejecución de más de un BC (bounded context) debido a la emisión de mensajes mediante la publicación de eventos, que se trata a continuación.

En cambio, los módulos del sistema refieren a una división lógica que dicta cómo deben contemplarse los casos de uso de acuerdo con la utilidad que aportan.



#### 1.4.4. Consistencia Eventual

En todo negocio existe gran comunicación y asíncrona entre las partes involucradas, y la plataforma *Omics* no es la excepción.

Determinadas tareas no requieren que exista una comunicación directa entre las partes ya que estas se pueden diferir, e incluso subdividirse.

La consistencia eventual plantea el intercambio de mensajes entre las partes interesadas en realizar alguna acción cuando un evento determinado ocurra.

En primer lugar, beneficia al rendimiento de la plataforma ya que no es necesario hacer esperar al usuario para recibir una respuesta del sistema al tener que ejecutar múltiples módulos al mismo tiempo. Las distintas partes del sistema reaccionan a eventos emitidos por otras partes en cuanto reciben el evento, y hay tiempo de CPU disponible para ejecutar el manejador de eventos.

Se observa claramente que esto aporta aún más a un bajo acoplamiento del sistema ya que el emisor de un evento no tiene que preocuparse por quiénes lo reciben o son los interesados, como sí sucedería en la comunicación síncrona y directa (por ejemplo, al llamar un caso de uso desde otro para notificarle de alguna acción).

Por ejemplo, cuando un usuario se registra, el caso de uso "Registrar Cuenta de Usuario" emite un evento, el cual es captado por el BC (bounded context) de notificaciones, que también corresponde al módulo que tiene el mismo nombre, y envía un e-mail al usuario. Esta acción puede realizarse de forma diferida, y la demora en la ejecución es aceptable.

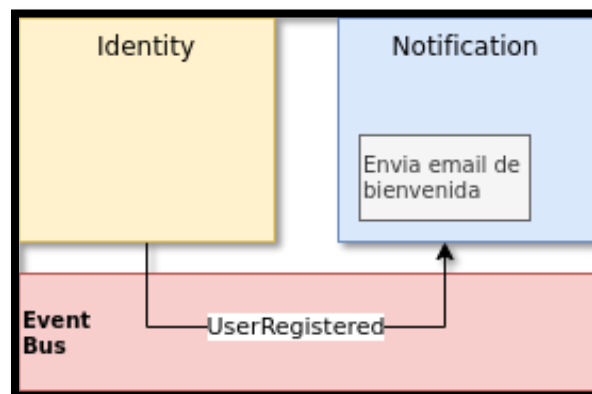


Figura 1. 110 Ejemplo de intercambio de mensaje de Notificaciones.

Otro ejemplo concreto es lo que sucede con la creación de una suscripción por parte de un usuario lector. Debido a la naturaleza de los sistemas de pago externos, los cuales reciben los datos de pago del usuario, gestionan el pago y luego notifican al

sistema mediante una llamada una url especificada por este (*callback\_url*, de forma asíncrona, con funcionamiento similar a los eventos nombrados aquí).

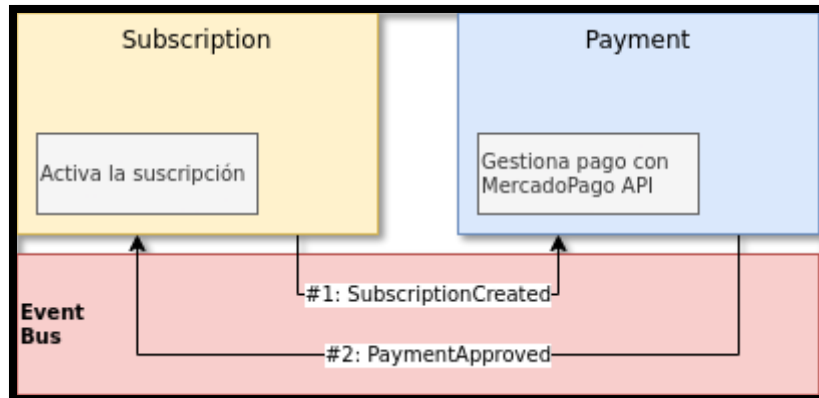


Figura 1. 111 Ejemplo de intercambio de mensaje de suscripción.

## 2. Planificación de Capacitación

### 2.1. Introducción

Para poder llevar a cabo la planificación se organiza una guía para las personas que se relacionan con el sistema, a fin de alcanzar los objetivos que permitan utilizar la plataforma, y sus funcionalidades con facilidad y claridad.

### 2.2. Alcance

Se ha considerado que el tiempo estimado de la capacitación es de 7 días (1 semana) y se contempla a los usuarios finales que interactúan con la plataforma. El objetivo es aprender el funcionamiento del sistema, y que puedan interactuar con todas las opciones que ofrece *Omics*.

### 2.3. Destinatarios

Los destinatarios de las capacitaciones son las personas que hacen uso de la plataforma, los cuales son:

- ❖ Administradores del Sistema.
- ❖ Gestor de Contenido.
- ❖ Usuario (Autor y Lector).

### 2.4. Objetivos de la Planificación de Capacitación

El objetivo principal de la planificación es poder organizar las capacitaciones de los trabajadores de los destinatarios del sistema para lograr comprender el uso y manejo de la plataforma, y que de esta forma puedan incrementar sus conocimientos. También, se pueden impartir sistemáticamente nuevos recursos para que puedan adquirirlas de forma rápida. Esto provoca un incentivo y mejora en el uso de los módulos dentro de la plataforma.

Otros objetivos que se tienen en cuenta son los siguientes:

- ❖ Explicar el manejo de roles y permisos al que se tiene acceso, como los parámetros del sistema.
- ❖ Explicar la forma de crear publicaciones de obras.
- ❖ Detallar el proceso de gestión de publicaciones.
- ❖ Detallar la solicitud de contratos y qué se tiene en cuenta al aceptarlos.
- ❖ Enumerar las diferentes opciones de búsqueda que se pueden realizar y los filtros a utilizar.
- ❖ Explicar cómo se pueden calificar las publicaciones.

- ❖ Explicar la forma de pago disponible y sus respectivos métodos de pago.
- ❖ Explicar los beneficios de la suscripción/donación y cómo hacerlo.
- ❖ Detallar los diferentes reportes que existen con la información obtenida de las estadísticas.

## 2.5. Temas de Capacitación

Para seleccionar los temas de la capacitación, se tiene en cuenta que las inversiones que se hacen son para el crecimiento y el desarrollo de los destinatarios dentro de la plataforma, aunque los beneficios se vean a mediano o largo plazo. Se identifican las habilidades o cualidades que se quieren fortalecer y que resulten clave para el logro de resultados.

El equipo de *Omics* considera importante promover la cultura de aprendizaje, ya que la idea es que las personas que hagan uso de la plataforma, ya sea de la parte interna o externa del sistema, comprendan las funcionalidades y actividades que pueden realizar con la misma.

Antes de impartir capacitación a grupos numerosos, se corre una prueba piloto con un grupo más reducido de personas y permita que los participantes expresen sus puntos de vista. De esta manera se pueden realizar los ajustes necesarios antes de ofrecer la capacitación de forma masiva. Cuando esto comience, se recalca el objetivo de la capacitación para que las personas que asistan sientan que lo enseñado les será útil para el desenvolvimiento con la plataforma, y no que sientan que es una pérdida de tiempo.

Los temas a tratar en la capacitación se han considerado por destinatarios para poder cubrir las necesidades específicas de cada uno:

- ❖ Administradores del Sistema:
  - Gestión de Usuarios.
  - Gestión de Publicaciones.
  - Gestión de Contratos.
  - Gestión de Reportes estadísticos.
  - Administración de configuraciones del sistema.
- ❖ Content Manager (Gestor de Contenido):
  - Gestión de Contratos.
  - Gestión de Publicaciones.
- ❖ Usuario (Autor-Lector):
  - Gestión de Configuración de Cuenta.
  - Gestión de Catálogo y Contenido.
  - Manejo de Visualización y Seguimiento del Contenido.
  - Manejo de Reseñas de Obras.

- Gestión de Suscripción y Donación.
- Gestión de Publicación de Obras.
- Gestión de Contratos.

## 2.6. Método o modalidad de Capacitación

Los métodos o modalidad de capacitación se refieren a las diversas formas que existen para organizar, implementar y ejecutar los procesos de enseñanza de manera de alcanzar los objetivos de aprendizaje previstos.

La elección depende, fundamentalmente, de los recursos disponibles, de los objetivos de aprendizaje que se persiguen en la acción de capacitación de que se trate, y especialmente, del perfil de las personas a las que estará destinada, en todo lo relacionado con su capacidad y voluntad para aprender.

Los métodos o modalidades que se contemplan son las siguientes:

- ❖ **Capacitación online o e-learning:** Son capacitaciones virtuales. Se caracterizan por emplear íntegramente recursos de las tecnologías de la información, en especial internet. Mediante una PC, una laptop, una Tablet, e incluso un Smartphone es posible conectarse en línea con los instructores y recibir las clases, intervenir en ellas, entregar trabajos, etcétera. Está destinado a los usuarios (Autor y Lector), los temas a tratar son los nombrados anteriormente en la **SECCIÓN 2.5 TEMAS DE CAPACITACIÓN**. Se cuenta con material a disposición en la plataforma a través de videos tutoriales y preguntas frecuentes (FAQs).
- ❖ **Capacitación presencial:** Son capacitaciones o formaciones tradicionales, donde los participantes y el capacitador acuden y se encuentran en un lugar físico acondicionado para desarrollar las clases. Estas se realizan y cuentan con intervenciones de los participantes, produciéndose debates, exposiciones, comentarios verbales que enriquecen el tema tratado in situ. Este esquema tradicional puede ser flexible desde el punto de vista de la adaptación a la realidad de cada participante, con contenidos, ejercicios prácticos y tratamientos de experiencias sobre la base de situaciones reales. En la capacitación presencial se puede dar una retroalimentación más dinámica, pues los que asisten pueden tratar en simultáneo un tema específico y recibir rápidamente los aportes de los otros participantes incluyendo a la persona que dicta la clase. Se encuentra destinado al personal interno de Omics, como los Administradores del Sistema y Gestores de Contenido. La capacitación consiste en clases dictadas en un espacio de coworking donde se puede interactuar con las personas que participan de la misma. En cada clase, se presenta y explica un tema particular que está contenido en el manual de usuario.

- ❖ **Capacitación grupal:** Modalidad de capacitación en la cual el sujeto de aprendizaje es un grupo de participantes, y la enseñanza se imparte en forma colectiva. Se utiliza en las clases presenciales cuando se aborda una temática en común con los destinatarios presentes. Facilita la dinámica y comprensión de los participantes, ya que se pueden ayudar entre ellos en caso de que se genere alguna duda.
- ❖ **Capacitación individual:** Es la capacitación que se adapta a las características individuales de cada participante, considerando sus potencialidades y limitaciones en cuanto a conocimientos previos, experiencia, intereses y estilo de aprendizaje. Por un lado, está destinada al usuario que tiene el material para interactuar con la plataforma, en el que puede ir aumentando sus conocimientos a medida que navegue y utilice la misma. Otros destinatarios de este tipo de capacitación son los administradores del sistema y gestores de contenido, que pueden reforzar lo aprendido con el manual de usuario.

## 2.7. Recursos Necesarios

El esfuerzo y la inversión no tienen el efecto deseado si quien imparte la capacitación no cuenta con el dominio del tema y del manejo de grupos. Para esto, se contacta al que facilita el curso y asegurarse que los materiales de apoyo sean claros y atractivos. También cerciorarse de que cuenta con todos los requerimientos técnicos necesarios.

Los recursos son importantes para determinar los elementos con los que se cuenta para llevar a cabo la estrategia de capacitación, y así cumplir con las expectativas del programa.

Se han considerado los siguientes recursos:

- ❖ Recursos necesarios para la realización de Capacitación online o e-learning:
  - **Videos tutoriales:** Es una clase o lección virtual, que se puede repetir en el tiempo para su aprendizaje. En ella se detalla parte integral del funcionamiento del sistema para los usuarios finales.
  - **Notebook, Smartphone, Tablet, o PC:** Se utiliza como medio funcional para la creación, y reproducción de los videos.
  - **Software de grabación de video y audio:** Se puede utilizar distintos tipos de software dependiendo del dispositivo que se utiliza. Por ejemplo: Camtasia u OBS como software privativo en notebook y computadora de escritorio, InShot para grabación desde celulares o Tablet, entre otros.
  - **Manual de usuario:** En el que se hace hincapié en las FAQs (preguntas frecuentes de los usuarios).

- **Creador del video:** Persona que se encarga del manejo de la herramienta para la creación, grabación, y edición de los videos tutoriales.
- **Narrador del video, especialista o instructor:** Se encarga de la presentación visual y explicación del contenido a tratar. Es una persona con fluidez para el habla y que posee conocimientos del sistema.
- ❖ Recursos necesarios para la realización de la Capacitación presencial:
  - **Infraestructura:** Ambiente adecuado donde se realizan las capacitaciones presenciales. Por ejemplo: Espacio de coworking.
  - **Proyectores:** Destinado para mostrar presentaciones o materiales didácticos de la clase desde una notebook o computadora de escritorio.
  - **Notebook, Smartphone, Tablet o PC:** Se utiliza como medio funcional para realizar la presentación en la clase.
  - **Manual de usuario:** Se detallan las funcionalidades de todo el sistema que sirven para consultar las dudas que se van presentando.
  - **Creador del manual de usuario:** Persona que tiene acceso al sistema, y conoce a la perfección sus funcionalidades para poder crear el manual de usuario.
  - **Editor de imágenes:** Software especializado en la edición de fotos o capturas de imágenes que sirve para resaltar las funcionalidades de las pantallas del sistema.

## 2.8. Tiempos de Capacitación

Se establecen los tiempos para luego poder medir los resultados, y parámetros para controlar los avances.

Según la planificación que se ha propuesto seguir, el tiempo destinado a la capacitación es el siguiente:

- ❖ Realizar manual de usuario (15 días).
- ❖ Realizar tutoriales (10 días).
- ❖ Diseñar plan de capacitación (11 días).
- ❖ Entrega de material en la plataforma para usuarios finales (2 días).
- ❖ Capacitar usuarios internos de la organización (3 días).
  - Capacitación de Administradores del Sistema (1 día).
  - Capacitación de Gestor de Contenido (1 día).
  - Capacitación de Usuario (Autor y Lector) (1 día).
- ❖ Evaluación de las capacitaciones (5 días).

Se tiene en cuenta que el tiempo total para realizar todas las actividades relacionadas a las capacitaciones es de 36 días. Se logra acortar los tiempos ya que varias

se hacen en paralelo. La primera tarea comienza el miércoles 30/9/20 y la finalización de todo es el lunes 26/10/20.

Para una mayor comprensión de las fechas establecidas de cada tarea mencionadas anteriormente [VER ANEXO N.º 1 DIAGRAMA DE GANTT](#).

## 2.9. Evaluación de Capacitación

El tiempo que se destina a la evaluación de las capacitaciones es de 5 días corridos. La fecha de inicio es jueves 22/10/20 y finaliza el lunes 26/10/20.

Para poder considerar los resultados de las capacitaciones, a partir de las actividades realizadas se tienen en cuenta varias acciones, entre esas, buscar diferentes opiniones de los participantes, el alcance del aprendizaje, el cambio en el comportamiento y el logro de los objetivos de la capacitación y el desarrollo.

Para medir el rendimiento es importante que los participantes de la actividad de capacitación reaccionen de manera favorable hacia ella, refiriéndonos al interés, intención y motivación. Esto sirve para medir cuan satisfechos están con aspectos tales como la metodología, los instructores, los contenidos y la infraestructura de formación, entre otros.

Las opiniones sirven para realizar mejoras y tomar decisiones para rediseñar elementos particulares que ayuden a las capacitaciones. Se realiza mediante cuestionarios y encuestas que se mandan de manera escrita e individual captando los temas más importantes de la plataforma *Omics*. La información recopilada debe ser analizada, cuantificada e interpretada. Sirve información referente para evaluar el alcance e impacto que han tenido las actividades organizadas.

La evaluación permite observar el cambio de conducta de los que participan y el grado en que los conocimientos, habilidades y actitudes aprendidos, se han ido plasmando en una mejor utilización de las funcionalidades en la plataforma.

La planificación de las capacitaciones se basa en las necesidades de los usuarios, contemplando los valores y emociones de los mismos, para poder inspirarlos y lograr satisfacer las expectativas que tienen para aprender, y disfrutar de las propuestas de *Omics*.



### 3. Planificación, Ejecución y Documentación de Pruebas

El siguiente plan de pruebas de software se elabora con el propósito de cumplir con los siguientes objetivos:

- ❖ Identificar la información existente en el proyecto y los componentes que deben ponerse a prueba.
- ❖ Listar los principales requisitos a probar.
- ❖ Definir las estrategias de prueba que deben emplearse.
- ❖ Identificar los recursos que necesarios.

#### 3.1. Alcance

El plan de pruebas consiste en la realización de los siguientes tipos de pruebas:

- ❖ Pruebas de validación de ingreso de datos.
- ❖ Pruebas de lógica de los módulos principales.
- ❖ Pruebas de integración entre los módulos del sistema.
- ❖ Pruebas de seguridad por niveles de usuario.
- ❖ Pruebas de carga.

Se ejecutan pruebas de unidad en cada Sprint, y dichas pruebas están documentadas usando la siguiente Plantilla:

<b>Nombre el Caso de Prueba:</b> Nombre del caso de prueba.			
<b>Tipo de caso de prueba:</b> Tipo del caso de prueba (éxito/fallo).			
<b>Caso de uso asociado:</b> Caso de uso asociado.			
<b>Fecha de creación</b>	Fecha de creación de la prueba.		
<b>Fecha de ejecución</b>	Fecha de ejecución de la prueba.		
<b>Autor</b>	Nombre del usuario que ejecuta la prueba.		
<b>Objetivo:</b> Se debe describir que se desea probar.			
<b>Precondiciones:</b> Se debe describir el estado en el que debe estar el sistema antes de ejecutar el caso de prueba.			
<b>N.º de Pasos</b>	<b>Acciones</b>	<b>Resultados Esperados</b>	<b>Resultados Obtenidos</b>

1	Los pasos que realizaría el usuario.	Los pasos que se espera que realice el sistema.	Los pasos que realiza en realidad el sistema.
<b>Estado de la prueba</b>	Describe si la prueba fue exitosa, fallida, o no pudo ser ejecutada.		
<b>Acciones correctivas</b>	Si el caso de prueba es fallido se debe indicar que acción se debe realizar para corregir el error.		
<b>Pantalla</b>	Captura de pantalla, de la ejecución del caso de prueba.		

Tabla 1. 6 Planilla base de pruebas.

### 3.2. Planificación de pruebas

#### 3.2.1. Pruebas de Validación de Ingreso de Datos

**Se realizan:**

Para poder detectar fallas en cada uno de los componentes que permiten el ingreso de los datos al Sistema. Ejemplo: De formato de los datos, de racionalidad (cuando un dato que ingresa depende de otro que está ingresando en el mismo formulario), de valores límite inferior o superior o de restricciones de valores según la regla del negocio, o de existencia del dato en la base de datos, etc.

**Objetivo:**

Validar que se ha especificado lo que el usuario realmente quería. Además, se comprueba si los valores que ha introducido el usuario cumplen las restricciones impuestas por la aplicación.

**Estrategia:**

Detectar defectos probando (testeando) el sistema.

**Periodo de ejecución:**

Se ejecutan a medida que se finaliza cada módulo.

**Recursos:**

Las pruebas son ejecutadas por los usuarios del sistema.

**Resultados esperados:**

- Validación correcta de los datos ingresados.

### 3.2.2. Pruebas de Lógica de los Módulos Principales

**Se realizan:**

Para poder detectar fallas en cada una de las componentes más importantes del Sistema, que representan las reglas del negocio principales o requisitos tecnológicos destacados. Ejemplo: Pueden ser pruebas respecto de procesos, de cálculos, de generación de información, de regla de negocio, de funciones específicas importantes del Sistema, de persistencia en las bases de datos, de controles programados, de automatización, de generación de interfaces, de integración de APIs, de transferencia automática de archivos o transacciones, de generación de alertas, de solicitud de autorización en línea, de Gateways a otros sistemas, de triggers, etc.

**Objetivo:**

Lograr combinar módulos para que trabajen correctamente de forma conjunta.

**Estrategia:**

Se comienza con módulos que no requieren de ningún otro para ejecutar, y luego se sigue con módulos que si requieren a otro para ejecutar.

**Periodo de ejecución:**

Se ejecutan de acuerdo con el grado de importancia de cada módulo, en orden ascendente.

**Recursos:**

Las pruebas son ejecutadas por los desarrolladores que participaron en cada módulo.

**Resultados esperados:**

- Interacción correcta entre los módulos principales del sistema.

### 3.2.3. Pruebas de Integración entre los Módulos del Sistema

**Se realizan:**

Se verifica que los distintos módulos se comunican e interactúan correctamente y el sistema una vez integrado cumple con las funciones especificadas de forma adecuada.

**Objetivo:**

El objetivo de las pruebas de integración entre módulos del Sistema es encontrar fallas en el Sistema y verificar el correcto ensamblaje entre los distintos componentes

con el fin de comprobar que interactúan correctamente a través de sus interfaces, tanto internas como externas, cubren la funcionalidad establecida y se ajustan a los requisitos no funcionales especificados en los alcances del sistema.

**Estrategia:**

Comprobar las funcionalidades en orden ascendente de complejidad, partiendo de línea base con la integración correcta entre módulos.

**Periodo de ejecución:**

Se ejecutan a medida que se hayan terminado los módulos involucrados en la prueba.

**Recursos:**

Las pruebas son ejecutadas por los desarrolladores que participaron en cada módulo.

**Resultados esperados:**

- Interacción correcta entre componentes de distintos módulos.

### 3.2.4. Pruebas de Seguridad por Niveles de Usuario

**Se realizan:**

Para poder encontrar fallas en el módulo de seguridad desarrollado y comprobar la integración con el resto del sistema de información, verificando el funcionamiento correcto de las interfaces que lo componen y el resto de los módulos con los que se comunica. Ejemplo: Ingreso de usuarios, contraseñas correctas e incorrectas (longitud, vencimiento) comprobar algunos de los privilegios asignados a determinados tipos de usuarios (roles), funciones que pueden realizar, etc.

**Objetivo:**

Comprobar que cada usuario sólo puede acceder a las funcionalidades pertinentes para ese rol, evitando accesos no deseados a funciones que no debe poder ejecutar. Y solo permitir al usuario administrador ejecutar funciones propias de la configuración general del sistema.

**Estrategia:**

Identificar cada tipo de usuario, las funciones y datos a los que se debe autorizar. Se crearán pruebas de acceso a funcionalidades para cada tipo de usuario.

**Periodo de ejecución:**

Se realizan a la par de las pruebas de integración, comprobando que el usuario sólo pueda acceder en la interfaz a las funciones deseadas.

**Recursos:**

Son realizadas por el programador front-end, que limitará las opciones disponibles en la interfaz y por el programador back-end.

**Resultados esperados:**

- Visualización de funciones adecuadas al rol.

### 3.2.5. Pruebas de Carga

**Se realizan:**

Para poder encontrar fallas en el funcionamiento simulado del Sistema, en lo que hace carga máxima o en situaciones de stress o límite, respecto de sesiones simultáneas, VPNs, acceso a datos, autorizaciones en línea, tasas de transferencia, rendimiento de código, rendimiento de APIs, tiempo de respuesta transaccional, servidores, servicios, integración con otros Sistemas, etc.

**Objetivo:**

Validar el rendimiento de funciones críticas en términos de complejidad de procesamiento, utilización de recurso o cantidad de usuarios en línea.

**Estrategia:**

- Se definen umbrales para el funcionamiento de funciones críticas.
- Que cada condición tenga por lo menos una vez un resultado positivo y/o negativo.
- Que cada bucle del sistema se pueda probar.

**Periodo de ejecución:**

Estas pruebas requieren el funcionamiento de módulos distintos, por lo que se realizarán al finalizar las pruebas de integración.

**Recursos:**

Las pruebas son ejecutadas por el líder de proyecto, en conjunto con el cliente, que tendrá una vista más clara en cuanto a los criterios de aceptación.

**Resultados esperados:**

- Debe ser un resultado que debe cumplir con los criterios de tiempo y rendimiento de cada prueba.

### 3.2.6. Estimación de tiempos

Para una mayor comprensión de las fechas establecidas de cada prueba mencionada anteriormente [VER ANEXO N.º 1 DIAGRAMA DE GANTT](#).

## 3.3. Casos de prueba

### Introducción:

Debido a que se sigue una metodología ágil utilizando Domain-Driven Design para el desarrollo de la plataforma, y Test-Driven Design para realizar testing unitario sobre los componentes básicos o constructos fundamentales.

Toda entidad principal, denominada Aggregate Root en el contexto de DDD, no se compone de propiedades de tipo de datos primitivos (cadena de texto, número, etc.) sino de "objetos de valor" (Value Objects). Un Value Object es una entidad con una única propiedad que envuelve a un valor primitivo. De esta forma, un objeto de valor valida en el momento de su instanciación que el valor interno que va a contener sea correcto. Así, se asegura que toda entidad principal, compuesta de Value Objects como propiedades, siempre se encuentre en un estado consistente con datos válidos.

Esto, además facilita el testing debido a que no se tienen que realizar validaciones en forma de tests de integración para comprobar que se haga una validación correcta de datos de entrada, sino que se pueden ejecutar tests unitarios directamente sobre cada objeto de valor para validar que no acepten datos primitivos que dejarían a la entidad en un estado inconsistente.

Por ejemplo, la entidad Usuario tiene un nombre de usuario asignado y, aplicando metodología DDD, este nombre de usuario no será una propiedad del tipo cadena de texto (dato primitivo), sino un Value Object denominado Username. Este será representado por una clase individual cuyo constructor se encarga de validar que su valor interno sea consistente.

Actualmente, el backend de la plataforma de Omics tiene 142 tests que verifican algunas partes del sistema críticas en términos de integración y, sobre todo, validan los datos que pueden tomar los Value Objects utilizados por las diversas entidades que componen el negocio.

Estos tests validan la consistencia de los principales módulos o contextos (Bounded Contexts): núcleo (entidades y lógica principales utilizadas en todo el sistema, como el bus de eventos y la generación de errores), identidad (autorización y autenticación, modelo principal de usuarios), publicación (todo lo referido a publicaciones, colecciones, categorías, autores y lectores, lo que podríamos contemplar como la razón de ser de la plataforma), pagos (bounded context complementario, de

apoyo, que permite obtener un beneficio o rédito económico tanto para Omics como para los creadores de contenido).

### Pruebas Generales realizadas:

```
Finished test [unoptimized + debuginfo] target(s) in 0.11s
Running target/debug/deps/common-73c08408b2d6128c

running 33 tests
test error::tests::basic ... ok
test counter::tests::test ... ok
test error::tests::wrap ... ok
test error::tests::merge ... ok
test infrastructure::cache::inmem::tests::initialize ... ok
test infrastructure::cache::inmem::tests::get_set_delete ... ok
test infrastructure::cache::inmem::tests::find ... ok
test infrastructure::cache::inmem::tests::filter ... ok
test infrastructure::event::inmem_event_bus::tests::create ... ok
test mocks::counter::tests::test ... ok
test mocks::calls_tracker::tests::test ... ok
test model::aggregate_root::tests::create ... ok
test model::aggregate_root::tests::equals ... ok
test model::aggregate_root::tests::properties ... ok
test model::id::tests::crete ... ok
test model::events::tests::events ... ok
test model::status_history::tests::compare ... ok
test model::status_history::tests::create ... ok
test model::status_history::tests::history ... ok
test model::status_history::tests::deserialize ... ok
test request::include::tests::basic ... ok
test request::include::tests::none_include ... ok
test model::status_history::tests::serialize ... ok
test request::include::tests::to_lowercase ... ok
test sql::where_builder::tests::null_params ... ok
test request::include::tests::with_witespace ... ok
test sql::where_builder::tests::without_null_params ... ok
test infrastructure::event::inmem_event_bus::tests::polymorphic ... ok
test infrastructure::event::inmem_event_bus::tests::multiple_publishers ... ok
test infrastructure::event::inmem_event_bus::tests::errors ... ok
test infrastructure::event::inmem_event_bus::tests::publish_all ... ok
test infrastructure::event::inmem_event_bus::tests::match_topics_with_regex ... ok
test infrastructure::event::inmem_event_bus::tests::publish_subscribe ... ok
```

Figura 1. 112 Test del Núcleo del sistema

```
Running target/debug/deps/identity-76989c7aaeb5e93
running 53 tests
test application::user::change_password::tests::invalid_password ... ok
test application::user::change_password::tests::success ... ok
test application::user::delete::tests::delete ... ok
test application::user::get_by_id::tests::not_owner ... ok
test application::user::get_by_id::tests::without_fullname ... ok
test application::user::get_by_id::tests::owner ... ok
test application::user::recover_password::tests::non_existing_user ... ok
test application::user::get_by_id::tests::with_fullname ... ok
test application::user::recover_password::tests::password_recovery_code_generated ... ok
test application::user::login::tests::not_validated_user ... ok
test application::user::get_by_id::tests::admin_not_owner ... ok
test application::user::update::tests::non_existing_user ... ok
test application::user::register::tests::new_user ... ok
test application::user::update::tests::invalid_data ... ok
test domain::role::permission::tests::permission_contains ... ok
test domain::role::tests::create_role ... ok
test application::user::update::tests::valid_data ... ok
test application::user::register::tests::invalid_data ... ok
test application::user::validate::tests::valid_code ... ok
test application::user::validate::tests::invalid_code ... ok
test domain::role::tests::permissions ... ok
test domain::user::birthdate::tests::from_str ... ok
test domain::user::birthdate::tests::too_old ... ok
test domain::role::permission::tests::create_permissions ... ok
test domain::token::service::tests::create_validate_invalidate ... ok
test application::user::login::tests::validated_user ... ok
test domain::user::email::tests::create ... ok
test domain::user::fullname::tests::create ... ok
test domain::user::authorization_service::tests::authorize ... ok
test domain::user::birthdate::tests::too_young ... ok
test domain::user::password::tests::invalid ... ok
test domain::user::service::tests::change_password ... ok
test domain::user::service::tests::generate_password ... ok
test domain::user::email::tests::valid ... ok
test domain::user::fullname::tests::invalid ... ok
test domain::user::tests::create ... ok
test domain::user::tests::delete ... ok
test domain::user::tests::validate ... ok
test domain::user::email::tests::invalid ... ok
test infrastructure::persistence::inmem::token_repository::tests::test ... ok
test application::user::register::tests::existing_user ... ok
test domain::user::fullname::tests::valid ... ok
test domain::user::authentication_service::tests::authenticate ... ok
test domain::user::username::tests::create ... ok
test infrastructure::service::jwt_encoder::tests::encode_decode ... ok
test infrastructure::persistence::inmem::user_repository::tests::next_id ... ok
test domain::user::service::tests::check_availability ... ok
test infrastructure::persistence::inmem::user_repository::tests::find_by_id ... ok
test mocks::token_encoder::tests::test ... ok
test mocks::password_hasher::tests::test ... ok
test domain::user::username::tests::invalid ... ok
test domain::user::username::tests::valid ... ok
test infrastructure::service::bcrypt_hasher::tests::hash_compare ... ok
```

Figura 1. 113 Test de Identidad: autenticación, autorización, usuario y roles.

```
Running target/debug/deps/omics-d3be445e6bd022ad
running 3 tests
test authorization::tests::invalid_token ... ok
test error::tests::with_internal_error ... ok
test authorization::tests::valid_token ... ok
```

Figura 1. 114 Test Utilitarios para los controladores de la API REST



```
Running target/debug/deps/payment-f0c8a32d6cc5f43d

running 15 tests
test domain::contract::tests::already_has_a_contract ... ok
test domain::payment::tests::deserialize ... ok
test domain::payment::tests::is_current ... ok
test domain::contract::tests::valid ... ok
test domain::contract::service::tests::calculate_summaries ... ok
test domain::subscription::tests::free_plan ... ok
test domain::payment::tests::serialize ... ok
test domain::subscription::tests::create ... ok
test domain::subscription::tests::pay ... ok
test domain::contract::tests::cancel ... ok
test domain::contract::tests::approve ... ok
test domain::contract::tests::reject ... ok
test domain::contract::tests::pay_unpaid_summaries ... ok
test domain::contract::tests::add_summaries ... ok
test domain::subscription::tests::subscription_status_from_payments_and_change_plan ... ok
```

Figura 1. 115 Test de pagos


```
Running target/debug/deps/publishing-a1b97f773cd90827

running 38 tests
test application::collection::update::tests::non_existing_category ... ok
test application::collection::update::tests::not_owner ... ok
test application::publication::create::tests::invalid_data ... ok
test application::collection::delete::tests::invalid ... ok
test application::collection::delete::tests::valid ... ok
test application::collection::update::tests::update ... ok
test application::publication::delete::tests::invalid ... ok
test application::publication::create::tests::not_existing_category ... ok
test application::publication::delete::tests::valid ... ok
test application::publication::create::tests::valid ... ok
test application::publication::like::tests::invalid_ids ... ok
test application::publication::like::tests::valid ... ok
test application::publication::publish::tests::publish ... ok
test application::publication::approve::tests::approve ... ok
test application::publication::read::tests::not_published ... ok
test application::publication::get_by_id::tests::owner_view_of_draft ... ok
test application::publication::get_by_id::tests::invalid_id ... ok
test application::publication::read::tests::valid ... ok
test application::publication::get_by_id::tests::reader_interaction ... ok
test application::publication::like::tests::not_published ... ok
test application::publication::update::tests::not_owner ... ok
test application::publication::update::tests::published_publication ... ok
test application::publication::reject::tests::reject ... ok
test domain::publication::image::tests::image ... ok
test application::publication::update_pages::tests::valid ... ok
test domain::publication::statistics::tests::prom_stars ... ok
test domain::publication::statistics::tests::view ... ok
test domain::publication::tag::tests::create ... ok
test domain::publication::tests::interaction_with_draft_publication ... ok
test domain::publication::tests::create ... ok
test application::publication::update_pages::tests::invalid ... ok
test application::publication::update::tests::valid ... ok
test domain::publication::tests::status ... ok
test application::publication::update::tests::non_existing_category ... ok
test domain::publication::tests::interaction_with_published_publication ... ok
test application::publication::read::tests::invalid_ids ... ok
test application::publication::get_by_id::tests::reader_view_of_draft ... ok
test application::publication::get_by_id::tests::reader_view_of_published ... ok
```

Figura 1. 116 Test Publicaciones, colecciones, autores, lectores y categorías.

### 3.3.1. Pruebas de Validación de Ingreso de Datos

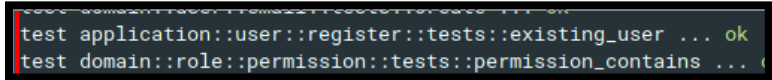
Caso de Prueba de Validación de Ingreso de Datos 01: Usuario crea cuenta en la plataforma.			
<b>Tipo de caso de prueba:</b> Caso de fallo.			
<b>Caso de uso asociado:</b> Registrar Cuenta de Usuario.			
<b>Fecha de creación</b>	10/08/2020		
<b>Fecha de ejecución</b>	05/09/2020		
<b>Autor</b>	Usuario.		
<b>Objetivo:</b> Comprobar que un usuario al querer crearse una cuenta en la plataforma falla al no ingresar de forma correcta los datos de registración.			
<b>Precondiciones:</b>			
<ul style="list-style-type: none"> <li>• Usuario no registrado en la plataforma.</li> <li>• Rol de "usuario" instanciado.</li> </ul>			
N.º de Pasos	Acciones	Resultados Esperados	Resultados Obtenidos
1	Usuario selecciona la opción "Registrarse".	Se muestra en pantalla la opción para registrarse utilizando: ❖ E-mail.	Sistema espera que se seleccione la opción de registración. ❖ E-mail.
2	Usuario selecciona opción registrarse por e-mail.	Se despliega el formulario de crear cuenta de usuario para completar con los datos personales: ❖ E-mail. ❖ Nombre de usuario. ❖ Contraseña. ❖ Reescribir contraseña.	Campos habilitados para completar con datos personales y crear cuenta de usuario. ❖ E-mail. ❖ Nombre de usuario. ❖ Contraseña. ❖ Reescribir contraseña.

3	Usuario ingresa el e-mail = <a href="mailto:mariobross1234@gmail.com">mariobross1234@gmail.com</a> .	Se valida si el e-mail = <a href="mailto:mariobross1234@gmail.com">mariobross1234@gmail.com</a> tiene formato correcto y que no exista en el sistema.	El e-mail = <a href="mailto:mariobross1234@gmail.com">mariobross1234@gmail.com</a> es correcto, y no existe en el sistema.
4	Usuario ingresa nombre de usuario: "mario bross"	Se valida si el nombre de usuario: "mario bross" tiene el formato correcto y que no exista en el sistema.	Se muestra mensaje de error: "Error nombre de usuario- Los nombres de usuario no pueden tener espacios".
<b>Estado de la prueba</b>	Fallido.		
<b>Acciones correctivas</b>	Para corregir el error se debe ingresar un nombre de usuario que no tenga espacios. Ejemplo: nombre de usuario válido: "mariobross".		
<b>Pantalla</b>	 <pre> #[test] fn valid() {     assert!(Username::new("username").is_ok());     assert!(Username::new("user123").is_ok());     assert!(Username::new("11user").is_ok());     assert!(Username::new("username.123").is_ok());     assert!(Username::new("user-123").is_ok());     assert!(Username::new("user_123").is_ok());     assert!(Username::new("USER.789").is_ok());     assert!(Username::new("USER.name").is_ok()); }  #[test] fn invalid() {     assert!(Username::new("user name").is_err());     assert!(Username::new("úser").is_err());     assert!(Username::new("usër").is_err());     assert!(Username::new("@user").is_err());     assert!(Username::new("us/er").is_err());     assert!(Username::new("-user").is_err());     assert!(Username::new("_user").is_err());     assert!(Username::new("user-").is_err());     assert!(Username::new("user_").is_err());     assert!(Username::new("user.").is_err());     assert!(Username::new(".user").is_err());     assert!(Username::new("-_user").is_err());     assert!(Username::new("user-.").is_err());     assert!(Username::new("-.").is_err()); }           </pre>		

	<pre>test infrastructure::persistence::inmem::user_reposi test domain::user::username::tests::valid ... ok test domain::user::authentication_service::tests::au test domain::user::username::tests::invalid ... ok test infrastructure::service::jwt_encoder::tests::en</pre> <p>Se comprueba que los nombres de usuario aceptables son realmente válidos y que los nombres inválidos retornan un error. Entre estos últimos se encuentra, como primer test, un nombre de usuario con un espacio.</p>
--	---

Tabla 1. 7 Caso de Prueba de Validación de Ingreso de Datos 01.

Caso de Prueba de Validación de Ingreso de Datos 02: Usuario crea cuenta en la plataforma.			
<b>Tipo de caso de prueba:</b> Caso de fallo.			
<b>Caso de uso asociado:</b> Registrar Cuenta de Usuario.			
<b>Fecha de creación</b>	13/08/2020		
<b>Fecha de ejecución</b>	07/09/2020		
<b>Autor</b>	Usuario.		
<b>Objetivo:</b> Comprobar que un usuario al querer crearse una cuenta en la plataforma falla al no ingresar de forma correcta los datos de registración.			
<b>Precondiciones:</b> <ul style="list-style-type: none"> <li>• Usuario no registrado en la plataforma.</li> <li>• Rol de "usuario" instanciado.</li> </ul>			
N.º de Pasos	Acciones	Resultados Esperados	Resultados Obtenidos
1	Usuario selecciona la opción: "Registrarse".	Se muestra en pantalla la opción para registrarse utilizando: ❖ E-mail.	Sistema espera se selecciona la opción de registración: ❖ E-mail.
2	Usuario selecciona la opción de	Se despliega el formulario de crear cuenta de usuario para	Campos habilitados para completar con datos personales, y


	<p>registro: "E-mail".</p>	<p>completar con los datos personales:</p> <ul style="list-style-type: none"> <li>❖ E-mail.</li> <li>❖ Nombre de usuario.</li> <li>❖ Contraseña.</li> <li>❖ Reescribir contraseña.</li> </ul>	<p>crear cuenta de usuario.</p> <ul style="list-style-type: none"> <li>❖ E-mail.</li> <li>❖ Nombre de usuario.</li> <li>❖ Contraseña.</li> <li>❖ Reescribir contraseña.</li> </ul>
3	<p>Usuario ingresa E-mail = <a href="mailto:mariobross@gmail.com">mariobross@gmail.com</a>.</p>	<p>Se valida si el E-mail = <a href="mailto:mariobross@gmail.com">mariobross@gmail.com</a> tiene el formato correcto, y no existe en el sistema.</p>	<p>Se muestra mensaje de error: "Error al registrarse- El correo no está disponible".</p>
<b>Estado de la prueba</b>	Fallido.		
<b>Acciones correctivas</b>	<p>Para corregir el error se debe ingresar otro E-mail que no esté en uso en el sistema.</p> <p>Ejemplo: E-mail = <a href="mailto:mariobross1234@gmail.com">mariobross1234@gmail.com</a>.</p>		
<b>Pantalla</b>			

	<pre> #[tokio::test] async fn existing_user() {     let c = mocks::container();     let uc = Register::new(c.event_pub(), c.user_repo(), c.user_serv());      let mut user = mocks::user(         "user-1",         "username",         "user@omics.com",         "P@asswd!",         true,         None,         None,         "user",     );     c.user_repo().save(&amp;mut user).await.unwrap();      assert!(uc         .exec(RegisterCommand {             username: user.identity().username().to_string(),             email: user.identity().email().to_string(),             password: "P@asswd!".to_owned(),         })         .await         .is_err());      assert!(uc         .exec(RegisterCommand {             username: "other".to_owned(),             email: user.identity().email().to_string(),             password: "P@asswd!".to_owned(),         })         .await         .is_err());      assert!(uc         .exec(RegisterCommand {             username: user.identity().username().to_string(),             email: "other@other.com".to_owned(),             password: "P@asswd!".to_owned(),         })         .await         .is_err());     }         </pre> <p>En el test anterior, se observa que se corrobora que dentro de la funcionalidad de registrar usuario no pueda registrarse un nuevo usuario si su nombre de usuario o su dirección de correo electrónico ya está en uso. En el segundo "assert" se valida que al llamar al caso de uso (lógica en capa de aplicación) "Register" con un email ya utilizado (usuario existente) éste retorna un error.</p>
--	---

Tabla 1. 8 Caso de Prueba de Validación de Ingreso de Datos 02.

<b>Caso de Prueba de Validación de Ingreso de Datos 03:</b> Usuario quiere registrarse en la plataforma.	
<b>Tipo de caso de prueba:</b> Caso de éxito.	
<b>Caso de uso asociado:</b> Registrar Cuenta de Usuario.	
<b>Fecha de creación</b>	14/08/2020
<b>Fecha de ejecución</b>	09/09/2020

<b>Autor</b>	Usuario.		
<b>Objetivo:</b> Comprobar que un usuario, al querer crearse una cuenta en la plataforma, lo logra con éxito debido a que ingresa de forma correcta todos los datos de registración.			
<b>Precondiciones:</b>			
<ul style="list-style-type: none"> <li>• Usuario no registrado en la plataforma.</li> <li>• Rol de "usuario" instanciado.</li> </ul>			
<b>N.º de Pasos</b>	<b>Acciones</b>	<b>Resultados Esperados</b>	<b>Resultados Obtenidos</b>
1	Usuario selecciona la opción "Registrarse".	Se muestra en pantalla la opción para registrarse utilizando: ❖ E-mail.	Sistema espera se selecciona la opción de registración: ❖ E-mail.
2	Usuario selecciona la opción de registración: "E-mail".	Se despliega el formulario de crear cuenta de usuario para completar con los datos personales: ❖ E-mail. ❖ Nombre de usuario. ❖ Contraseña. ❖ Reescribir contraseña.	Campos habilitados para completar con datos personales, y crear cuenta de usuario. ❖ E-mail. ❖ Nombre de usuario. ❖ Contraseña. ❖ Reescribir contraseña.
3	Usuario ingresa E-mail = <a href="mailto:mariobross1234@gmail.com">mariobross1234@gmail.com</a>	Se valida si el E-mail = <a href="mailto:mariobross1234@gmail.com">mariobross1234@gmail.com</a> tiene el formato correcto y que no existe en el sistema.	El e-mail = <a href="mailto:mariobross1234@gmail.com">mariobross1234@gmail.com</a> es correcto, y no existe en el sistema.
4	Usuario ingresa nombre de usuario = "mariobross".	Validar si el nombre de usuario = "mariobross" tiene el formato correcto y que no existe en el sistema.	El nombre de usuario = "mariobross" es correcto, y no existe en el sistema.

5	Usuario ingresa contraseña = "Thor8889".	Validar si la contraseña = "Thor8889" tiene el formato correcto.	La contraseña= "Thor8889" tiene el formato correcto.
6	Usuario reescribe la contraseña = "Thor8889".	Validar si la contraseña reescrita= "Thor8889", se ha escrito de manera idéntica que la contraseña ingresada anteriormente.	La contraseña= "Thor8889" se ha escrito de manera idéntica que la contraseña ingresada anteriormente.
7	El usuario selecciona la opción: "Confirmar".	Se aceptan los términos de servicios y política de privacidad.	Se muestra mensaje: "Registro exitoso- Revisa tu correo para confirmar tu cuenta y comenzar a usar la plataforma Omics".
<b>Estado de la prueba</b>	Exitosa.		
<b>Acciones correctivas</b>	-----		
<b>Pantalla</b>	 <pre> #[tokio::test] async fn new_user() {     let c = mocks::container();     let uc = Register::new(c.event_pub(), c.user_repo(), c.user_serv());      let cmd = RegisterCommand {         username: "new-user".to_owned(),         email: "new@user.com".to_owned(),         password: "P@asswd!".to_owned(),     };      let res = uc.exec(cmd).await.unwrap();     let saved_user = c         .user_repo()         .find_by_id(&amp;UserId::new(&amp;res.id).unwrap())         .await         .unwrap();     assert_eq!(res.id, saved_user.base().id().value());     assert_eq!(saved_user.identity().username().value(), "new-user");     assert_eq!(saved_user.identity().email().value(), "new@user.com");     assert_ne!(         saved_user.identity().password().unwrap().value(),         "P@asswd!"     );      assert_eq!(c.event_pub().events().await.len(), 1); } </pre>		



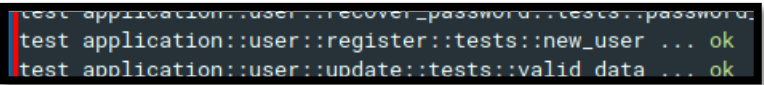
	 <pre>test application::user::recover_password::tests::password ... ok test application::user::register::tests::new_user ... ok test application::user::update::tests::valid data ... ok</pre> <p>Se comprueba que un nuevo usuario, con nombre de usuario, dirección de correo electrónico y contraseña válidos (estos ya verificados por los Value Objects anteriormente mencionados) puede ser registrado llamando al caso de uso "Register" y que, a su vez, este nuevo usuario es almacenado en la base de datos (mediante el uso de un UserRepository, otro de los constructos de DDD) correctamente.</p> <p>Por último, vale la pena mencionar que también se valida que al crear un nuevo usuario se emite un evento denominado UserRegistered indicando que un nuevo usuario ha sido registrado. Este evento es utilizado por otras partes independientes del sistema, como por ejemplo, el módulo de notificaciones, para enviar un email de bienvenida.</p>
--	--

Tabla 1. 9 Caso de Prueba de Validación de Ingreso de Datos 03.

### 3.3.2. Pruebas de Lógica de los Módulos Principales

Caso de Prueba de Lógica de los Módulos Principales 01: Usuario Paga Suscripción.			
<b>Tipo de caso de prueba:</b> Caso de fallo.			
<b>Caso de uso asociado:</b> Pagar Suscripción.			
<b>Fecha de creación</b>	12/08/2020		
<b>Fecha de ejecución</b>	19/09/2020		
<b>Autor</b>	Usuario.		
<b>Objetivo:</b> Comprobar que el usuario no puede realizar el pago de la Suscripción porque tiene saldo insuficiente.			
<b>Precondiciones:</b>			
<ul style="list-style-type: none"> <li>• Usuario registrado instanciado.</li> <li>• Estados de Suscripción existentes.</li> <li>• Suscripción en estado = "Esperando pago".</li> </ul>			
N.º de Pasos	Acciones	Resultados Esperados	Resultados Obtenidos

1	<p>El usuario "Chuck Norris" selecciona la opción "Suscribirse".</p>	<p>Que se visualice la interfaz con los medios de pago de Mercado Pago:</p> <ul style="list-style-type: none"> <li>❖ Dinero en mercado pago.</li> <li>❖ Tarjeta de débito.</li> <li>❖ Tarjeta de crédito.</li> </ul> <p>El monto del total a pagar.</p>	<p>Se despliega la interfaz:</p> <p>"Introduce tu medio de pago:</p> <ul style="list-style-type: none"> <li>❖ Dinero en mercado pago.</li> <li>❖ Tarjeta de débito.</li> <li>❖ Tarjeta de crédito."</li> </ul> <p>"Total a pagar= \$150".</p>
2	<p>El usuario "Chuck Norris" selecciona el medio de pago = "Tarjeta de crédito".</p>	<p>Que se habiliten los campos para llenar los datos del pago:</p> <ul style="list-style-type: none"> <li>❖ Nombre.</li> <li>❖ Número de tarjeta.</li> <li>❖ Fecha de vencimiento.</li> <li>❖ CVC.</li> </ul>	<p>Campos habilitados para completar con datos personales:</p> <ul style="list-style-type: none"> <li>❖ Nombre.</li> <li>❖ Número de tarjeta.</li> <li>❖ Fecha de vencimiento.</li> <li>❖ CVC.</li> </ul> <p>Se visualiza el botón "Confirmar".</p>
3	<p>El usuario "Chuck Norris" ingresa los datos de pago:</p> <ul style="list-style-type: none"> <li>❖ Nombre = "Chuck Norris".</li> <li>❖ Número de tarjeta = "5468 4256 1235 4568".</li> <li>❖ Fecha de vencimiento = "10/19".</li> <li>❖ CVC = "254".</li> </ul>	<p>El sistema se comunica con la API externa de Mercado Pago.</p>	<p>Mensaje de error: "Algo salió mal... Tu tarjeta no tiene límite suficiente".</p>

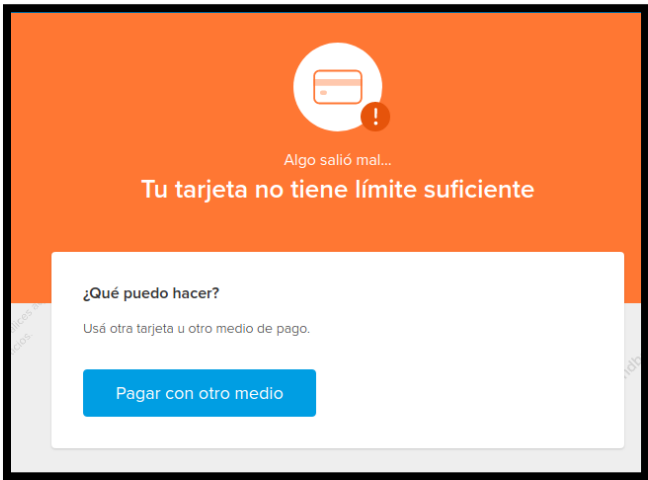
	Y selecciona el botón "Confirmar".		
4	El usuario "Chuck Norris" espera la verificación del pago.	El sistema recibe la información del API de Mercado pago del estado del pago.	No se puede realizar el pago de la suscripción. El estado de la suscripción queda en "Pendiente".
<b>Estado de la prueba</b>	Fallida.		
<b>Acciones correctivas</b>	Usuario debe tener saldo en la tarjeta para poder suscribirse.		
<b>Pantalla</b>	 <pre data-bbox="475 1402 1310 1518"> {   id: "342fa9bb-fff1-47d1-828a-92df26427383", ...   id: "342fa9bb-fff1-47d1-828a-92df26427383"   payment_link: "https://www.mercadopago.com.ar/checkout/v1/redirect?pref_id=17211509" } </pre> <p>El sistema retorna una URL de pago, a la cual tiene que acceder el lector mediante el navegador para llenar sus datos de pago (de forma automática). Se valida síncronamente si el medio de pago que eligió dispone de fondos.</p> <p>Si el proceso de pago de Mercado Pago falla, como en el caso anterior, por falta de fondos, nunca se recibe un evento proveniente de este servicio externo indicando que la suscripción y el pago realizado han sido validados. De esta forma, la suscripción recientemente creada nunca pasa al estado "activa", el cual indica realmente que un lector está suscripto a la plataforma, para poder acceder al contenido premium.</p>		

Tabla 1. 10 Caso de Prueba de Lógica de los Módulos Principales 01.

<b>Caso de Prueba de Lógica de los Módulos Principales 02:</b> Autor Adjunta la Portada al Publicar una Obra.			
<b>Tipo de caso de prueba:</b> Caso de fallo.			
<b>Caso de uso asociado:</b> Publicar Obra.			
<b>Fecha de creación</b>	30/07/2020		
<b>Fecha de ejecución</b>	17/09/2020		
<b>Autor</b>	Usuario.		
<b>Objetivo:</b> Comprobar que el usuario, al subir una imagen de portada, no cumple las dimensiones requeridas para publicar la obra.			
<b>Precondiciones:</b>			
<ul style="list-style-type: none"> <li>El autor debe estar logueado.</li> </ul>			
<b>N.º de Pasos</b>	<b>Acciones</b>	<b>Resultados Esperados</b>	<b>Resultados Obtenidos</b>
1	El autor "ArivIE" selecciona "Deskboard".	El sistema carga las obras y colecciones del autor "ArivIE".	Se muestra la pantalla Deskboard en la opción "Mis Comics".
2	El autor "ArivIE" selecciona el botón "Añadir Publicación".	El sistema carga las categorías a seleccionar: <ul style="list-style-type: none"> <li>Terror.</li> <li>Suspense.</li> <li>Acción.</li> </ul> Se muestran las colecciones existentes del autor "ArivIE": <ul style="list-style-type: none"> <li>"Colección de ArivIE 1".</li> <li>"Colección de ArivIE 2".</li> </ul>	Se muestra la pantalla de publicación de obra con los siguientes campos: <ul style="list-style-type: none"> <li>Portada.</li> <li>Título.</li> <li>Nombre de colección.</li> <li>Categorías existentes.</li> <li>Descripción.</li> </ul>

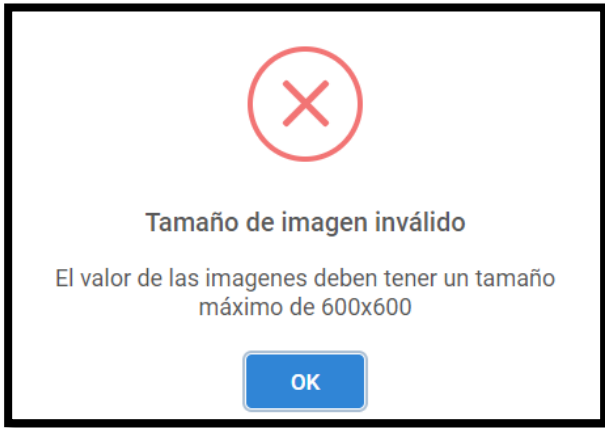

3	El autor "ArivIE" selecciona el botón "Subir imagen de portada".	Se espera mostrar una ventana para examinar en la computadora del autor "ArivIE" una imagen.	Se muestra ventana para examinar en la computadora del autor "ArivIE" una imagen.
4	El autor "ArivIE" selecciona imagen de "zAfija-01" con extensión JPG de más de 800 pixeles de ancho.	Se verifica que el formato del archivo, el peso en MB y el tamaño en pixeles sea válido.	El sistema muestra mensaje de error: "Tamaño de imagen inválido- El valor de las imágenes debe tener un tamaño máximo de 600x600".
<b>Estado de la prueba</b>	Fallida.		
<b>Acciones correctivas</b>	Para corregir el error, el autor "ArivIE" debe buscar una imagen de menor resolución.		
<b>Pantalla</b>			

Tabla 1. 11 Caso de Prueba de Lógica de los Módulos Principales 02.

<b>Caso de Prueba de Lógica de los Módulos Principales 03:</b> Administrador Aprueba Publicación.
<b>Tipo de caso de prueba:</b> Caso de éxito.
<b>Caso de uso asociado:</b> Aceptar/Rechazar Publicación.

<b>Fecha de creación</b>	11/08/2020		
<b>Fecha de ejecución</b>	05/10/2020		
<b>Autor</b>	Administrador.		
<b>Objetivo:</b> Aprobar una Publicación pendiente.			
<b>Precondiciones:</b>			
<ul style="list-style-type: none"> <li>• El Administrador debe estar logueado.</li> <li>• Estados de Publicación existentes.</li> <li>• Debe haber al menos una Publicación en estado = "Pendiente de aprobación", que cumpla con los requisitos para estar en la plataforma.</li> </ul>			
<b>N.º de Pasos</b>	<b>Acciones</b>	<b>Resultados Esperados</b>	<b>Resultados Obtenidos</b>
1	El administrador "Gustavo Dayn" selecciona: "Gestión de Contenido".	Que el sistema muestre la ventana de "Gestión de Contenido".	Se visualizan las publicaciones hasta la fecha que se encuentran en estado "Pendiente de aprobación": ❖ Public prueba 1. ❖ Public prueba 2.
2	El administrador "Gustavo Dayn" selecciona "Ver Detalle" de la publicación "Public prueba 1" del Autor "Vlad".	Que se verifique su contenido para ver si se Acepta/Rechaza la publicación "Public prueba 1" del Autor "Vlad".	Se visualiza la publicación "Public prueba 1" del Autor "Vlad" con la opción de "Aceptar" o "Rechazar" la misma.
3	El administrador "Gustavo Dayn" selecciona "Aceptar" de la publicación "Public prueba 1" del Autor "Vlad".	Que el sistema envíe un E-mail al correo = <a href="mailto:Vald56@gmail.com">Vald56@gmail.com</a> , del autor "Vlad" para notificarle que la publicación "Public	Se visualiza el mensaje: "Publicación realizada- ¡Se ha subido correctamente!".

		<p>prueba 1" ha sido aprobada y está publicada en la plataforma.</p>	<p>Se cambia el estado de la publicación "Public prueba 1" de "Esperando Aprobación" a "Publicada".</p>
<p><b>Estado de la prueba</b></p>	<p>Exitosa.</p>		
<p><b>Acciones correctivas</b></p>	<p>-----</p>		
<p><b>Pantalla</b></p>	 <pre> 4  #[tokio::test] 5  async fn approve() { 6      let c = mocks::container(); 7      let uc = Approve::new(c.event_pub(), c.publication_repo(), c.user_repo()); 8 9      let mut user = identity_mocks::user( 10         "#content-manager01", 11         "content-manager-1", 12         "content-manager@omics.com", 13         "P@sswd!", 14         true, 15         None, 16         None, 17         "content-manager", 18     ); 19     c.user_repo().save(&amp;mut user).await.unwrap(); 20 21     let mut publication = mocks::publication( 22         "#publication01", 23         "#user01", 24         "Publication 01", 25         "category-1", 26         vec!["Tag 1", "Tag 2"], 27         "domain.com/cover.jpg", 28         3, 29         true, 30         false, 31         false, 32     ); 33     c.publication_repo().save(&amp;mut publication).await.unwrap(); 34 35     uc.exec( 36         user.base().id().to_string(), 37         publication.base().id().to_string(), 38         ApproveCommand { 39             comment: "All is OK".to_owned(), 40         }, 41     ) 42     .await 43     .unwrap(); 44 45     let publication = c 46         .publication_repo() 47         .find_by_id(publication.base().id()) 48         .await     </pre> <pre> test application::publication::like::tests::valid ... ok test application::publication::approve::tests::approve ... ok test application::publication::read::tests::invalid_ids ... ok     </pre>		

	<p>Se verifica que el caso de uso para aprobar una publicación sea correcto. Esto implica corroborar que el usuario tiene el rol de gestor de contenido o administrador asignado.</p> <p>Además, se puede realizar testing manual para mayor seguridad:</p> <p>Un usuario sin roles de administrador ni gestor de contenido crea una nueva publicación, al cual se coloca en estado de "borrador":</p> <pre> 1 { 2   "publication": { 3     "id": "7188baf9-0961-44e4-95ff-cd4ba7ce221a", 4     "author_id": "be8f97bb-2066-4bbc-9787-1f0c275a7e49", 5     "author": null, 6     "name": "Publication 2", 7     "synopsis": "Synopsis...", 8     "category_id": "adventure", 9     "category": null, 10    "tags": [ 11      "Tag 1", 12      "Tag 2" 13    ], 14    "cover": "domain.com/image.jpg", 15    "statistics": { 16      "views": 0, 17      "unique_views": 0, 18      "readings": 0, 19      "likes": 0, 20      "reviews": 0, 21      "stars": 0.0 22    }, 23    "pages": [], 24    "contract": false, 25    "status": { 26      "status": "draft", 27      "changed_at": "2020-10-26T05:33:35.860542956+00:00", 28      "changed_by": null, 29      "comment": null 30    }, 31    "created_at": "2020-10-26T05:33:35.860540+00:00", 32    "updated_at": null 33  }, 34  "reader": null 35 } </pre>
--	---



Tabla 1. 12 Caso de Prueba de Lógica de los Módulos Principales 03.

### 3.3.3. Pruebas de Integración entre los Módulos del Sistema

<p><b>Caso de Prueba de integración entre los Módulos del Sistema 01:</b> Colecciones y Categorías Existentes.</p>
<p><b>Tipo de caso de prueba:</b> Caso de éxito.</p>
<p><b>Caso de uso asociado:</b> Gestionar Categorías.</p>
<p><b>Módulos que se integran:</b></p> <ul style="list-style-type: none"> <li>• Módulo de Configuración.</li> <li>• Base de datos.</li> </ul>



<b>Fecha de creación</b>	17/07/2020		
<b>Fecha de ejecución</b>	14/09/2020		
<b>Autor</b>	Administrador.		
<b>Objetivo:</b> El administrador quiere gestionar las Categorías de las Publicaciones.			
<b>Precondiciones:</b>			
<ul style="list-style-type: none"> <li>• Administrador instanciado.</li> <li>• Categorías instanciadas.</li> </ul>			
<b>N.º de Pasos</b>	<b>Acciones</b>	<b>Resultados Esperados</b>	<b>Resultados Obtenidos</b>
1	El administrador "Sephirot" selecciona: "Gestionar Categorías".	<p>El sistema muestra un formulario con todas las Categorías existentes.</p> <p>Categorías:</p> <ul style="list-style-type: none"> <li>❖ Drama.</li> <li>❖ Romance.</li> <li>❖ Comedia.</li> <li>❖ Terror.</li> <li>❖ Superhéroes.</li> <li>❖ Fantasía.</li> <li>❖ Vida diaria.</li> <li>❖ Sci-Fi.</li> <li>❖ Acción.</li> </ul>	<p>Se visualizan las Categorías:</p> <ul style="list-style-type: none"> <li>❖ Drama.</li> <li>❖ Romance.</li> <li>❖ Comedia.</li> <li>❖ Terror.</li> <li>❖ Superhéroes.</li> <li>❖ Fantasía.</li> <li>❖ Vida diaria.</li> <li>❖ Sci-Fi.</li> <li>❖ Acción.</li> </ul>
2	El administrador "Sephirot" selecciona "Editar" en la categoría = "Acción".	El sistema muestra una pantalla para editar la categoría = "Acción".	Se visualiza el valor ID = 25689, y nombre de la categoría = "Acción".
3	El administrador "Sephirot" cambia el nombre de la categoría =	El sistema muestra la opción de cambios realizados.	Se verifica que el valor ingresado = "Acción y Aventura" sea válido.

	"Acción", por "Acción y Aventura".		
4	El administrador "Sephrot" selecciona el botón "Guardar".	El sistema muestra un mensaje: "Modificado- La categoría Acción y Aventura ha sido modificada".	Se guarda en el servidor la modificación realizada en la categoría = "Acción y Aventura", con el ID = 25689 seleccionado.
<b>Estado de la prueba</b>	Exitosa.		
<b>Acciones correctivas</b>	-----		
<b>Pantalla</b>	 <pre> 1 { 2   "categories": [ 3     { 4       "id": "action", 5       "name": "Acción", 6       "created_at": "2020-10-26T00:37:45.221825+00:00", 7       "updated_at": null 8     }, 9     { 10      "id": "adventure", 11      "name": "Aventura", 12      "created_at": "2020-10-26T00:37:45.221825+00:00", 13      "updated_at": null 14     }, 15     { 16      "id": "comedy", 17      "name": "Comedia", 18      "created_at": "2020-10-26T00:37:45.221825+00:00", 19      "updated_at": null 20     }, </pre> <p>Todos los usuarios pueden visualizar las Categorías disponibles, pero sólo el administrador puede editarlas.</p>  <pre> 1 { 2   "id": "action", 3   "name": "Acción", 4   "created_at": "2020-10-26T00:37:45.221825+00:00", 5   "updated_at": null 6 } </pre> <p>Un administrador puede crear una nueva Categoría. A la misma se le asigna un ID automáticamente.</p>		

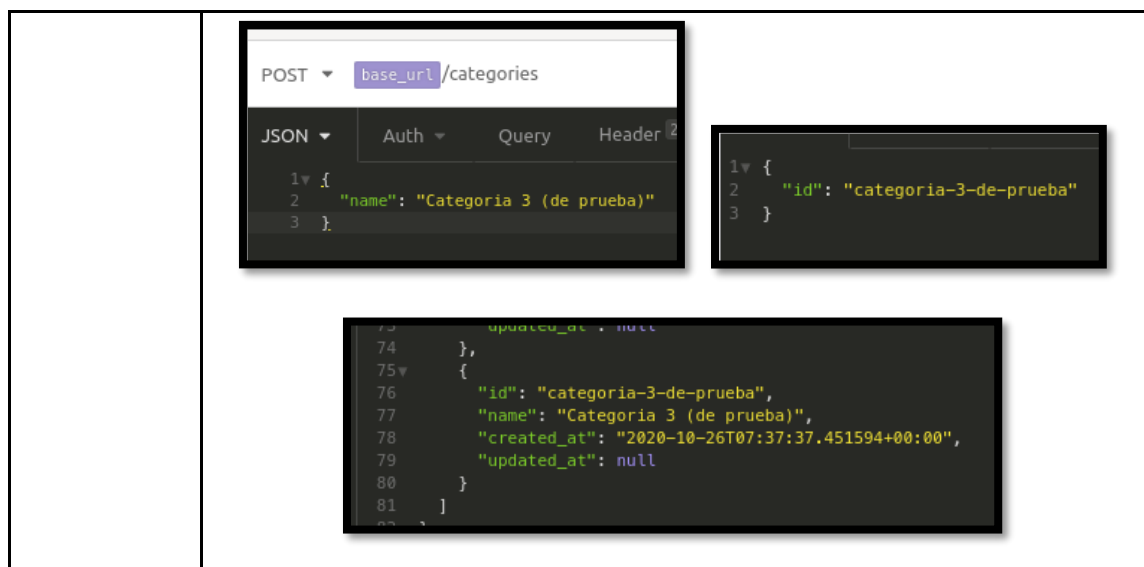


Tabla 1. 13 Caso de Prueba de integración entre los Módulos del Sistema 01.

<b>Caso de Prueba de integración entre los Módulos del Sistema 02:</b> Ver Estadística de Publicación al Dar Like.	
<b>Tipo de caso de prueba:</b> Caso de éxito.	
<b>Caso de uso asociado:</b>	
<ul style="list-style-type: none"> <li>• Ver Estadísticas de Publicación.</li> <li>• Dar like a Publicación.</li> </ul>	
<b>Módulos que se integran:</b>	
<ul style="list-style-type: none"> <li>• Módulo de Reseñas.</li> <li>• Módulo de Publicación de Obras.</li> </ul>	
<b>Fecha de creación</b>	10/08/2020
<b>Fecha de ejecución</b>	16/09/2020
<b>Autor</b>	<ul style="list-style-type: none"> <li>• Autor.</li> <li>• Lector.</li> </ul>
<b>Objetivo:</b> Comprobar la correcta interacción al dar Like a una publicación por parte del lector, y que el autor la pueda visualizar en las estadísticas de sus publicaciones.	
<b>Precondiciones:</b>	
<ul style="list-style-type: none"> <li>• Autor instanciado.</li> </ul>	

<ul style="list-style-type: none"> <li>• Lector instanciado.</li> <li>• Publicación instanciada.</li> </ul>			
N.º de Pasos	Acciones	Resultados Esperados	Resultados Obtenidos
1	El lector "Sephrot" seleccionar el botón "Like" de la publicación = "Jojo Bizarre Adventure".	El sistema muestra el botón de "Like" resaltado.	El botón de "Like" se encuentra resaltado y se le suma 1 al total de Likes de la publicación = "3 Likes" de la publicación "Jojo Bizarre Adventure".
2	El autor = "Till Lindemann" selecciona el botón "Mis Comics" en la interface "Deskboard".	El sistema muestra la interfaz con todas las obras publicadas por el autor = "Till Lindemann", con los siguientes datos: <ul style="list-style-type: none"> <li>❖ Nombre de la Publicación.</li> <li>❖ Fecha de publicación.</li> <li>❖ Likes.</li> <li>❖ Comentarios.</li> <li>❖ Números de páginas.</li> <li>❖ ID de la publicación.</li> </ul>	Se obtienen la información de todas las obras publicadas del autor = "Till Lindemann": <ul style="list-style-type: none"> <li>❖ "Jojo Bizarre Adventure", 05/06/2020, 3 likes, "Muy buena", 10 páginas, 265897.</li> <li>❖ Metal Gear, 12/05/2020, 5 likes, "Excelente", 7 páginas, 658974.</li> <li>❖ Diese, 09/06/2020, 2 likes, "Buena", 11 páginas, 123456.</li> </ul>
3	El autor = "Till Lindemann", selecciona la publicación = "Jojo Bizarre Adventure".	El sistema muestra la interfaz con la información de la Publicación = "Jojo Bizarre Adventure".	Se obtienen todos los datos de la Publicación = "Jojo Bizarre Adventure", 265897, 3 likes.
<b>Estado de la prueba</b>	Exitosa.		


<p><b>Acciones correctivas</b></p> <p>-----</p>	
<p><b>Pantalla</b></p>	 <p>The image shows a terminal window with test results and a code editor with Rust code. The test results show three tests passing: 'test application::publication::like::tests::invalid_ids ... ok', 'test application::publication::like::tests::valid ... ok', and 'test application::publication::approve::tests::approve ... ok'. The code is a Rust test function named 'valid' that sets up a mock container, creates a 'Like' entity, saves a 'reader' and a 'publication', and then uses 'uc.exec' to perform a 'like' action. Finally, it asserts that the publication's like count is 1 and that no events were emitted.</p> <pre> 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 </pre> <p>En la línea 3 se instancia el caso de uso para dar Like a una Publicación, y en la línea 26 se ejecuta (especificando el ID de lector y el ID de Publicación). La ejecución de esto conlleva a que se genera una nueva entidad Like representando esta interacción y, más importante, el contador de likes en las estadísticas de la Publicación debe incrementarse en uno, como se observa su aserción en la línea 38.</p> <p>Si la Publicación tenía 0 likes, y se ejecuta este caso de uso (integración) en forma correcta, el contador debería incrementarse en 1.</p>

Tabla 1. 14 Caso de Prueba de integración entre los Módulos del Sistema 02.

<b>Caso de Prueba de integración entre los Módulos del Sistema 03:</b> Publicar y Visualizar el Contenido de la Obra.			
<b>Tipo de caso de prueba:</b> Caso de éxito.			
<b>Caso de uso asociado:</b> <ul style="list-style-type: none"> <li>• Publicar Obra.</li> <li>• Buscar y Filtrar Contenido.</li> </ul>			
<b>Módulos que se integran:</b> <ul style="list-style-type: none"> <li>• Módulo de Publicación de Obras.</li> <li>• Módulo de Catálogo y Búsqueda.</li> </ul>			
<b>Fecha de creación</b>	19/08/2020		
<b>Fecha de ejecución</b>	18/09/2020		
<b>Autor</b>	<ul style="list-style-type: none"> <li>• Autor-Lector</li> </ul>		
<b>Objetivo:</b> Comprobar la visualización en el catálogo por parte de un Lector sin que aparezca la obra que se encuentre pendiente de aprobación por un Autor.			
<b>Precondiciones:</b> <ul style="list-style-type: none"> <li>• Autor-Lector instanciado</li> <li>• Estados de Publicación existentes.</li> <li>• Publicación en estado = "Borrador" con los datos necesarios para publicarse.</li> </ul>			
<b>N.º de Pasos</b>	<b>Acciones</b>	<b>Resultados Esperados</b>	<b>Resultados Obtenidos</b>
1	El autor "Retsuko" selecciona el botón "Publicar" para la obra "Retsuko Tales" en estado "Borrador".	Que se envíe la obra "Retsuko Tales" para su revisión y próxima aprobación.	Se cambia el estado de la obra "Retsuko Tales" del autor Retsuko de "Borrador" a "Esperando aprobación".
2	El lector "Jotaro" selecciona el botón "Biblioteca".	Que se muestre la página de búsqueda con el catálogo de publicaciones sin filtros aplicados.	Se obtienen las últimas 2 publicaciones y sus datos principales:

			<ul style="list-style-type: none"> <li>❖ "Public Obra 1", "3 Likes", "6 visualizaciones", "[Imagen1]", "Ciencia Ficción".</li> <li>❖ "Public Obra 2", "5 Likes", "7 visualizaciones", "[Imagen2]", "Romance".</li> </ul>
3	El lector "Jotaro" selecciona el campo "Buscar" de filtros y escribe "Retsuko Tales".	Que el sistema muestre las obras con el nombre "Retsuko Tales".	<p>Se obtienen las últimas 2 publicaciones que tengan el nombre o alguna etiqueta con la palabra "Retsuko Tales" en orden de última fecha publicada a la más antigua:</p> <ul style="list-style-type: none"> <li>❖ "Retsuko Tales: En el principio del fin", "Retsuko", "[Imagen Portada 1]", "2 Likes", "4 Visualizaciones", "Terror".</li> <li>❖ "Cielo estrellado", "Retsuko", "[Imagen Portada 2]", "6 Likes", "12 Visualizaciones", "Romance".</li> </ul>
4	El lector "Jotaro" selecciona la publicación	Que el sistema muestre la pantalla con la publicación "Retsuko	Se muestra la publicación "Retsuko Tales: En el principio del fin" y sus detalles:

	<p>"Retsuko Tales: En el principio del fin".</p>	<p>Tales: En el principio del fin" y sus detalles.</p>	<p>❖ "Retsuko Tales: En el principio del fin", "Retsuko", "[Imagen Portada 1]","2 Likes","4 Visualizaciones","Terror", "Muy Bueno", "Obra fascinante de miedo e historias", "[Favorito]".</p>
<p><b>Estado de la prueba</b></p>	<p>Exitosa.</p>		
<p><b>Acciones correctivas</b></p>	<p>-----</p>		
<p><b>Pantalla</b></p>	<div data-bbox="475 1126 1305 1906" style="border: 1px solid black; padding: 10px; background-color: #2e3436; color: #eeeeec; font-family: monospace;"> <pre> 0    #[tokio::test] 1    async fn publish() { 2        let c = mocks::container(); 3        let uc = Publish::new(c.event_pub(), c.author_repo(), c.publication_repo()); 4    5        let mut author = mocks::author("#user01", "user"); 6        c.author_repo().save(&amp;mut author).await.unwrap(); 7        let mut publication = mocks::publication( 8            "#publication01", 9            "#user01", 10           "Publication 01", 11           "category-1", 12           vec!["Tag 1", "Tag 2"], 13           "domain.com/cover.jpg", 14           3, 15           false, 16           false, 17           false, 18       ); 19       c.publication_repo().save(&amp;mut publication).await.unwrap(); 20   21       uc.exec( 22           author.base().id().to_string(), 23           publication.base().id().to_string(), 24       ) 25       .await 26       .unwrap(); 27   28       let publication = c 29           .publication_repo() 30           .find_by_id(&amp;publication.base().id()) 31           .await 32           .unwrap(); 33       assert_eq!( 34           publication.status_history().current().to_string(), 35           "waiting-approval" 36       ); 37   } </pre> </div> <p>En esta instancia, la publicación se coloca en estado de "esperando aprobación".</p>		



Luego, un administrador o gestor de contenido aprueba la obra, como ya vimos en un test anterior. Una vez aprobada, aparece en el catálogo y puede visualizarse su contenido.

```

0  #[tokio::test]
1  async fn owner_view_of_draft() {
2      let c = mocks::container();
3      let uc = GetById::new(
4          c.event_pub(),
5          c.author_repo(),
6          c.category_repo(),
7          c.interaction_repo(),
8          c.publication_repo(),
9          c.reader_repo(),
10         c.user_repo(),
11         c.statistics_serv(),
12     );
13
14     let (mut user1, mut author1, mut reader1) = user(1);
15     c.user_repo().save(&mut user1).await.unwrap();
16     c.author_repo().save(&mut author1).await.unwrap();
17     c.reader_repo().save(&mut reader1).await.unwrap();
18
19     let (mut user2, mut author2, mut reader2) = user(2);
20     c.user_repo().save(&mut user2).await.unwrap();
21     c.author_repo().save(&mut author2).await.unwrap();
22     c.reader_repo().save(&mut reader2).await.unwrap();
23
24     let mut category = mocks::category("Category 1");
25     c.category_repo().save(&mut category).await.unwrap();
26
27     let mut publication = mocks::publication(
28         "#publication01",
29         "#user01",
30         "Publication 01",
31         "category-1",
32         vec!["Tag 1", "Tag 2"],
33         "domain.com/cover.jpg",
34         3,
35         false,
36         false,
37         false,
38     );
39     c.publication_repo().save(&mut publication).await.unwrap();
40
41     let res = uc
42         .exec(
43             Some(reader1.base().id().to_string()),
44             publication.base().id().to_string(),
45             Include::default().add_field("author").add_field("category"),
46         )
47         .await
48         .unwrap();

```

Como ejemplo, podemos ver que se comprueba que el dueño de una publicación puede ver su contenido incluso si esta se encuentra en estado "borrador".

```

test application::publication::get_by_id::tests::owner_view_of_draft ... ok
test application::publication::read::tests::valid ... ok

```

```

test application::publication::get_by_id::tests::reader_view_of_published ... ok
test application::publication::reject::tests::reject ... ok

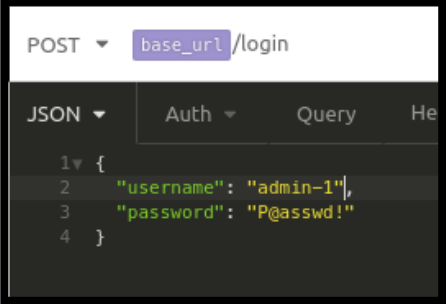
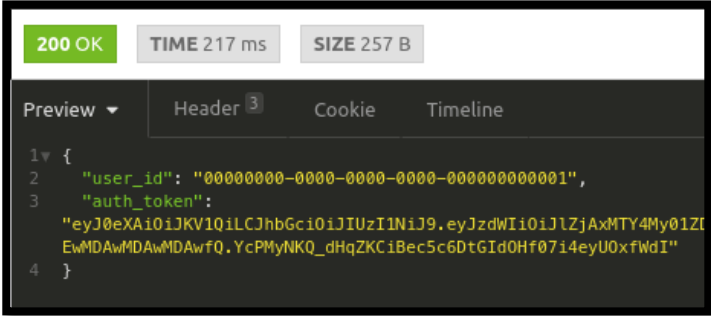
```

Como última verificación, se comprueba la visualización del contenido que tiene un lector de la Publicación en estado "Publicada" (aprobada).

Tabla 1. 15 Caso de Prueba de integración entre los Módulos del Sistema 03.

### 3.3.4. Pruebas de Seguridad por Niveles de Usuario

Caso de Prueba de Seguridad por Niveles de Usuario 01: Ingreso de Usuario Administrador.			
<b>Tipo de caso de prueba:</b> Caso de éxito.			
<b>Caso de uso asociado:</b> Iniciar Sesión.			
<b>Fecha de creación</b>	17/08/2020		
<b>Fecha de ejecución</b>	28/08/2020		
<b>Autor</b>	Administrador.		
<b>Objetivo:</b> Comprobar que el administrador puede visualizar las acciones que le corresponden a su rol: Reportes, Gestionar Publicaciones y Contratos, Gestionar Reglas de Negocio, Gestionar Categorías, Favoritos, Biblioteca, Dashboard, Autores, Sobre Omics.			
<b>Precondiciones:</b>			
<ul style="list-style-type: none"> <li>Administrador instanciado.</li> </ul>			
N.º de Pasos	Acciones	Resultados Esperados	Resultados Obtenidos
1	El usuario "Gabriel Perez" selecciona "Ingresar".	Que se despliegue el formulario para ingresar a la plataforma: <ul style="list-style-type: none"> <li>❖ Usuario o E-mail.</li> <li>❖ Contraseña.</li> <li>❖ Reestablecer contraseña.</li> </ul>	Se muestra la pantalla con los campos: <ul style="list-style-type: none"> <li>❖ Usuario o E-mail.</li> <li>❖ Contraseña.</li> <li>❖ Reestablecer contraseña.</li> </ul>
2	El usuario "Gabriel Perez" ingresa con el usuario = "gabito" y contraseña = "gabi1234".	Que se validen los datos del usuario ingresado.	Se muestra el menú principal de la plataforma con las acciones que puede realizar un administrador:

		<ul style="list-style-type: none"> <li>❖ Reportes.</li> <li>❖ Gestionar Publicaciones y Contratos.</li> <li>❖ Gestionar Reglas de Negocio.</li> <li>❖ Gestionar Categorías.</li> <li>❖ Favoritos.</li> <li>❖ Biblioteca.</li> <li>❖ Deskboard.</li> <li>❖ Autores.</li> <li>❖ Sobre Omics.</li> </ul>
<p><b>Estado de la prueba</b></p>	<p>Exitosa.</p>	
<p><b>Acciones correctivas</b></p>	<p>-----</p>	
<p><b>Pantalla</b></p>	<div style="text-align: center;">  <p>Se loguea un administrador.</p>  <p>Sistema le devuelve un token para realizar todas sus futuras acciones.</p> </div>	

	<pre> 1 { 2   "id": "00000000-0000-0000-0000-000000000001", 3   "username": "admin-1", 4   "email": "admin-1@omics.com", 5   "name": "Admin", 6   "lastname": "Omics", 7   "birthdate": "1994-08-01T15:30:00+00:00", 8   "gender": "other", 9   "biography": "I am a administrator.", 10  "profile_image": "https://via.placeholder.com/200.jpg", 11  "validated": true, 12  "role_id": null, 13  "role": { 14    "id": "admin", 15    "name": "Administrador" 16  }, 17  "created_at": "2020-10-26T00:37:45.221825+00:00", 18  "updated_at": null 19 }</pre>	
<p>Utiliza el token para obtener sus datos (tiene el rol de administrador asignado).</p>		

Tabla 1. 16 Caso de Prueba de Seguridad por Niveles de Usuario 01.

Caso de Prueba de Seguridad por Niveles de Usuario 02:			
Ingresar a una Ruta Sin Permisos.			
<b>Tipo de caso de prueba:</b> Caso de fallo.			
<b>Caso de uso asociado:</b> Gestionar Colecciones.			
<b>Fecha de creación</b>	16/08/2020		
<b>Fecha de ejecución</b>	08/09/2020		
<b>Autor</b>	<ul style="list-style-type: none"> <li>Autor.</li> <li>Lector.</li> </ul>		
<b>Objetivo:</b> Lograr ingresar a una ruta de la plataforma por la cual no se tiene permisos, y recibir una indicación del hecho.			
<b>Precondiciones:</b>			
<ul style="list-style-type: none"> <li>Tener la ruta de acceso para Gestionar Colecciones.</li> <li>Usuario no tiene permisos de Administrador.</li> </ul>			
N.º de Pasos	Acciones	Resultados Esperados	Resultados Obtenidos

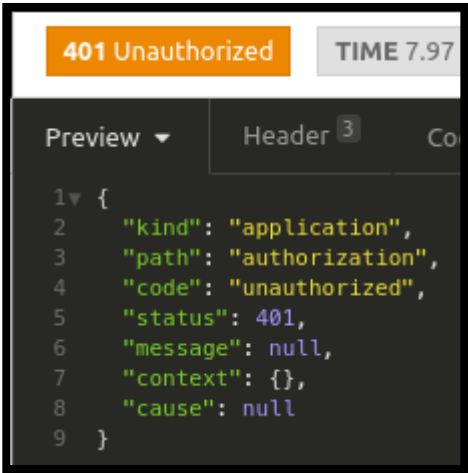

1	El usuario "Nolan" ingresa a la ruta de "Gestionar".	Que el sistema valide los permisos del usuario.	Se muestra en la pantalla el siguiente mensaje: "Error- No es dueño del perfil que desea acceder".
2	El usuario "Nolan" selecciona "Aceptar".	Que el sistema compruebe los permisos que tiene habilitado el usuario.	El usuario es redirigido a la ruta de inicio de la plataforma.
<b>Estado de la prueba</b>	Fallida.		
<b>Acciones correctivas</b>	En caso de fallo el usuario debe solicitar los permisos de administrador al encargado del sistema.		
<b>Pantalla</b>			

Tabla 1. 17 Caso de Prueba de Seguridad por Niveles de Usuario 02.

<b>Caso de Prueba de Seguridad por Niveles de Usuario 03:</b> Autor puede Visualizar Reportes y Detalles de sus Publicaciones.	
<b>Tipo de caso de prueba:</b> Caso de éxito.	
<b>Caso de uso asociado:</b> Publicar Obra.	
<b>Fecha de creación</b>	06/09/2020
<b>Fecha de ejecución</b>	10/10/2020

<b>Autor</b>	Autor.		
<b>Objetivo:</b> Comprobar que el autor al subir una obra por primera vez puede visualizar los reportes y detalles de sus publicaciones.			
<b>Precondiciones:</b> <ul style="list-style-type: none"> <li>• Autor instanciado.</li> <li>• Autor no tiene Publicaciones existentes.</li> </ul>			
<b>N.º de Pasos</b>	<b>Acciones</b>	<b>Resultados Esperados</b>	<b>Resultados Obtenidos</b>
1	El autor "Lee Stan" selecciona: "Publicar Obra".	Que se permita subir su primera obra.	Se muestra un mensaje que dice: "¿Quiere subir su primera obra?".
2	El autor "Lee Stan" selecciona: "Aceptar".	Que se muestre el formulario necesario para subir una obra con la lista de Categorías existentes: Categorías: ❖ Drama. ❖ Romance. ❖ Comedia. ❖ Terror. ❖ Superhéroes. ❖ Fantasía. ❖ Vida diaria. ❖ Sci-Fi. ❖ Acción.	Se muestra el formulario con la lista de Categorías existentes: Categorías: ❖ Drama. ❖ Romance. ❖ Comedia. ❖ Terror. ❖ Superhéroes. ❖ Fantasía. ❖ Vida diaria. ❖ Sci-Fi. ❖ Acción.  Se visualiza la interfaz de publicar una obra con los campos: ❖ Título. ❖ Nombre de colección. ❖ Categoría. ❖ Descripción. ❖ Etiquetas.

			<ul style="list-style-type: none"> <li>❖ Portada.</li> <li>❖ Páginas.</li> </ul>
3	<p>El autor "Lee Stan" coloca los siguientes datos a la Publicación:</p> <ul style="list-style-type: none"> <li>❖ "Dead Note".</li> <li>❖ "Serie Dead Note".</li> <li>❖ "Drama".</li> <li>❖ "Historia de un muchacho vengativo".</li> <li>❖ "Violencia".</li> <li>❖ "deathNotethumbail.jpg".</li> <li>❖ "10 páginas".</li> </ul>	<p>Que el sistema valide que los datos ingresados sean correctos para guardar la publicación = "Dead Note".</p>	<p>Se muestra resaltado con verde los campos correctos ingresados y se habilita el botón: "Publicar obra".</p>
4	<p>El autor "Lee Stan" selecciona: "Publicar Obra".</p>	<p>Que el sistema guarde la obra = "Dead Note", con estado = "Borrador".</p>	<p>Se muestra un mensaje: "Pendiente- Su publicación aún se encuentra en revisión".</p>
5	<p>El autor "Lee Stan" selecciona el botón "Guardar como Borrador".</p>	<p>Que el sistema guarde la obra = "Dead Note", con estado = "Esperando aprobación".</p>	<p>No aparezca la obra = "Dead Note", en búsquedas ni se pueda acceder a su detalle en la plataforma.</p>
6	<p>El autor "Lee Stan" selecciona del menú el botón "Dashboard " para ver el estado de la publicación "Dead Note".</p>	<p>Que el sistema apruebe la obra = "Dead Note", para que sea publicada en la plataforma.</p>	<p>Se cambia el estado de la obra "Dead Note", de "Esperando aprobación" a "Publicada".</p> <p>Se publica la obra "Dead Note" en la plataforma.</p>

			Se le envía una notificación al autor "Lee Stan" de la aprobación de la obra "Dead Note".
7	El autor "Lee Stan" selecciona la publicación "Dead Note".	Que se pueda tener acceso a los detalles de la obra = "Dead Note".	Se visualiza la publicación del autor "Lee Stan" con los detalles correspondientes de "Dead Note": <ul style="list-style-type: none"> <li>❖ "Dead Note".</li> <li>❖ "Serie Dead Note".</li> <li>❖ "Drama".</li> <li>❖ "Historia de un muchacho vengativo".</li> <li>❖ "Violencia".</li> <li>❖ "deathNotethumba il.jpg".</li> <li>❖ "10 páginas".</li> </ul>
<b>Estado de la prueba</b>	Exitosa.		
<b>Acciones correctivas</b>	-----		
<b>Pantalla</b>	 <p>Un autor puede consultar el contenido y detalle de sus Publicaciones, lo que incluye estadísticas y reportes.</p>		



	<pre> 1▼ { 2   "offset": 0, 3   "limit": 100, 4   "total": 173, 5   "matching_criteria": 3, 6   "count": 3, 7▼  "items": [ 8▼   { 9     "id": "bf0797cb-1ae0-4321-b020-dffb2864df5", 10    "author_id": "afddd2ef-0f7f-4d45-9623-66ec0bd9702e", 11    "author": null, 12    "name": "Star Wars: Bounty Hunters (2020) #6", 13    "synopsis": "Harum repudiandae molestias blanditiis laborum. Ut aut suscipit nihil rerum asperiores facere ut modi. Velit odit facilis et sit qui. Quo ex esse perspiciatis numquam porro earum earum voluptatem. Nisi ea velit.", 14    "category_id": "science-fiction", 15    "category": null, 16▼   "tags": [ 17     "Hola", 18     "Chau" 19   ], 20   "cover": "http://i.annihil.us/u/prod/marvel/i/mg/d/20/5f3ee0df8c488.jpg", 21▼  "statistics": { 22    "views": 151, 23    "unique_views": 0, 24    "readings": 119, 25    "likes": 43, 26    "reviews": 24, 27    "stars": 2.75 28  }, </pre>
--	--

Tabla 1. 18 Caso de Prueba de Seguridad por Niveles de Usuario 03.

### 3.3.5. Pruebas de Carga

<b>Caso de Prueba de Carga 01:</b> Peticiones por segundo de la API REST al endpoints de usuarios.	
<b>Tipo de caso de prueba:</b> Caso de éxito.	
<b>Caso de uso asociado:</b> Ver Perfil Usuario.	
<b>Fecha de creación</b>	10/08/2020
<b>Fecha de ejecución</b>	23/09/2020
<b>Autor</b>	Administrador.
<b>Objetivo:</b> Lograr superar la mínima cantidad aceptable (7000 req/s) de peticiones por segundo que provee la API REST y el back-end al obtener los datos de los usuarios.	
<b>Precondiciones:</b> <ul style="list-style-type: none"> <li>Back-end en funcionamiento.</li> <li>Motor de base de datos en funcionamiento.</li> <li>Back-end conectado al motor de base de datos real que se utiliza en producción.</li> <li>API definida y funcionando.</li> <li>ab (Programa de Apache para realizar pruebas de carga sobre una API).</li> <li>500 cuentas de usuarios creadas.</li> </ul>	

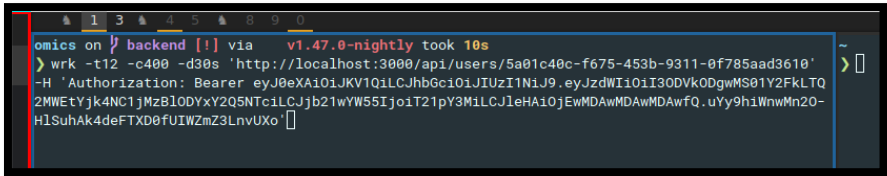
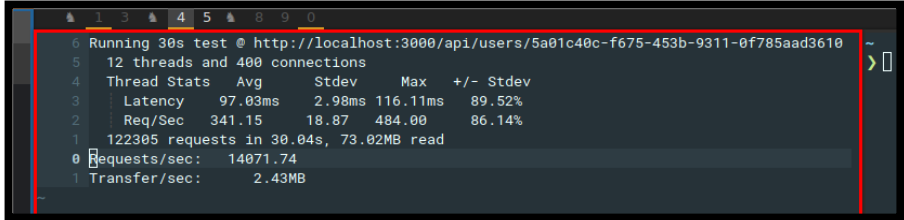
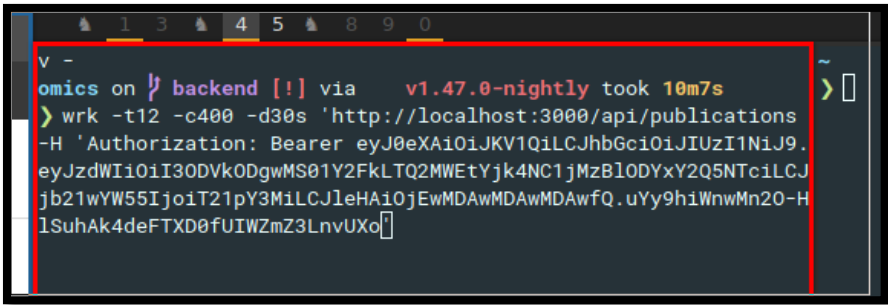
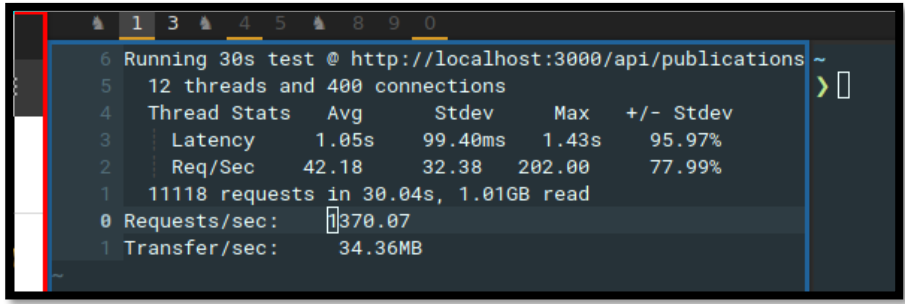
N.º de Pasos	Acciones	Resultados Esperados	Resultados Obtenidos
1	El administrador "Pepo" ejecuta <i>ab</i> sobre la API para obtener un usuario por ID.	Cantidad de peticiones por segundo que soporta el endpoint (7000 req/s).	Resultados mínimos obtenidos = 14071 req/s.
<b>Estado de la prueba</b>	Exitosa.		
<b>Acciones correctivas</b>	-----		
<b>Pantalla</b>	 <p>Se corre wrk con 12 threads y 400 conexiones simultáneas durante 30 segundos.</p> 		

Tabla 1. 19 Caso de Prueba de Carga 01.

<b>Caso de Prueba de Carga 02:</b>	
Peticiones por segundo de la API REST al endpoints de catálogo.	
<b>Tipo de caso de prueba:</b> Caso de éxito.	
<b>Caso de uso asociado:</b> Buscar y Filtrar contenido.	
<b>Fecha de creación</b>	16/09/2020
<b>Fecha de ejecución</b>	28/09/2020

<b>Autor</b>	Administrador.		
<b>Objetivo:</b> Lograr superar la mínima cantidad aceptable (1000 req/s) de peticiones por segundo que provee la API REST y el back-end al obtener los datos de las obras contenido en catálogos.			
<b>Precondiciones:</b> <ul style="list-style-type: none"> <li>• Back-end en funcionamiento.</li> <li>• Motor de base de datos en funcionamiento.</li> <li>• Back-end conectado al motor de base de datos real que se utiliza en producción.</li> <li>• API definida, y funcionando.</li> <li>• ab (Programa de Apache para realizar pruebas de carga sobre una API).</li> <li>• 200 obras publicadas en la plataforma.</li> </ul>			
N.º de Pasos	Acciones	Resultados Esperados	Resultados Obtenidos
1	El administrador "Pepo" ejecuta <i>ab</i> para obtener el catálogo ("inicio" de la plataforma).	Cantidad de peticiones por segundo que soporta el endpoint (1000 req/s).	Resultados mínimos obtenidos = 1370 req/s.
<b>Estado de la prueba</b>	Exitosa.		
<b>Acciones correctivas</b>	-----		
<b>Pantalla</b>	 <pre> v - omics on backend [!] via v1.47.0-nightly took 10m7s &gt; wrk -t12 -c400 -d30s 'http://localhost:3000/api/publications -H 'Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9. eyJzdWIiOiI3ODVhODgwMS01Y2FkLTQ2MWEtYjk4NC1jMzBlODYxY2Q5NTciLCJ jb21wYW55IjoiaT21pY3MiLCJleHAiOjEwMDAwMDAwMDAwfQ.uYy9hiWnwMn20-H lSuhAk4deFTXD0fUIWZmZ3LnvUXo </pre>		



```

6 Running 30s test @ http://localhost:3000/api/publications ~
5 12 threads and 400 connections >
4 Thread Stats Avg Stdev Max +/- Stdev
3 Latency 1.05s 99.40ms 1.43s 95.97%
2 Req/Sec 42.18 32.38 202.00 77.99%
1 11118 requests in 30.04s, 1.01GB read
0 Requests/sec: 370.07
1 Transfer/sec: 34.36MB
    
```

Tabla 1. 20 Caso de Prueba de Carga 02.

<b>Caso de Prueba de Carga 03:</b>			
Peticiones por segundo de la API REST al endpoints de catálogo.			
<b>Tipo de caso de prueba:</b> Caso de éxito.			
<b>Caso de uso asociado:</b> Publicar obra.			
<b>Fecha de creación</b>	19/09/2020		
<b>Fecha de ejecución</b>	30/09/2020		
<b>Autor</b>	Administrador.		
<b>Objetivo:</b> Lograr superar la mínima cantidad aceptable (20 req/s) de peticiones por segundo que provee la API REST y el back-end al subir imágenes al publicar una nueva obra.			
<b>Precondiciones:</b>			
<ul style="list-style-type: none"> <li>• Back-end en funcionamiento.</li> <li>• Motor de base de datos en funcionamiento.</li> <li>• Back-end conectado al motor de base de datos real que se utiliza en producción.</li> <li>• API definida y funcionando.</li> <li>• ab (Programa de Apache para realizar pruebas de carga sobre una API).</li> </ul>			
N.º de Pasos	Acciones	Resultados Esperados	Resultados Obtenidos
1	El administrador "Pepo" ejecuta <i>ab</i> para publicar una	Cantidad de peticiones por segundo que	Resultados mínimos aceptables = 22.01 req/s.

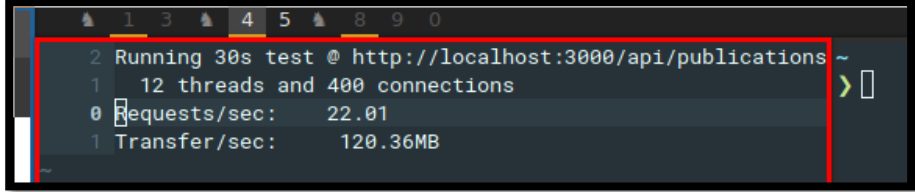
	nueva obra con 50 imágenes.	soporta el endpoint (10 req/s).	
<b>Estado de la prueba</b>	Exitosa.		
<b>Acciones correctivas</b>	-----		
<b>Pantalla</b>			

Tabla 1. 21 Caso de Prueba de Carga 03.

<b>Caso de Prueba de Carga 04:</b> Peticiones por segundo de la API REST al endpoints de catálogo.	
<b>Tipo de caso de prueba:</b> Caso de fallo.	
<b>Caso de uso asociado:</b> Publicar obra.	
<b>Fecha de creación</b>	19/09/2020
<b>Fecha de ejecución</b>	05/10/2020
<b>Autor</b>	Administrador.
<b>Objetivo:</b> que no se pueda lograr superar la mínima cantidad aceptable (10 req/s) de peticiones por segundo que provee la API REST y el back-end al subir imágenes al publicar una nueva obra.	
<b>Precondiciones:</b> <ul style="list-style-type: none"> <li>• Back-end en funcionamiento.</li> <li>• Motor de base de datos en funcionamiento.</li> <li>• Back-end conectado al motor de base de datos real que se utiliza en producción.</li> <li>• API definida y funcionando.</li> <li>• ab (Programa de Apache para realizar pruebas de carga sobre una API).</li> </ul>	

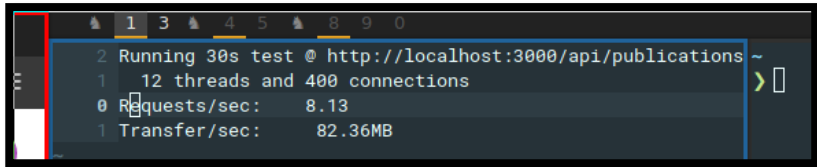
N.º de Pasos	Acciones	Resultados Esperados	Resultados Obtenidos
1	El administrador "Pepo" ejecuta <i>ab</i> para publicar una nueva obra con 50 imágenes.	Cantidad de peticiones por segundo que soporta el endpoint (10 req/s).	Resultados obtenidos = 8.13 req/s.
<b>Estado de la prueba</b>	Fallida.		
<b>Acciones correctivas</b>	<p>Optimizar endpoint haciendo uso de caché.</p> <p>Subir imágenes directo desde el front-end al servicio de almacenamiento de imágenes.</p> <p>Generar tablas de visualización, aplicar CQRS.</p>		
<b>Pantalla</b>			

Tabla 1. 22 Caso de Prueba de Carga 04.

## 4. Manual de Usuario del Sistema Completo

### Manual de Usuario

Expone los procesos que el usuario puede realizar con el sistema implantado. Para lograr esto, es necesario que se detallen todas y cada una de las características que tienen los programas y la forma de acceder e introducir información. Permite a los usuarios conocer el detalle de qué actividades ellos deberán desarrollar para la consecución de los objetivos del sistema. Reúne la información, normas y documentación necesaria para que el usuario conozca y utilice adecuadamente la aplicación desarrollada.

### Objetivos:

- ❖ Que el usuario conozca cómo preparar los datos de entrada.
- ❖ Que el usuario aprenda a obtener los resultados y los datos de salida.
- ❖ Servir como manual de aprendizaje.
- ❖ Servir como manual de referencia.
- ❖ Definir las funciones que debe realizar el usuario.
- ❖ Informar al usuario de la respuesta a cada mensaje de error.

### Pasos a seguir para definir como desarrollar el manual de usuario.

1. Identificar los usuarios del sistema: personal que se relacionará con el sistema.
2. Definir los diferentes tipos de usuarios: Se presentan los diferentes tipos de usuarios que usarían el sistema. En el caso de *OmicS*: Administrador del Sistema, Gestor de Contenido y Usuarios (Autor-Lector).
3. Definir los módulos en que cada usuario participará: Se describen los módulos o procesos que se ejecutarán por cada usuario en forma narrativa breve y clara.

### Importancia Del Manual De Usuario

El Manual de Usuario facilita el conocimiento de:

- ❖ Los documentos a los que se puede dar entrada por computadora.
- ❖ Los formatos de los documentos.
- ❖ Las operaciones que utiliza de entrada y salida de los datos.
- ❖ El orden del tratamiento de la computadora con los datos introducidos.
- ❖ El momento en que se debe solicitar una operación deseada.
- ❖ Los resultados de las operaciones realizadas a partir de los datos introducidos.

Al elaborar el Manual de Usuario, hay que tener en cuenta a quién va dirigido. Por consiguiente, debe redactarse de forma clara y sencilla para que lo entienda cualquier tipo de usuario (**VER ANEXO N° 5 MANUAL DE USUARIOS**).

## 5. Planificación de Implementación del Sistema

Se describe a continuación la Planificación de la Implementación de la Plataforma Omics.

### 5.1. Objetivos de la Implementación de la Plataforma

La fase de implementación es la más costosa porque se ven involucradas muchas personas, herramientas y recursos, y la que consume más tiempo porque culmina todo el trabajo realizado en las etapas anteriores del ciclo de vida del sistema.

Es por ello que se ha desarrollado un plan de implementación en donde se tiene como objetivo general lograr una productiva y eficaz implementación.

#### Los objetivos específicos son:

- ❖ Definir actividades de instalación, y personalización.
- ❖ Realizar la capacitación a los usuarios.
- ❖ Brindar documentación, y manuales para comprender mejor el funcionamiento de la Plataforma.

En la **Implementación** se requiere mayor trabajo manual. Las actividades que se tienen en cuenta para efectuar la implementación son las detalladas a continuación:

- Planificación de las tareas: Establecer las tareas con previsión acerca de las actividades que se deben realizar y estimar los tiempos de cada una.
- Asignar los responsables: Deben asignarse de forma precisa las tareas a realizar, para que cada responsable pueda identificar su relación con las etapas de puesta en funcionamiento del sistema.
- Verificación y control de infraestructura: Actividad centrada en evaluar la infraestructura general (servidores, base de datos, servicios externos, seguridad).
- Instalación de herramientas que soportan la arquitectura del sistema: Instalación y configuración de todas las dependencias del proyecto, siendo estas librerías, frameworks, drivers, instancias de servicios (como bases de datos y bus para comunicación asíncrona).
- Carga del código del sistema: El código completo del sistema se encuentra en repositorios, por lo que es necesario extraer de las ramas correspondientes los archivos para poner el sistema en funcionamiento, para cargarlos en el servidor principal donde operará.
- Ejecución de migraciones: Se ejecutan los scripts iniciales para crear las bases de datos correspondientes y también las migraciones que realizarán modificaciones sobre el esquema de datos. Las migraciones permitirán,



en un futuro, alterar este esquema de datos, ampliando la estructura de la base de datos, según lo plantean los requerimientos.

- Ejecución de scripts para poblar las bases de datos: Ya establecido el esquema de los datos en las bases de datos relacionales, se ejecutan scripts que poblarán las tablas con los datos iniciales: inserción de datos globales como categorías, estados, configuraciones y creación de usuarios con niveles de administrador.
- Configuración de puntos de acceso: Se configuran los puntos de acceso de cada servicio, sea interno o del mismo sistema. Técnicamente, se establecen las redes virtuales mediante las cuales se comunicará el sistema con las instancias de servicio como las bases de datos y buses de comunicación, y también se gestionan los puertos en los que operará cada servicio.
- Capacitación: Se capacita el personal encargado de monitorear el funcionamiento del sistema, los servidores y los servicios de los que depende. A su vez, se capacita al personal interno que hará uso de funcionalidades de administrador para gestionar el contenido.
- Monitoreo del sistema en funcionamiento: De acuerdo a lo planificado, se monitorea el sistema operando para asegurar su disponibilidad las 24 horas del día.
- Difusión en redes sociales: En la implementación inicial y futuros incrementos que representan grandes cambios, los cuales pueden aportar nuevos o mejores beneficios a los usuarios finales, el esfuerzo de Marketing es fundamental para dar a conocer estas características.

## 5.2. Recursos Involucrados

Se definen tanto recursos físicos, como humanos.

### ❖ **Requerimientos de PC:**

- Procesador: i7 7700 (Intel).
- Memoria: 16 GB de RAM.
- Almacenamiento: 2 TB de disco duro, 256 GB SSD M.2.
- Placa de vídeo: Asus GTX 1700, 8 GB de memoria de video.
- Sistema operativo: Arch Linux.

### ❖ **Perfiles del Equipo de Implementación:**

- Product Manager (1).
- Arquitecto de Software (1).
- Scrum Master (1).

- Desarrollador Back-end (1).
- Desarrollador Front-end (1).
- Tester Jr. (1).

❖ **Roles Involucrados luego de la Implementación:**

- Administrador del Sistema.
- Content Manager (Gestor de Contenido).
- Usuario (Lector-Autor).

## 5.3. Actividades

### 5.3.1. Especificaciones de Servidores

- ❖ Se utilizan dos servidores para correr el back-end, uno principal, y otro como respaldo. En éste corren los sistemas de gestión de bases de datos utilizados y servicios de procesamiento. Se cuenta, también, con un servidor enfocado en el almacenamiento de archivos, del estilo servidor CDN (Content Delivery Network).

Dado que la aplicación será alojada en un servidor web mediante algún paquete de hosting, no es necesario contar con servidores propios para correr el servidor ni la base de datos, todo esto será administrado por el Proveedor de servicio de alojamiento.

En base al análisis realizado en las pruebas de carga, se requieren Servidores que cumplan al menos con las siguientes características:

- Memoria RAM: Al menos 16GB DDR4.
  - Procesador: 4 núcleos o más. Frecuencia de reloj del procesador mayor a 2,3 GHz
  - Almacenamiento: 4 TB o superior.
- ❖ Servidor S3 donde se almacenan las imágenes, con las siguientes características:
    - Almacenamiento: 5 TB o superior.

### 5.3.2. Equipos de Conectividad

Los equipos a utilizar inicialmente para conectar los puestos de trabajos y para contar con acceso a internet son:

- ❖ Provistos por un Coworking alquilado de Campus Olegario. Ellos son los encargados de proveer toda la infraestructura y el mantenimiento de los equipos.

### 5.3.3. Comunicaciones

Para que el equipo de *Omic*s esté conectado se utilizan 5 celulares corporativos, comprados a la compañía "CLARO". La comunicación interna se realiza utilizando: Slack, Telegram y Discord (canal Omics). La difusión de las noticias de la plataforma Omics, se realizan por medio del Facebook de Omics <https://www.facebook.com/OmicReader>.



Figura 1. 117 Página de Facebook de Omics.

### 5.3.4. Puestos de Trabajo

En la plataforma *Omic*s el Administrador de Sistemas es responsable de ejecutar, mantener, operar y asegurar el correcto funcionamiento del sistema.

#### Algunas funciones son:

- ❖ Realizar copias de respaldo.
- ❖ Montar repositorio de las herramientas utilizadas en el proyecto.
- ❖ Administrar cuentas de usuarios (instalación y mantenimiento).
- ❖ Monitorear el rendimiento del sistema.
- ❖ Instalar el software.
- ❖ Crear la política de copias de seguridad y recuperación.

### 5.3.5. Infraestructura

La infraestructura necesaria para que la plataforma *Omic*s se ponga en funcionamiento es:

- ❖ Acceso a internet.
- ❖ Sistema de gestión de Base de Datos: PostgreSQL.

- ❖ Back end: Rust.
- ❖ Front end: JavaScript como lenguaje de programación, TypeScript, y Angular como Framework.
- ❖ GitHub: como repositorio para el versionado del código.
  
- **Para la administración y mantenimiento de la plataforma:**
  - Equipos para cada puesto, pertenecientes a los Administradores de Omics.
  - Conexión con la red y acceso a internet para la conexión al servidor, provisto por el coworking.
  - Sistema operativo: Windows y Linux.
  - Insomnia para hacer testing a la API.
  - Editor de código fuente: Visual Studio Code.
  
- **Para los Administradores de la plataforma:**
  - Puestos de trabajo en el coworking.
  - Acceso a internet en el coworking.

### 5.3.6. Carga Inicial de Datos

En la carga de datos se identifican tres partes diferentes:

- ❖ **Carga inicial:**
  - Roles: usuario, gestor de contenido y admin.
  - Categorías: Drama, Romance, Comedia, Terror, Superhéroes, Fantasía, Vida diaria, Sci-Fi, Acción.
  - Usuario admin por defecto: admin-1.
  - Usuario gestor de contenido por defecto: content-manager-1.
  - Estados de entidades (Publicación, Suscripción y Contrato).
  - Configuración general de la plataforma (Roles, etc).
  
- ❖ **Conversión de datos:**

Este ítem no aplica ya que la plataforma se implementa por primera vez y no hay datos que cambien de formato.

- ❖ **Depuración:**

Para realizar la depuración se debe asegurar:

- De hacer un backup de los datos de desarrollo.
- Que, si faltan datos, se los agregue.
- Que, si hay datos incorrectos, se los corrija.
- Que no haya datos duplicados.

- Que los datos sean válidos en forma lógica (ejemplo: una dirección que no existe no se debería permitir).
- Integridad de las referencias en las bases de datos.

### 5.3.7. Migración

Las migraciones se realizan con scripts hechos por el equipo de desarrollo de Omics que se encargan de migrar las tablas de la base de datos relacional (PostgreSQL). Que se almacenan en una tabla especial las migraciones ya ejecutadas y ejecuta las que faltan. Cada script de migración no puede eliminarse, sino sólo se pueden agregar a la carpeta donde se encuentran las migraciones: omics/scripts/migrations.

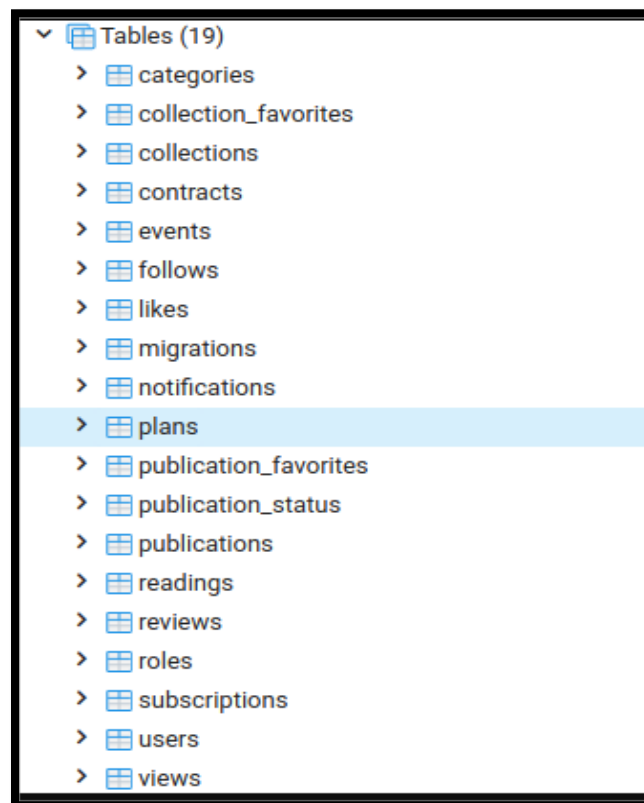


Figura 1. 118 Interfaz gráfica de PGAdmin de las tablas que componen la Base de Datos de Omics.

### 5.3.8. Métodos de Replicación

El sistema de gestión de bases de datos elegido "PostgreSQL" posee un método de replicación.

La replicación es la transmisión de información derivada de las operaciones DML de una Base de Datos a otra, es decir, se transmite a otra Base de Datos las instrucciones

INSERT, UPDATE y DELETE que se realicen en una Base de datos de forma que ambas BBDD tengan la misma información, así pues, se obtiene una redundancia de datos.

### **Replicación en PostgreSQL:**

Uno de los métodos más usados para conseguir una alta disponibilidad en nuestro servidor Postgres es implementar la Replicación. Con ello nos aseguramos de que nuestra plataforma Omics, esté activa el 99,9% del año.

PostgreSQL dispone de varias herramientas y métodos para implementar:

#### ❖ **Tipos:**

- Basada en triggers.
- Basada en ficheros WAL (Write Ahead Log).

#### ❖ **Técnicas:**

- Archivado continuo (WAL Shipping).
- Streaming Replication.

#### ❖ **Forma de transmisión:**

- Asíncrona.
- Síncrona.

### **Tipos de replicación:**

Un trigger (Disparador), es una acción que se ejecuta con cada operación DML. Utiliza una cola para almacenar los cambios y enviarla a la Base de Datos remota de forma asíncrona. Su ventaja, se puede seleccionar un subconjunto de una o varias tablas de las Bases de Datos. Un fichero WAL es un registro con los INSERT, UPDATE y DELETES que se han realizado en una Base de Datos. Los ficheros WAL se generan para garantizar la durabilidad de la Base de Datos.

### **Técnicas de implementación:**

PostgreSQL puede enviar la información de los registros WAL mediante un archivado continuo (WAL Shipping) entre el servidor maestro y el servidor esclavo. También se puede configurar para que se envíen los ficheros WAL con las operaciones DML de una Base de Datos a otra mediante la red (Streaming Replication). El modo de envío de estos ficheros puede ser de forma asíncrona o síncrona.

### 5.3.9. Backup

En este apartado se contempla el realizado de backup, ya sea tanto automático como manual, y, además, la ejecución de migraciones para actualizar los esquemas del motor de base de datos relacional utilizado: PostgreSQL.

#### Para la administración y mantenimiento de la plataforma:

- ❖ Equipos para cada puesto, pertenecientes a los Administradores de *Omics*.
- ❖ Conexión con la red y acceso a internet para la conexión al servidor, provisto por el coworking.
- ❖ Sistema operativo: Windows y Linux.
- ❖ Insomnia para hacer testing a la API.
- ❖ Editor de código fuente: Visual Studio Code.

#### Para los Administradores de la plataforma:

- ❖ Puestos de trabajo en el coworking.
- ❖ Acceso a internet en el coworking.

#### 5.3.9.1. Requisitos

- ❖ Tener el motor de base de datos PostgreSQL instalado.
- ❖ Tener Docker instalado.
- ❖ Preferentemente, correr en entorno GNU/Linux.
- ❖ Tener **make** instalado.

#### 5.3.9.2. Carga inicial

En la carga de datos se identifican tres partes diferentes:

##### Carga inicial:

- ❖ Roles: Usuario, Content Manager (Gestor de Contenido) y Admin.
- ❖ Categorías: Drama, Romance, Comedia, Terror, Superhéroes, Fantasía, Vida diaria, Sci-Fi, Acción.
- ❖ Usuario Admin por defecto: admin-1.
- ❖ Usuario Content Manager por defecto: content-manager-1.
- ❖ Estados de entidades: Publicación, Suscripción y Contrato.
- ❖ Configuración general de la plataforma: Roles, etc.

##### Conversión de datos:

Este ítem no aplica inicialmente ya que la plataforma se implementa por primera vez, y no hay datos que cambien de formato.

En caso de que se requieran conversiones entre tipos de datos, las migraciones se encargarán de llevarlos a cabo. [VER SECCIÓN 5.3.9.3.4. MIGRACIONES.](#)

**Depuración:**

Para realizar la depuración se debe asegurar:

- ❖ Se debe asegurar de hacer un backup de los datos de desarrollo.
- ❖ Que, si faltan datos, se los agregue.
- ❖ Que, si hay datos incorrectos, se los corrija.
- ❖ Que no haya datos duplicados.
- ❖ Que los datos sean válidos en forma lógica (ejemplo: una dirección que no existe no se debería permitir).
- ❖ Integridad de las referencias en las bases de datos.

**5.3.9.2.1. Creación de la base de datos**

En caso de que la base de datos aún no haya sido creada, debe crearse manualmente. Para esto se accede al panel de administración pgAdmin4, utilizado para gestionar la base de datos de forma gráfica y remota.

Las imágenes a continuación están a modo ilustrativo. La URL de acceso y el puerto pueden variar acorde al entorno en el que se encuentre corriendo el sistema. En caso de ser así, consultar con un administrador.

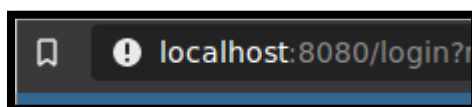


Figura 1. 119 Ingreso a localhost:8080.

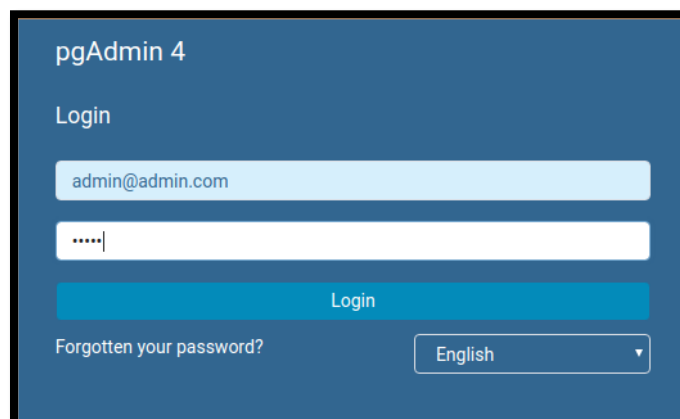


Figura 1. 120 Autenticación del administrador. Estos datos pueden variar por razones de seguridad.



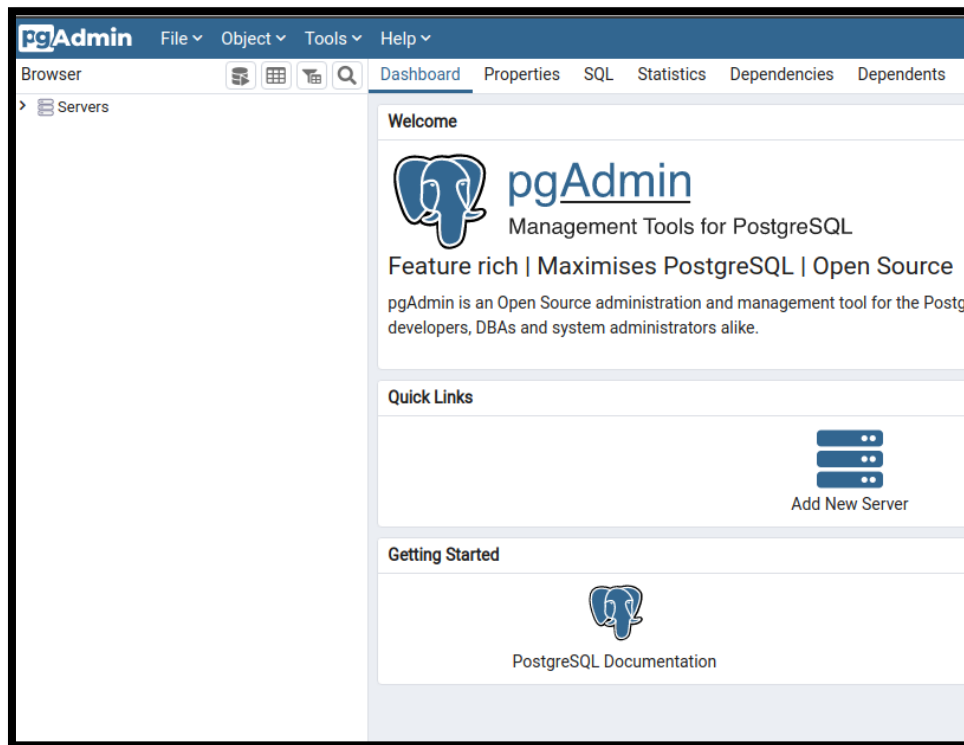


Figura 1. 121 Tablero de administración de PostgreSQL mediante pgAdmin.

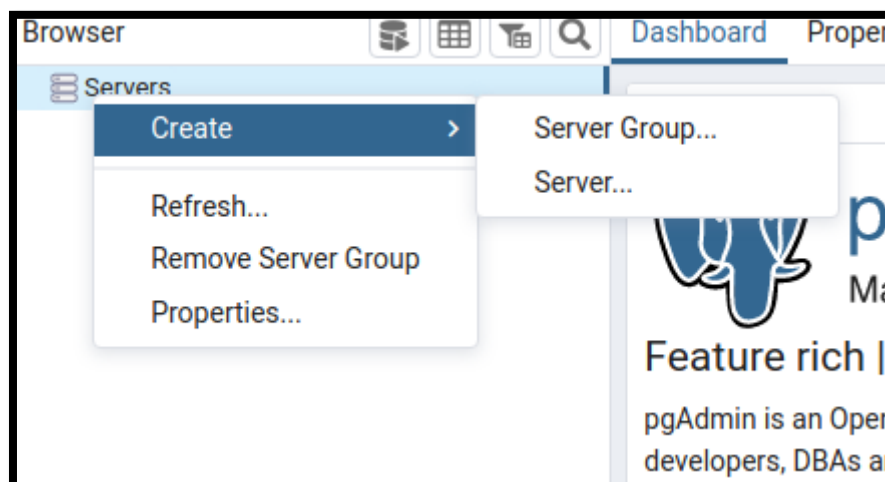


Figura 1. 122 Creación de la conexión con el motor de base de datos.

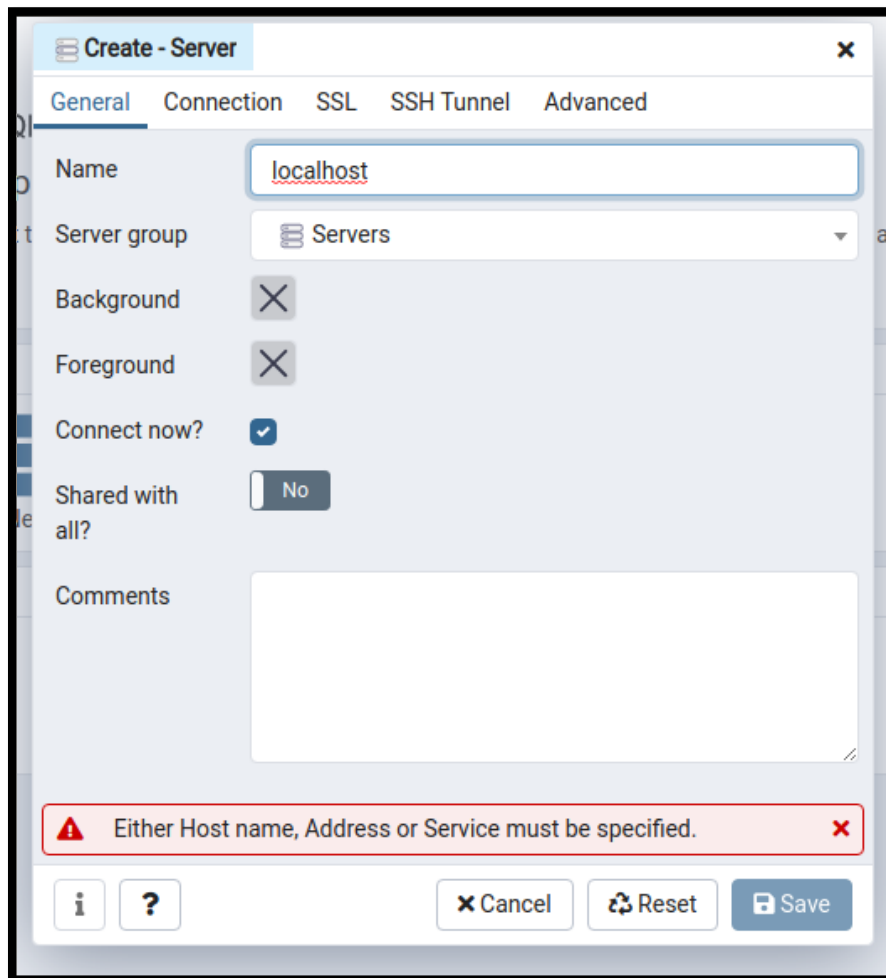


Figura 1. 123 Ingreso de dirección IP donde se encuentra el SGBD corriendo. Puerto por defecto.

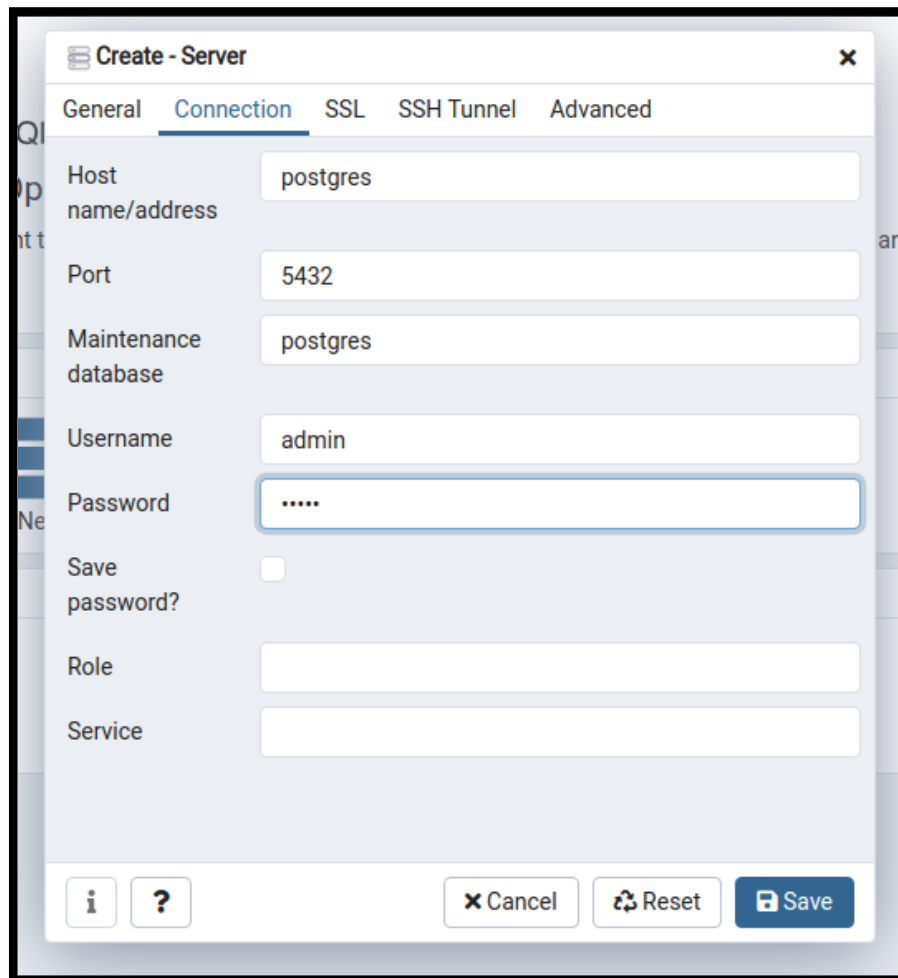


Figura 1. 124 Ingreso de datos de credenciales para acceso a base de datos.

Es necesario considerar que se utiliza la red creada por Docker para lograr la interacción entre los diversos servicios utilizados. Es por esto que utilizamos "postgres" como *hostname*.

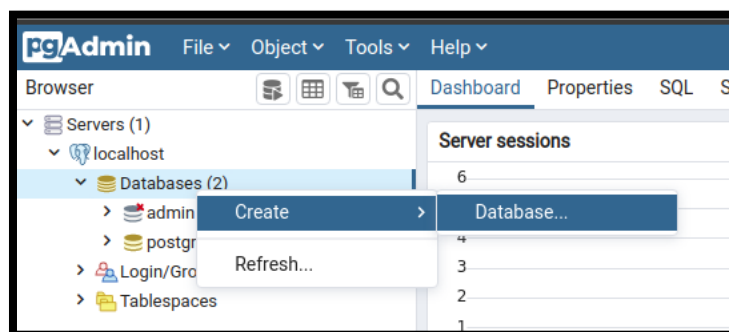


Figura 1. 125 Creamos la base de datos.

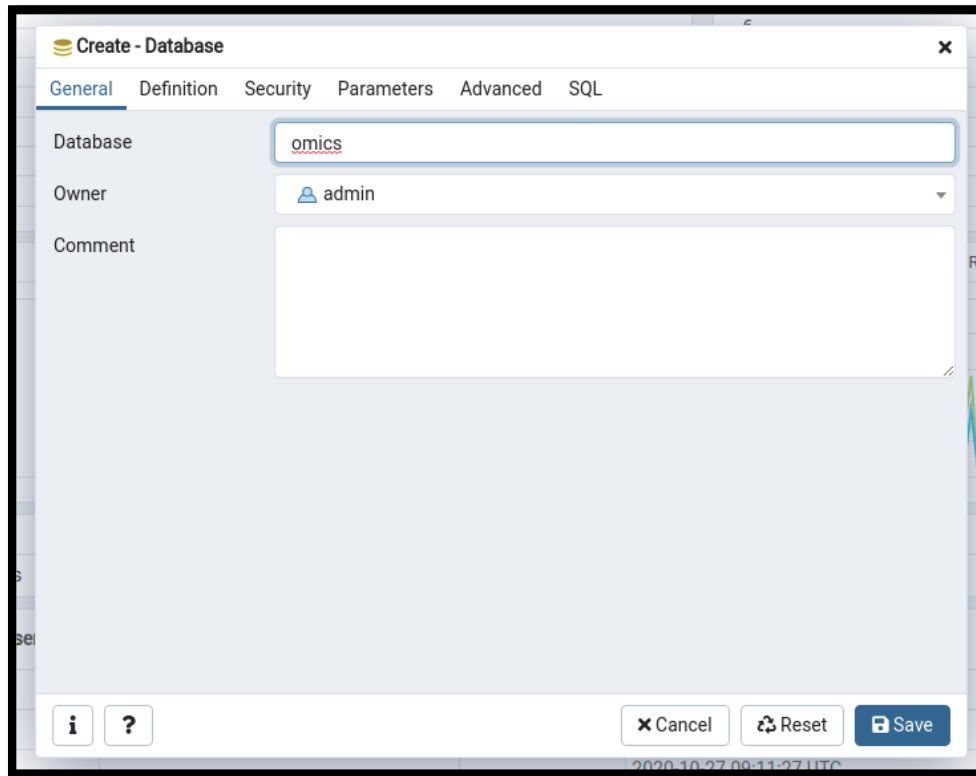


Figura 1. 126 Base de datos de nombre "omics". Es importante respetar el nombre.

### 5.3.9.2.2. Correr las migraciones

Debido a que se utiliza un motor de base de datos relacional, es necesario crear los esquemas principales que rigen las tablas de la base de datos. Para esto se hace uso de los scripts desarrollados para tal fin.

La plataforma utiliza el comando **make** y *Makefiles* (archivos de scripts) para automatizar las diversas tareas.

Desde el repositorio principal, raíz del proyecto, correr:

```
$ make clean-db migrate
```

```

+ > make clean-db migrate
make -C scripts clean-db
make[1]: se entra en el directorio '/home/kiriost/dev/omics/scripts'
npm run clean-db

> api@1.0.0 clean-db /home/kiriost/dev/omics/scripts
> node scripts/clean-db

[ CLEAN DATABASE ]
Dropping tables...READY
make[1]: se sale del directorio '/home/kiriost/dev/omics/scripts'
make -C scripts migrate
make[1]: se entra en el directorio '/home/kiriost/dev/omics/scripts'
npm run migrate

> api@1.0.0 migrate /home/kiriost/dev/omics/scripts
> node scripts/migrate

[ MIGRATE ]
Migrating...
0000_init.sql: RUN
0001_populate.sql: RUN
READY
make[1]: se sale del directorio '/home/kiriost/dev/omics/scripts'

```

Figura 1. 127 Ejecución del comando `make clean-db migrate`.

**make** permite correr más de un script al mismo tiempo y definir las dependencias entre ellos. Con lo anterior se ejecuta el script encargado de limpiar la base de datos y otro, más complejo, para correr las migraciones sobre la base de datos. Si volvemos a correr el script, las migraciones ya habrán impactado sobre la base de datos por lo que no volverán a ejecutarse.

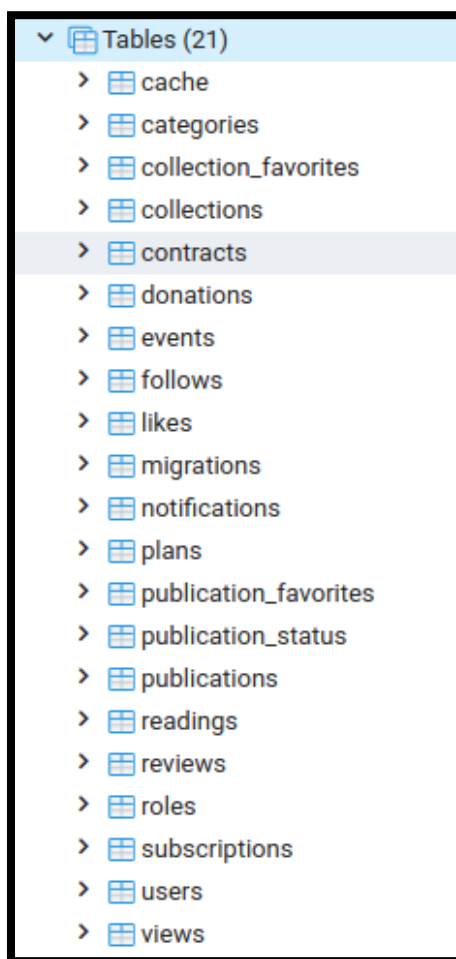


Figura 1. 128 Tablas de la base de datos de Omics

Así, obtenemos esta estructura de base de datos.

Cabe aclarar que **make migrate** además de generar los esquemas de las tablas también poblará la base de datos con los datos básicos iniciales: Reglas de negocio, roles, usuario administrador, usuario gestor de contenido, categorías, planes básicos.

### 5.3.9.2.3. Datos de prueba

En caso de que se necesitas datos de prueba, generados al azar, con el propósito de probar la plataforma, y la integración entre partes, puede ejecutarse:

```
$ make populate
```

Esto ejecutará un script desarrollado en NodeJS, encargado de poblar la base de datos con datos aleatorios, y, principalmente, consistentes.

```

> make populate
make -C scripts populate
make[1]: se entra en el directorio '/home/kiriost/dev/omics/scripts'
npm run populate

> api@1.0.0 populate /home/kiriost/dev/omics/scripts
> node scripts/populate

[ POPULATE ]
Populating DB...
READY
make[1]: se sale del directorio '/home/kiriost/dev/omics/scripts'

```

Figura 1. 129 Población de tablas con datos aleatorios con comando `make populate`.

### 5.3.9.3. Realización de backup

#### Estrategia para nombrar un archivo de backup

Es importante respetar la estandarización de nombres para generar archivos de backup.

Debe respetarse el siguiente formato: `YYYYMMDD_backup.sql`.

Siendo YYYY el año, MM el mes y DD el día actual. Seguido del sufijo “\_backup”.

#### 5.3.9.3.1. Manual

De **forma gráfica**, dirigirse al panel de administración pgAdmin. Autenticarse.

Hacer click derecho sobre la base de datos “omics” y seleccionar “Backup...”.

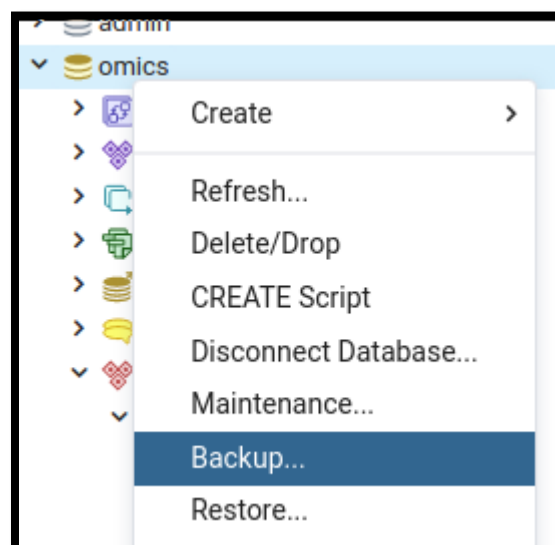


Figura 1. 130 Opción de Backup del Panel de administración de pgAdmin.

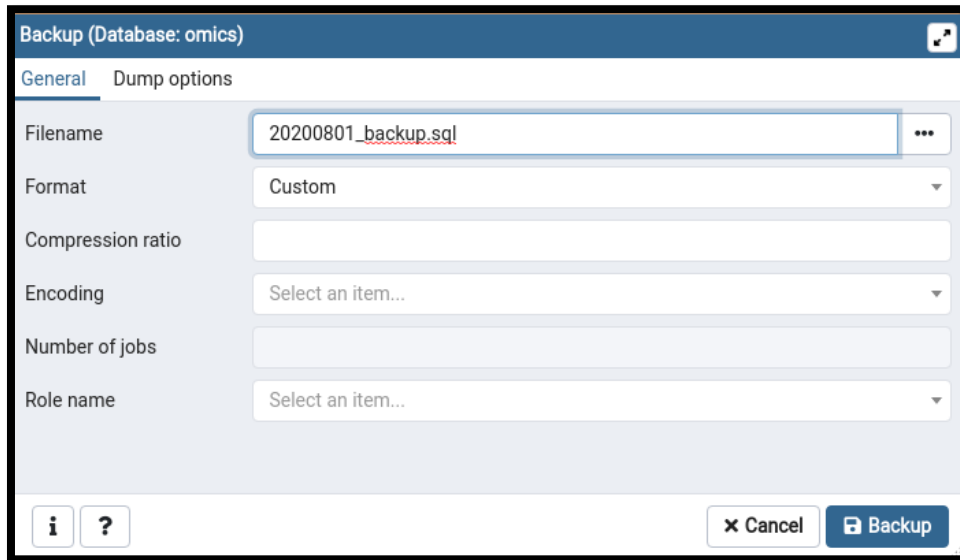


Figura 1. 131 Realización de backup por Interfaz gráfica.

Es recomendable realizar backups de forma manual a través de la consola, ya que se hace uso de un script generado para tal fin, el cual genera automáticamente los nombres de archivo.

Para esto último, se ejecuta:

```
$ make backup
```

Dicho comando ejecutará el script localizado en "omics/scripts/scripts/backup.sh". Así, de forma automática, se conectará a la base de datos y hará un backup completo de la base de datos "omics".

#### 5.3.9.3.2. Automático

Se genera una nueva entrada en **cron**, comando de GNU/Linux, en el servidor donde *Omics* se ejecute para producción.

Dicho comando únicamente debe correr **make backup** periódicamente. En caso de no estar disponible *make*, puede correr directamente el script que ejecuta:

```
$/omics/scripts/scripts/backup.sh
```

#### 5.3.9.3.3. Volcado de SQL

La idea de este método de volcado es generar un archivo de texto con comandos SQL que al realimentar al servidor, recreará la base de datos en el mismo estado que



tenía en el momento del volcado. PostgreSQL proporciona el programa de utilidad **pg\_dump** para este propósito.

### Por CLI (Cliente):

El uso básico de este comando es:

```
pg_dump -U usuario -W -h host basename > basename.sql  
pg_dump -U postgres -W -h localhost omics > 20201006_backup.sql
```

Se agrega la fecha del backup adelante del archivo sql de backup generado.

El comando anterior se automatiza mediante un proceso de cron para que corra todos los días a las 02:00 am.

### Explicación de los comandos:

- ❖ **U:** Se refiere al Usuario, en este caso puede ser el usuario propietario de la base de datos o el usuario postgres.
- ❖ **W:** Con este parámetro conseguiremos que nos solicite el password del usuario antes especificado.
- ❖ **h:** Con este indicamos cuál es el servidor PostgreSQL al que nos conectaremos para obtener nuestro dump, si estamos local podemos colocar localhost sino ponemos la IP del servidor PostgreSQL.

**basename:** Este es el último parámetro realmente en esta línea de comando, por esa razón no tiene alguna letra que indique que el siguiente parámetro es el nombre de la base de datos.

**basename.sql:** Esta parte en realidad solo indica que la salida de nuestro comando pg\_dump la guarde en un archivo basename.sql.

### Por GUI (Interfaz):

Se utiliza el panel de gestión de PostgreSQL pgAdmin4. Este dashboard es online, y se accede mediante el navegador web (editado).

A través de la GUI se pueden realizar backup de la DB manualmente, como se observa en las siguientes figuras:

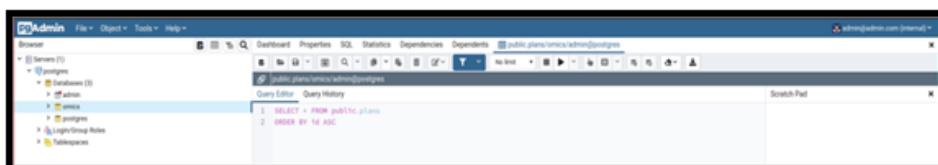


Figura 1. 132 Interfaz gráfica de PgAdmin.

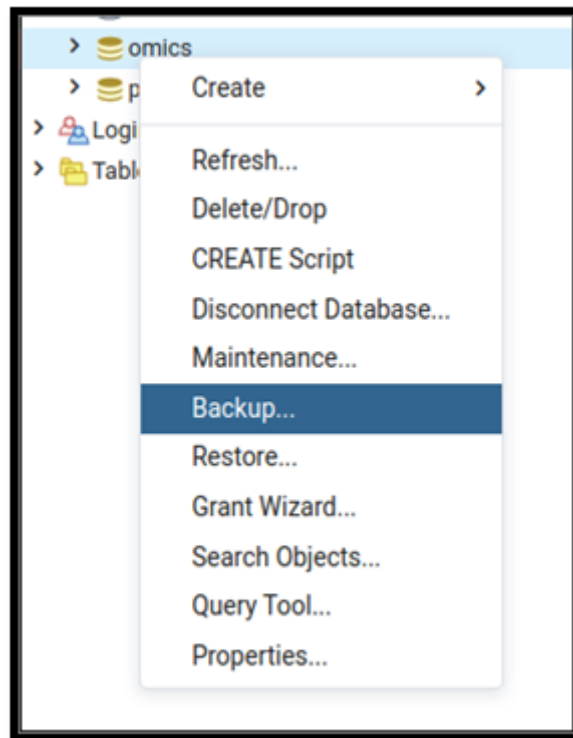


Figura 1. 133 Realizar backup por PgAdmin.

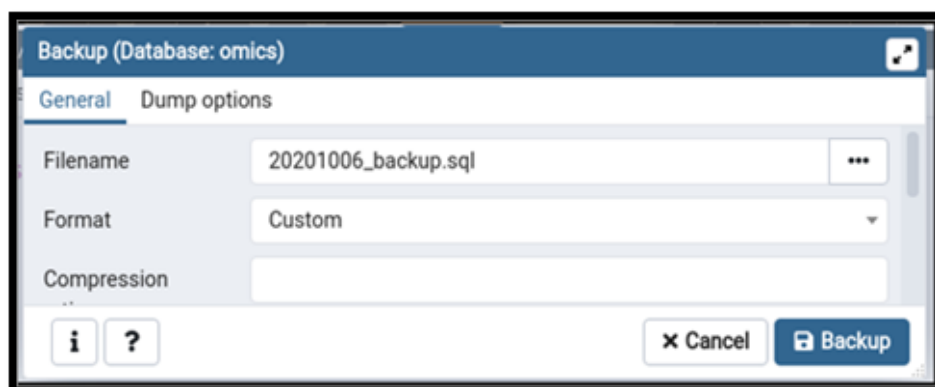


Figura 1. 134 Opciones de Backup.

#### 5.3.9.3.4. Migraciones

Por cada nueva versión de la plataforma que se quiera poner en producción, e incluso en entornos de desarrollo y testing, debe ejecutarse:

**\$ make migrate**

Lo que provocará que las nuevas migraciones, aún no impactadas en la base de datos sobre la que se trabaja, corran.

### 5.3.9.3.5. Eventos

Ya que la plataforma de Omics utiliza Event-Sourcing como metodología, donde cada cambio en las entidades emite un evento, y cada evento es almacenado en la DB a modo de event store, se puede realizar un backup únicamente de los eventos ocurridos.

De esta forma, si los eventos ocurridos desde el comienzo de la plataforma (corriendo en producción) se vuelven a aplicar a todos los modelos o entidades, se consigue el mismo estado en el que se encuentra actualmente.



id	topic	code	timestamp	payload
1	user	validated	2020-10-06 21:30:23.004982+00	[123, 34, 86, ...]
2	publication	created	2020-10-06 21:30:54.892643+00	[123, 34, 67, ...]
3	user	logged-in	2020-10-06 21:30:36.01145+00	[123, 34, 76, ...]
4	user	updated	2020-10-06 21:30:47.504593+00	[123, 34, 85, ...]
5	user	registered	2020-10-06 21:29:55.80632+00	[123, 34, 82, ...]

Figura 1. 135 Backups realizados.

De esta forma, si se vuelve a correr el gestor de migraciones (un script especial desarrollado en JavaScript) sólo se ejecutarán y su resultado será reflejado sobre el esquema de DB aquellos que no hayan sido ya ejecutados.

### 5.3.9.3.6. Métodos de Replicación

El sistema de gestión de bases de datos elegido "PostgreSQL" posee un método de replicación.

Es la transmisión de información derivada de las operaciones DML (Data Manipulation Language) de una Base de Datos a otra, es decir, se transmite a otra Base de Datos las instrucciones INSERT, UPDATE y DELETE que se realicen en una Base de datos de forma que ambas BBDD tengan la misma información, así pues, se obtiene una redundancia de datos.

#### Replicación en PostgreSQL:

Uno de los métodos más usados para conseguir una alta disponibilidad en nuestro servidor Postgres es implementar la Replicación. Con ello nos aseguramos de que nuestra plataforma Omics, esté activa el 99,9% del año.

PostgreSQL dispone de varias herramientas y métodos para implementar:

#### Tipos:

- ❖ Basada en triggers.
- ❖ Basada en ficheros WAL (Write Ahead Log).

### **Técnicas:**

- ❖ Archivado continuo (WAL Shipping).
- ❖ Streaming Replication.

### **Forma de transmisión:**

- ❖ Asíncrona.
- ❖ Síncrona.

### **Tipos de replicación:**

Un trigger (Disparador), es una acción que se ejecuta con cada operación DML. Utiliza una cola para almacenar los cambios y enviarla a la Base de Datos remota de forma asíncrona. Su ventaja, se puede seleccionar un subconjunto de una o varias tablas de las Bases de Datos.

Un fichero WAL es un registro con los INSERT, UPDATE y DELETES que se han realizado en una Base de Datos. Los ficheros WAL se generan para garantizar la durabilidad de la Base de Datos.

### **Técnicas de implementación:**

PostgreSQL puede enviar la información de los registros WAL mediante un archivado continuo (WAL Shipping) entre el servidor maestro y el servidor esclavo. También se puede configurar para que se envíen los ficheros WAL con las operaciones DML de una Base de Datos a otra mediante la red (Streaming Replication). El modo de envío de estos ficheros puede ser de forma asíncrona o síncrona.

### **Forma de Backup**

La base de dato PostgreSQL debe ser respaldada regularmente.

Existen tres enfoques fundamentales diferentes para hacer una copia de seguridad de los datos de PostgreSQL:

- ❖ Volcado de SQL.
- ❖ Copia de seguridad a nivel de sistema de archivos.
- ❖ Archivo continuo.

Cada uno tiene sus propias fortalezas y debilidades. Por el momento se va a utilizar el tipo "Volcado de SQL".

### **5.3.9.4. Recuperación**

Siguiendo el mismo método del apartado anterior para la restauración se va a aplicar el siguiente comando:

```
psql -U username -W -h host basename < basename.sql  
psql -U postgres -W -h localhost omics < omicsBackup.sql
```

#### 5.3.9.4.1. Las tareas para realizar la Recuperación son las siguientes:

Se selecciona el backup más reciente, lo cual puede ser determinado gracias al nombre del archivo del backup generado con anterioridad, como, por ejemplo: "20200501\_backup.sql", el cual es el archivo de backup correspondiente al 01/05/2020. Antes de cargar el archivo, se limpia la base de datos completa y se ejecutan las migraciones para reconstruir los esquemas necesarios, como ya se comentó: "make clean-db migrate".

Una vez realizado esto, se accede al dashboard de administración de la base de datos principal, PostgreSQL. Y se carga el archivo de backup.

Para comprobar el correcto funcionamiento, existen archivos de script que pueblan la base de datos con datos "fake", es decir, datos de prueba. Para esto se ejecuta: "make populate".

Esto sirve para comprobar que tanto el back-end como la DB están manejando los mismos esquemas de datos. Debería realizarse antes de cargar el backup real para apaciguar dudas.

Para comprobar que se ha cargado correctamente el backup, se puede corroborar desde el dashboard de administración de la DB, pgAdmin. Además, para comprobar que funciona correctamente junto a la lógica del back-end se han creado tests de integración en forma de scripts. Estos tests realizan llamadas a la API del sistema y comprueban que las respuestas que se reciben son las esperadas. Se pueden ejecutar con "make api-test", y corren gracias a NodeJS y el framework de testing Jest.

La plataforma Omics se inserta en el mercado para prestar servicio en dos fases:

La primera tiene como objetivo la generación de la comunidad en torno a la plataforma. Una vez establecida una comunidad lo suficientemente grande, entra en juego la segunda fase que se encarga de asentar las estrategias de negocio y el modelo económico que se quiere adoptar para generar beneficios en la organización, pero manteniendo también los beneficios que representa para todo tipo de usuario de la misma.

Cada fase, a su vez, utiliza distintas estrategias de conversión e implementación, por lo cual la conversión resultante es una combinación de distintos métodos. También, se tiene en cuenta el uso interno del sistema, para el Administrador encargado del "Control de Contenido".

La implementación e inserción al mercado de Omics, son críticas para el éxito de la plataforma. Dicha implementación no es tradicional en el sentido de que no conlleva únicamente adaptar a los usuarios para mejorar su eficiencia, sino que también se aprovecha la retroalimentación de estos para mejorar la calidad del servicio brindado por la plataforma, en términos de usabilidad y performance.

### 5.3.9.5. Backup por Sistema

#### ❖ Pantalla Backup Automático

El Administrador General, si así lo desea puede realizar el backup automático de la plataforma, para lo cual debe estar logueado y seguir los siguientes pasos:

**Paso 1)** - Ir al **Menú Desplegable** ubicado en la esquina superior izquierda y seleccionar la opción **Backups**.

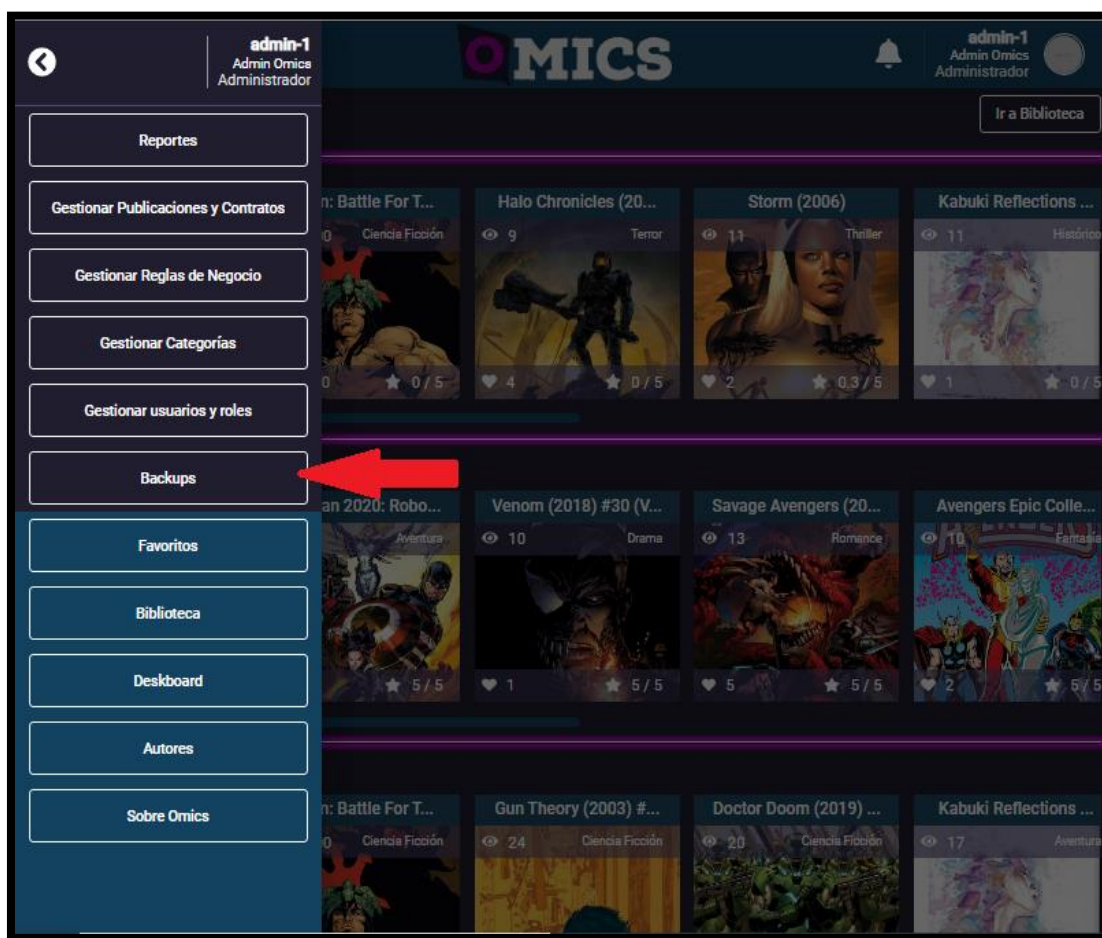


Figura 1. 136 Pantalla del Administrador Opción Backups.

**Paso 2)** - Se despliega una ventana donde se puede visualizar una tabla con la siguiente información: la fecha del último backup realizado, el archivo de backup

generado, la opción de descarga del archivo de backup, y el botón Generar nuevo backup.

**Paso 3)** - Para realizar el backup automático, seleccionar botón **Generar nuevo backup**.

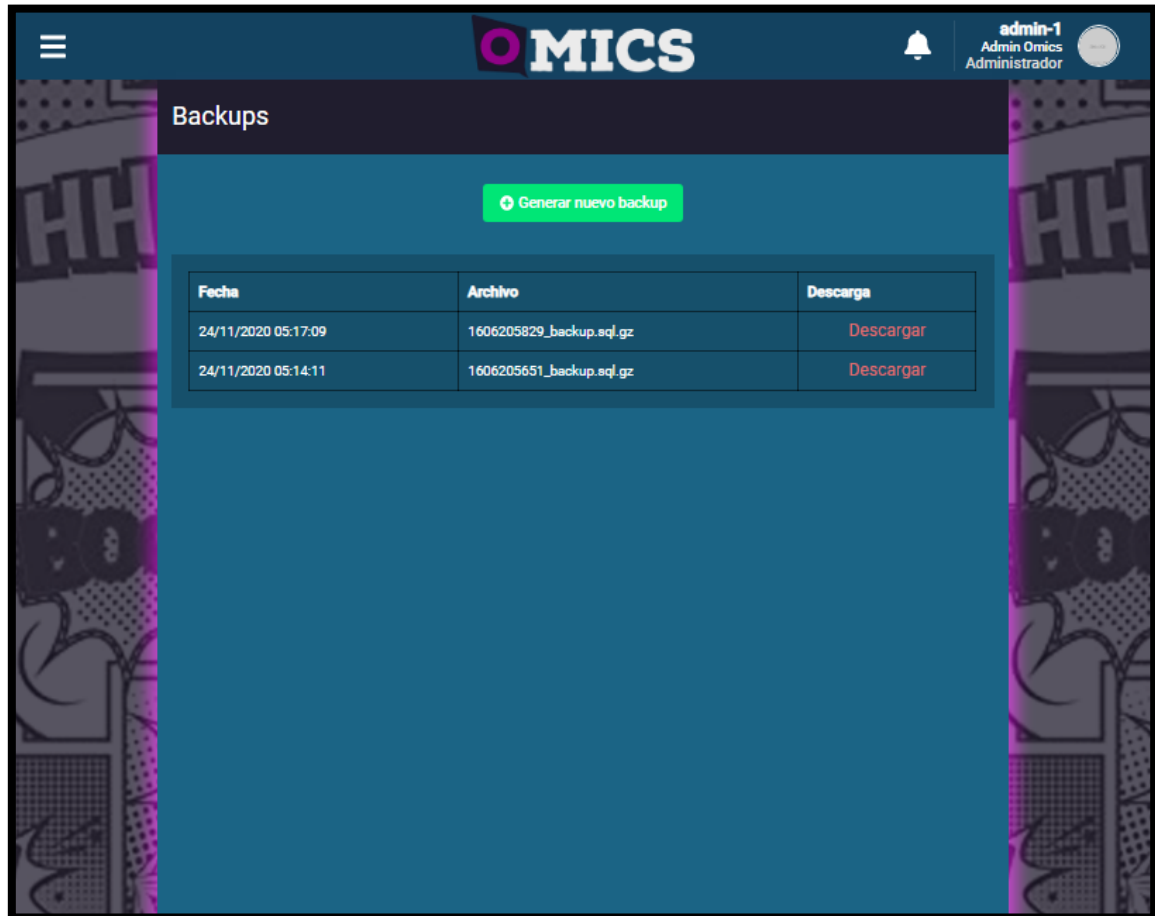


Figura 1. 137 Pantalla ventana de Backups.

Se pueden descargar los backups realizados para su uso posterior ante eventualidades, traslado de información y recuperación.

### 5.3.10. Etapas del Método de Conversión

Se nombran las etapas del método de conversión de la siguiente manera:

- ❖ **Primera Etapa:** Generación de comunidad, atracción de usuarios nuevos que ya hayan tenido experiencia con plataformas similares o no. Se enfoca en mejorar la usabilidad para convencer a los usuarios.

- **Piloto:** Se les brinda a los usuarios un acceso anticipado a la plataforma, en términos de pruebas *beta*, antes de lanzar oficialmente el sistema al mercado. Tiene como objetivo dar a conocer la plataforma, evaluar el desenvolvimiento de los usuarios en la misma al recibir retroalimentación, y preparar el sistema para lanzarlo oficialmente. Pone a prueba el nuevo sistema, tiene bajo costo, pero deriva en una etapa crítica para el éxito de la plataforma.

#### ❖ Segunda Etapa:

- **Directo:** En mayor medida no se puede obligar a un usuario a abandonar otras plataformas, incluso hasta podría ser negativo para *OmicS*. Sin embargo, a partir de esta etapa el foco ya no está en mantener funcionalidades similares a otros sistemas para hacer más amena la transición de nuevos usuarios entrantes, sino en expandir el rango de funcionalidades presentes y planificadas. Marcando así una diferencia que destaque a la plataforma frente a otras del mismo rubro. Este método de implementación, con el sistema funcionando de forma estable continuamente, se basa en evaluar las expectativas de los usuarios existentes para satisfacerlas mediante nuevas o mejores funcionalidades. Un usuario satisfecho con el sistema es probable que abandone las demás plataformas para hacer uso completo de *OmicS*.

### 5.3.11. Instalación del Sistema

A continuación, se explica la instalación del Back end y Front end:

#### 5.3.11.1. Instalación del Back-end

El back-end de *OmicS* se ejecuta sobre un VPS, servidor privado virtual.

Los servicios de los que depende el back-end se ejecutan sobre contenedores de Docker, por lo tanto, los servicios de base de datos, caché, archivos, etc., se ejecutan mediante Docker y están orquestados mediante `docker-compose`.

Por esto, para levantar los servicios sólo se requiere clonar el repositorio del proyecto, acceder a la rama master y ejecutar sobre línea de comandos (terminal):

```
docker-compose up -d
```

Con lo anterior, todos los servicios estarán corriendo en contenedores, con la configuración que se encuentran en "`docker-compose.yml`".



El backend, desarrollado en Rust, utiliza **cargo** como manejador de paquetes y gestor de configuración. Levantar el back-end sólo requiere compilarlo y ejecutarlo como un binario más, gracias a las bondades de Rust:

```
cargo build --release  
./target/release/omics
```

El proyecto se compila a un binario y luego se ejecuta. Esta compilación puede realizarse en un servicio de Integración Continua y realizar deploy únicamente de este binario al servidor, por lo que sólo el último comando sería requerido.

A su vez, todo lo anterior ha sido resumido en un archivo encargado de ejecutar las distintas tareas requeridas para levantar, operar, depurar y realizar testing del proyecto. Gracias a make (binario de UNIX encargado de automatizar comandos).

Por lo tanto, levantar los servicios en Docker, limpiar la base de datos (en caso de ser necesario), ejecutar las migraciones, correr los tests unitarios y de integración, y, por último, ejecutar el back-end, puede resumirse en una serie de comandos:

```
make run-services  
make clean-db  
make migrate  
make build  
make run
```

O todo junto:

```
make run-services clean-db migrate build run
```

En caso de que no se quiera limpiar la DB, sino únicamente ejecutar las migraciones para actualizar los esquemas de la base de datos, se omitiría *"clean-db"*.

Cabe aclarar que Rust, gracias a cargo, su manejador de paquetes oficial, instala automáticamente todas las dependencias declaradas en el archivo especial llamado *"Cargo.toml"*.

### 5.3.11.2. Instalación del Front end

- 1) Instalar node (versión 10 o superior) en [www.nodejs.org](http://www.nodejs.org).
- 2) Instalar angular CLI mediante un terminal: *"npm install -g @angular/cli"*.

- 3) Dentro de la carpeta del proyecto, ir a la carpeta "web" y abrir una terminal ahí para ejecutar: *"npm install"*.
- 4) Desde la carpeta web, ejecutar: *"ng serve"*.
- 5) Ir a *"http://localhost:4200/"* para visualizar la plataforma.

#### **5.4. Estimación de tiempos**

La planificación de la Implementación del Sistema se encuentra reflejada en **VER ANEXO N.º 1 DIAGRAMA DE GANTT**. En el que se determinan los tiempos de las actividades mencionadas en **SECCIÓN 5.3. ACTIVIDADES** de forma general.

# Trabajo Práctico N°2:

## *Planificación de Proyectos Informáticos*

## CAPÍTULO I: Actividades

### 1. Definición y descripción de actividades

De acuerdo con lo definido en Alcance del proyecto podemos, de forma general, definir las actividades generales, y la subdivisión de las mismas. Estas se desglosarán a continuación según lo establecido en la **FIGURA 2. 1**.

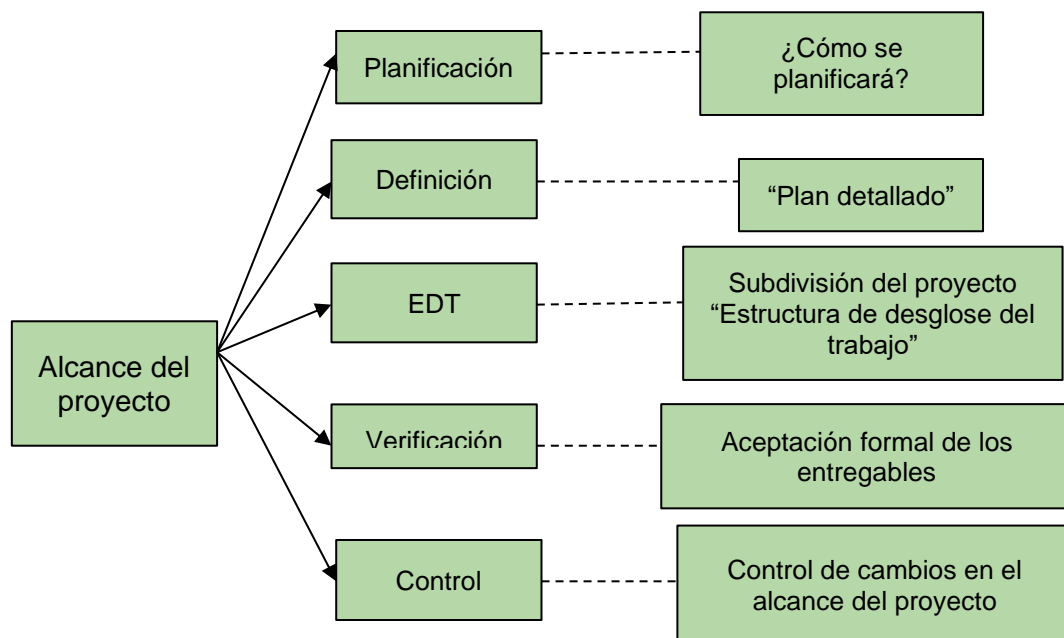


Figura 2. 1 Planificación a partir de la definición del alcance de proyecto.

El trabajo está dividido en actividades de la siguiente forma:

#### ❖ **Planificación.**

##### ➤ **Definir actividades.**

Identificar las acciones que deben ser llevadas a cabo para conseguir los entregables del proyecto y armar un cronograma de trabajo. Documentar las tareas y actividades a realizar durante el desarrollo del proyecto.

##### ➤ **Estimar tiempos de actividades.**

Estimación de tiempo y recursos que implica el desarrollo de las mismas. Se actualiza esta estimación en base al progreso del proyecto.

##### ➤ **Definir equipo de trabajo y funciones principales.**

Se determinan los distintos roles que son necesarios para afrontar las responsabilidades de la planificación y el desarrollo del sistema. Establece la base de división de tareas con sus funciones principales.

➤ **Definir herramientas de gestión de proyecto.**

Actividad importante ya que acompaña a la gestión y progreso de todo el proyecto. Investigar diferentes alternativas y definir la que mejor se adecue al proyecto y al equipo de trabajo.

➤ **Configurar herramientas de gestión de proyecto.**

Configuración e integración entre las herramientas utilizadas para realizar seguimiento al proyecto en su totalidad.

➤ **Definir métodos de comunicación formal.**

Se definen los métodos de comunicación por medio de los cuales queda acentuado lo más relevante de la comunicación entre el equipo de trabajo e, incluso, otros participantes.

➤ **Gestión de configuración del software.**

Se analizan las diferentes alternativas para determinar cómo se lleva a cabo el desarrollo y versionado del producto.

➤ **Realizar análisis de factibilidad.**

Se contemplan los riesgos y costos, para saber si el proyecto es viable.

➤ **Realizar análisis de riesgo.**

Se estudian los riesgos que se asumen al comenzar el proyecto, y las consecuencias de estos.

➤ **Tratamiento de medidas preventivas.**

- Entrenar a los trabajadores en procedimientos para detectar requerimientos. Desarrollar análisis de Seguridad de la Tarea y procedimientos seguros de trabajo.
- Organizar y llevar a cabo capacitaciones referidas a las tecnologías que vamos a usar en el proyecto. La persona con más habilidad en la tecnología puede capacitar al miembro del equipo que lo necesite. También se puede contar con la compra de cursos webs.
- Se realiza un estudio detallado e investigación para conocer mejor el negocio. Contamos con contactos que pueden brindarnos asesoramiento más detallado de la industria que planteamos para utilizar en el proyecto.
- Se lleva un control coordinado del desarrollo de funciones, replanteo de alguna librería o tecnología a usar, y una vez que finalice la función realizada por alguno de los integrantes se

pasará a otro del equipo para que lo revise. Realizar refactoring en las partes críticas del sistema regularmente.

- Se da prioridad a la tarea que genera el retraso, con asistencia al miembro que lo necesite para acelerar su finalización, o visualizar una solución alternativa que no cause consecuencia en cascada.
- Generar una comunidad que respalde el proyecto y ayude a dar a conocer el mismo. Establecer tiempos a largo plazo coordinando las tareas fundamentales. Considerar mercados potenciales basado en el contenido que se vaya generando en los períodos de prueba en desarrollo.

#### ❖ **Definición de Requerimientos.**

##### ➤ **Investigar aplicaciones similares.**

Averiguaciones sobre aplicaciones que sean parecidas para obtener más información.

##### ➤ **Relevar y analizar en general sistemas similares.**

Averiguar cuáles sistemas o aplicaciones existen en el mercado que ofrecen un servicio similar al propuesto por el proyecto.

##### ➤ **Detectar funciones e interfaces en sistemas similares.**

Exploración detallada de funciones e interfaces seleccionadas que ofrece cada aplicación. Se realiza un análisis de las mismas para dar forma a las ideas y funcionalidades planteadas inicialmente en el proyecto.

##### ➤ **Detectar tecnologías en sistemas similares.**

Exploración de las tecnologías utilizadas por aplicaciones y sistemas similares. Breve análisis de las mismas para encontrar oportunidades de utilización en el proyecto o falencias.

##### ➤ **Elaborar modelo lógico de sistemas relevados.**

Elaborar un modelo lógico que resuma de manera general las funcionalidades a implementar, provenientes de los sistemas similares.

##### ➤ **Elaborar comparación de funcionalidades de sistemas similares.**

Comparar las funcionalidades ofrecidas por las aplicaciones y sistemas relevados, seleccionar aquellas que sean de incumbencia para el proyecto y adaptarlas al mismo.

##### ➤ **Justificar proyecto a partir de necesidades.**

A partir de las necesidades que se han detectado, confeccionar la documentación relacionada a la fundamentación del desarrollo del proyecto, con los objetivos y soluciones propuestas.

##### ➤ **Definir objetivos y alcances preliminares del nuevo sistema.**

Objetivos y alcances preliminares a partir de las posibles necesidades detectadas.

❖ **Preparación entrega 1: “Etapa de Definición de Requerimientos”.**

❖ **Entrega 1.**

❖ **Diseño.**

➤ **Definir objetivos y alcances definitivos del sistema.**

Se refinan y definen los objetivos y alcances del sistema teniendo en cuenta el relevamiento anterior.

➤ **Definir módulos del sistema.**

De acuerdo con el modelo lógico general y el establecimiento de los objetivos y alcances del nuevo sistema, se definen los módulos del sistema, la interacción y relación entre los mismos.

➤ **Definir salida del sistema.**

Se definen las salidas del sistema que generan los módulos, con sus respectivos Casos de Uso. Cuentan con información que se muestra por pantalla en la interacción con el usuario.

➤ **Definir interfaces de usuario.**

Maquetado, bosquejos y creación de las interfaces de usuario que se utilizan en el nuevo sistema. Dan un vistazo general al flujo de información entre lo visible para el usuario, lo que este necesita en cada instancia (pantalla) de la aplicación y lo que el *back-end* resuelve.

➤ **Definir métodos de comunicación entre partes del sistema.**

Se definen los métodos de comunicación entre el *front-end* y el *back-end*, y principalmente se establece la arquitectura del sistema a utilizar.

➤ **Definir historias de usuario.**

Se definen los requisitos a implementar en forma de Historias de Usuario. Permite especificar los requisitos del sistema de una forma simple.

➤ **Definir Sprint.**

Se definen las actividades a implementar en cada iteración, las fechas de inicio y fin de cada una y demás consideraciones a tener en cuenta durante cada *sprint*.

➤ **Definir modelos de datos.**

Se diseñan y modelan las entidades con las que interactúa el sistema. Se modela la base de datos que da soporte al sistema. Depende de los módulos definidos genéricamente.

❖ **Preparación entrega 2: “Etapa de Diseño”.**

❖ **Entrega de Trabajo Práctico Integrador N.º 1**

- ❖ **Entrega 2.**
- ❖ **Correcciones generales de las etapas anteriores.**
- ❖ **Trazabilidad general de los elementos del sistema.**
- ❖ **Desarrollo.**

- **Definir tecnologías de desarrollo.**

Se definen todas las tecnologías a utilizar en el proyecto. Se realizan las pruebas necesarias para evaluar funcionalidades disponibles, rendimiento ofrecido y características importantes. Se consideran aquellas utilizadas en proyectos similares, para detectar pros y contras.

- **Capacitarse en las tecnologías elegidas.**

Investigación y capacitación profunda en aquellas tecnologías en las que se tenga poco conocimiento.

- **Planificar implementación de los módulos del sistema.**

Se planifica primero la implementación de los módulos más importantes para el mejoramiento del proyecto, y se plantea que progresivamente se adquieran los módulos faltantes. Esto trae mejor solución de problemas que se van presentando por falta de integración con algunos procesos.

- **Planificar pruebas del sistema.**

Es importante la planificación de pruebas del sistema para poder identificar la información existente en el proyecto y los componentes que deben ponerse a prueba; listar los principales requisitos a probar; definir las estrategias de prueba que deben emplearse e identificar los recursos que necesarios.

- Planificar pruebas de Validación de Ingresos de Datos.  
Para poder detectar fallas en cada uno de los componentes que permiten el ingreso de los datos al Sistema.
- Planificar pruebas de Lógica de los Módulos Principales.  
Para poder detectar fallas en cada una de las componentes más importantes del Sistema, que representan las reglas del negocio principales o requisitos tecnológicos destacados.
- Planificar pruebas de integración entre los módulos del sistema.  
Se verifica que los distintos módulos se comunican e interactúan correctamente y el sistema una vez integrado cumple con las funciones especificadas de forma adecuada.
- Planificar pruebas de carga.  
Para poder encontrar fallas en el funcionamiento simulado del Sistema, en lo que hace carga máxima o en situaciones de stress



o límite, respecto de sesiones simultáneas, VPNs, acceso a datos, autorizaciones en línea, tasas de transferencia, rendimiento de código, rendimiento de APIs, tiempo de respuesta transaccional, servidores, servicios, integración con otros Sistemas, etc.

- Planificar pruebas de seguridad por niveles de usuario.

Para poder encontrar fallas en el módulo de seguridad desarrollado y comprobar la integración con el resto del sistema de información, verificando el funcionamiento correcto de las interfaces que lo componen y el resto de los módulos con los que se comunica.

➤ **Programar módulos del sistema.**

Desarrollo (codificación, programación) de los diferentes módulos del sistema. Proceso amplio, soportado por metodología ágil. Cada módulo es propenso a ser modificado en el futuro, por lo que podemos considerar esta actividad como la más dinámica del proyecto.

- **Módulos del sistema back-end.**

Se programan los módulos del sistema en el back-end para contemplar las funcionalidades que se propusieron en el proyecto.

- Módulo de Registro y Actividad de Usuario.
- Módulo de Catálogo y Búsqueda.
- Módulo de Visualización y Seguimiento.
- Módulo de Reseñas.
- Módulo de Suscripción y Donación.
- Módulo de Publicación de Obras.
- Módulo de Notificaciones.
- Módulo de Contrato.
- Módulo de Reportes.
- Módulo de Configuración.

- **Módulos del sistema front-end.**

Se programan los módulos del sistema en el front-end para contemplar las funcionalidades que se propusieron en el proyecto.

- Módulo de Registro y Actividad de Usuario.
- Módulo de Catálogo y Búsqueda.
- Módulo de Visualización y Seguimiento.
- Módulo de Reseñas.
- Módulo de Suscripción y Donación.
- Módulo de Publicación de Obras.
- Módulo de Notificaciones.

- Módulo de Contrato.
- Módulo de Reportes.
- Módulo de Configuración.
- **Documentación Módulos del Sistema.**

La documentación se realiza para guardar información donde se muestra qué hace cada módulo en el sistema, cómo lo hacen y para quién. En este material se explican las características técnicas y la operación de los módulos.
- **Revisión de módulos del sistema.**

Para controlar si cumple con las funcionalidades establecidas o si hay que realizar alguna modificación.

  - Módulo de Registro y Actividad de Usuario.
  - Módulo de Catálogo y Búsqueda.
  - Módulo de Visualización y Seguimiento.
  - Módulo de Reseñas.
  - Módulo de Suscripción y Donación.
  - Módulo de Publicación de Obras.
  - Módulo de Notificaciones.
  - Módulo de Contrato.
  - Módulo de Reportes.
  - Módulo de Configuración.
- **Ejecución de pruebas unitarias.**

Para comprobar que un fragmento de código funciona correctamente. Estas pruebas validan el comportamiento de un objeto y la lógica. Demuestran que la lógica del código está en buen estado y funciona.

  - Ejecutar pruebas unitarias Módulo de Registro y Actividad de Usuario.
  - Ejecutar pruebas unitarias Módulo de Catálogo y Búsqueda.
  - Ejecutar pruebas unitarias Módulo de Visualización y Seguimiento.
  - Ejecutar pruebas unitarias Módulo de Reseñas.
  - Ejecutar pruebas unitarias Módulo de Suscripción y Donación.
  - Ejecutar pruebas unitarias Módulo de Publicación de Obras.
  - Ejecutar pruebas unitarias Módulo de Notificaciones.
  - Ejecutar pruebas unitarias Módulo de Contrato.
  - Ejecutar pruebas unitarias Módulo de Reportes.
  - Ejecutar pruebas unitarias Módulo de Configuración.
- **Resultados de Pruebas unitarias.**

Permiten al desarrollador refactorizar el código más adelante y tener la garantía de que el módulo sigue funcionando correctamente. Para ello se escriben casos de prueba para todas las funciones y métodos, para que cada vez que un cambio provoque un error, sea posible identificarlo y repararlo rápidamente. El resultado puede ser de éxito o fracaso, según el objetivo que se haya propuesto para la prueba.

➤ **Revisión completa de Pruebas unitarias.**

La ejecución de pruebas unitarias bien realizadas sirve como documentación del proyecto. Aumentan la legibilidad del código y ayudan a los desarrolladores a entender el código base, lo que facilita hacer cambios más rápidamente.

➤ **Ejecución de pruebas.**

Para proporcionar información sobre la calidad del software y llevar un control de las funcionalidades. Dependiendo del tipo de pruebas, será el nivel que corresponde de involucramiento en las actividades de desarrollo.

- **Ejecución pruebas de Validación de Ingresos de Datos.**

Para validar que se ha especificado lo que el usuario realmente quería. Además, se comprueba si los valores que ha introducido el usuario cumplen las restricciones impuestas por la aplicación.

- Usuario crea cuenta en la plataforma 01.
- Usuario crea cuenta en la plataforma 02.
- Usuario quiere registrarse en la plataforma 03.

- **Ejecución pruebas de Lógica de los Módulos principales.**

Para lograr combinar módulos para que trabajen correctamente de forma conjunta.

- Usuario paga suscripción 01.
- Autor adjunta la portada al publicar una obra 02.
- Administrador aprueba publicación 03.

- **Ejecución pruebas de integración entre los módulos del sistema.**

Para encontrar fallas en el Sistema y verificar el correcto ensamblaje entre los distintos componentes con el fin de comprobar que interactúan correctamente a través de sus interfaces, tanto internas como externas. Cubren la funcionalidad establecida y se ajustan a los requisitos no funcionales especificados en los alcances del sistema.

- Colecciones y Categorías existentes 01.

- Ver estadística de publicación al dar like 02.
- Publicar y visualizar el contenido de la obra 03.
- **Ejecución pruebas de carga.**

Para validar el rendimiento de funciones críticas en términos de complejidad de procesamiento, utilización de recurso o cantidad de usuarios en línea.

  - Peticiones por segundo de la API REST al endpoint de usuarios 01.
  - Peticiones por segundo de la API REST al endpoint de usuarios 02.
  - Peticiones por segundo de la API REST al endpoint de usuarios 03.
  - Peticiones por segundo de la API REST al endpoint de usuarios 04
- **Ejecución pruebas de seguridad por niveles de usuario.**

Para comprobar que cada usuario sólo puede acceder a las funcionalidades pertinentes para ese rol, evitando accesos no deseados a funciones que no debe poder ejecutar. Y solo permitir al usuario administrador ejecutar funciones propias de la configuración general del sistema.

  - Ingreso de Usuario Administrador 01.
  - Ingresar a una ruta sin permisos 02.
  - Autor puede visualizar reportes y detalles de sus publicaciones 03.
- **Resultados de Ejecución pruebas de Validación de Ingresos de Datos.**

Validación correcta de los datos ingresados.
- **Resultados de Ejecución pruebas de Lógica de los Módulos Principales.**

Interacción correcta entre los módulos principales del sistema.
- **Resultados de Ejecución pruebas de integración entre los módulos del sistema.**

Interacción correcta entre componentes de distintos módulos.
- **Resultados de Ejecución pruebas de pruebas de carga.**

Debe ser un resultado que debe cumplir con los criterios de tiempo y rendimiento de cada prueba.
- **Resultados de Ejecución pruebas de seguridad por niveles de usuario.**

Visualización de funciones adecuadas al rol.
- **Preparar demo 1 del sistema.**

Se realiza el ensayo de la presentación a exponer, y se revisa que la demo esté acorde a lo establecido, para evaluar su posterior entrega.

- **Entrega Trabajo Práctico Integrador N.º 2**
- **Entregar demo 1.**
- **Preparar demo 2 del sistema.**

Se revisan las correcciones realizadas sobre la demo 1 y se realiza el ensayo de la presentación a exponer para corroborar que la demo esté acorde a lo establecido, para evaluar así su posterior entrega.

- **Entregar demo 2.**
- **Revisión de módulos del sistema.**

Para controlar si cumple con las funcionalidades establecidas o si hay que realizar alguna modificación.

- Módulo de Registro y Actividad de Usuario.
- Módulo de Catálogo y Búsqueda.
- Módulo de Visualización y Seguimiento.
- Módulo de Reseñas.
- Módulo de Suscripción y Donación.
- Módulo de Publicación de Obras.
- Módulo de Notificaciones.
- Módulo de Contrato.
- Módulo de Reportes.
- Módulo de Configuración.

#### ❖ **Implementación.**

- **Planificación de las tareas.**

Se establecen las tareas con previsión acerca de las actividades que se deben realizar y estimar los tiempos de cada una.

- **Asignar los responsables.**

Se deben asignar las tareas a realizar, para que cada responsable pueda identificar su relación con las etapas de puesta en funcionamiento del sistema.

- **Especificaciones de Servidores.**

Se detallan los servidores que se utilizan en la implementación, las características y funcionalidad que presentan.

- **Equipos de Conectividad.**

Se establece el lugar de realización de las actividades del equipo.

- **Comunicaciones.**

Se establece la forma en que los equipos se comunican entre sí y la difusión de las actividades por redes sociales.

- **Puestos de Trabajo.**

Se establecen los puestos de trabajos que se encuentran presentes en el desarrollo de la plataforma *Omics*.

➤ **Infraestructura.**

Se detallan los elementos fundamentales para que la plataforma *Omics* se ponga en funcionamiento.

➤ **Carga Inicial de Datos.**

Se identifica y documenta la carga inicial de los datos con los que se llena la base de datos de la plataforma. Se establece el orden de ejecución de las partes, como son: Carga Inicial, Conversión de Datos y Depuración.

➤ **Migración.**

Se detalla la forma de migración de las tablas de la base de datos.

➤ **Métodos de Replicación.**

Se especifica el método de replicación que utiliza la base de datos Postgres y las herramientas de la misma. Se detallan los tipos de replicación y las técnicas que se utilizan en la implementación.

➤ **Backups.**

Se establecen los pasos y procedimientos para realizar el respaldo de los datos de la plataforma. Se explican los comandos que se utilizan para realizar los backups.

➤ **Recuperación.**

Se detallan los pasos para realizar la recuperación de los respaldos realizados en el paso anterior.

➤ **Etapas del Método de Conversión**

Se nombran las etapas de los métodos de conversión de manera detallada.

➤ **Instalación del Sistema.**

Se explican los pasos para realizar la instalación de la plataforma tanto para el back-end como el front-end.

➤ **Verificación de Instalación del Sistema.**

Se verifica que todos los componentes y ambientes se hayan instalado de manera correcta y el sistema se encuentre en funcionamiento.

❖ **Preparación de entrega 3: "Etapas de Desarrollo e implementación".**

❖ **Entrega 3.**

❖ **Realización del Póster.**

➤ **Diseñar Póster.**

Se diseña el póster del sistema para su presentación, en donde se resume de forma accesible los procesos y resultados de una investigación del proyecto.

➤ **Preparar entrega 1 póster.**

Se revisa el póster realizado, y se corrobora que todo es de acuerdo con lo especificado.

➤ **Entregar Póster (Entrega 1).**

➤ **Corregir Póster.**

Se realizan las correcciones pertinentes, de acuerdo con las observaciones realizadas por el *Product Owner* en la "Entrega 1".

➤ **Preparar entrega 2 póster.**

Se revisa el póster realizado, y se verifica que cumpla con las observaciones realizadas en los pasos anteriores.

➤ **Entregar Póster (Entrega 2).**

❖ **Capacitación.**

➤ **Realizar manual de usuario.**

Se desarrolla documentación de comunicación técnica destinado a dar asistencia a las personas que utilizan el sistema *Omics*. La guía para el usuario tiene contenido escrito, e imágenes asociadas para una mejor comprensión. El lenguaje utilizado es sencillo para brindar a los usuarios un lenguaje que no sea demasiado técnico.

➤ **Realizar tutoriales.**

Se desarrollan tutoriales con los fundamentos principales para poder utilizar el sistema, para orientar a los usuarios que realizan consultas.

➤ **Diseñar plan de capacitación.**

Se elabora el contenido del plan, actividades, cursos, talleres, y conferencias teniendo en cuenta varios aspectos fundamentales:

Los objetivos de la capacitación, es decir los resultados a los que se quiere llegar con el plan de capacitación.

Los objetivos finales, los cuales nos indican la conducta de los integrantes del equipo al finalizar la capacitación.

Los objetivos específicos, que se logran conforme avanzan el desarrollo del plan. Para que se tenga un aprendizaje óptimo, los participantes reconocen la necesidad del conocimiento, y de habilidades nuevas.

➤ **Entrega de material en la plataforma para usuarios finales.**

Se sube el material realizado para los usuarios finales en la plataforma para que puedan tener una guía de las funcionalidades de la misma y puedan provecharla al máximo.

➤ **Capacitar usuarios internos de la organización.**

Para poder proporcionarles las habilidades necesarias para realizar su trabajo y utilizar todas las funcionalidades que ofrece la plataforma. Esto favorece la confianza y desarrollo personal. Mejora las habilidades de comunicación y manejo de conflictos.

- Capacitación de Administradores del Sistema.
- Capacitación de Gestor de Contenido.
- Capacitación de Usuario (Autor y Lector).

➤ **Evaluación de las capacitaciones.**

Para valorar la efectividad y la eficiencia de los esfuerzos de la capacitación.

❖ **Revisión de módulos del sistema.**

Para controlar si cumple con las funcionalidades establecidas o si hay que realizar alguna modificación.

- Módulo de Registro y Actividad de Usuario.
- Módulo de Catálogo y Búsqueda.
- Módulo de Visualización y Seguimiento.
- Módulo de Reseñas.
- Módulo de Suscripción y Donación.
- Módulo de Publicación de Obras.
- Módulo de Notificaciones.
- Módulo de Contrato.
- Módulo de Reportes.
- Módulo de Configuración.

❖ **Correcciones generales del sistema.**

Se realizan todas las correcciones que sean necesarias del sistema para mantener la calidad que se ha propuesto desde un principio y que los usuarios queden satisfechos con lo que ofrece Omics.

❖ **14º Exposición Anual de Proyectos de Sistemas.**

Un proyecto consta de muchos procesos y actividades, las cuales intervienen en las decisiones que se toman para llevarlo a cabo. Desde el principio es importante identificar la manera correcta en que hay que organizar cada una de las actividades involucradas y su nivel de importancia.



Una correcta planificación ayuda a establecer la prioridad de cada actividad y a tener un mejor control del tiempo para ejecutar el proyecto con la calidad deseada y con éxito. Es la clave para llevar adelante las distintas fases del proyecto.

Se consideran varias características para ir desarrollando cada etapa, pero entre ellas hay tres muy significativas: debe ser sistemática, flexible y disciplinada. Se logra la comprensión adecuada de las tareas antes de realizar el trabajo, éstas pueden ser programadas. Esto facilita la adecuada organización al momento de asignar los recursos disponibles de una manera óptima. Si no se logra comprender las tareas, durante el proceso de ejecución surgirán cambios inesperados en la asignación de recursos, programas y por supuesto prioridades del proyecto.

Algo importante es aclarar la fase de desarrollo, la cual es guiada siguiendo metodologías ágiles, lo que implica que no necesariamente sea ejecutada en un período lineal, sin regresiones. En primer lugar, se apoya en el desarrollo de las pruebas unitarias, que guían el progreso de aquellas funcionalidades críticas del sistema, tanto para realizar refactorización como modificaciones.

Lo definido en la etapa de diseño es importante porque conforma la base del sistema desarrollado. Sin embargo, tampoco se considera como una fase estática. Aproximadamente, un gran porcentaje de la misma será establecida y decidida antes de avanzar hasta un punto crítico en el desarrollo e implementación, pero esto no deja de implicar que sea propensa a cambios.

## 2. Diagrama de Tiempos

Se utiliza como herramienta gráfica el diagrama de Gantt que tiene como objetivo exponer el tiempo de dedicación previsto para diferentes tareas o actividades a lo largo de un tiempo total determinado. Con ella y ejecutando las actividades de la misma, se puede completar los resultados del proyecto, y de esta manera, ser capaces de cumplir los objetivos propuestos. [VER ANEXO N.º 1: DIAGRAMA DE GANTT.](#)

A continuación, se muestran las listas de actividades reflejadas en el Diagrama de Gantt:

▼ Omics - Proyecto Final (61%)		16/3/20	10/11/20
▼ Planificación (100%)		16/3/20	31/7/20
<input checked="" type="checkbox"/> 100%	Definir actividades	30/3/20	3/4/20
<input checked="" type="checkbox"/> 100%	Estimar tiempos de actividades	6/4/20	17/4/20
<input checked="" type="checkbox"/> 100%	Definir equipo de trabajo y funciones principales	30/3/20	14/4/20
<input checked="" type="checkbox"/> 100%	Definir herramientas de gestión de proyecto	23/3/20	27/3/20
<input checked="" type="checkbox"/> 100%	Configurar herramientas de gestión de proyecto	30/3/20	14/4/20
<input checked="" type="checkbox"/> 100%	Definir métodos de comunicación formal	16/3/20	26/3/20
<input checked="" type="checkbox"/> 100%	Gestión de configuración del software	16/3/20	31/3/20
<input checked="" type="checkbox"/> 100%	Realizar análisis de factibilidad	1/5/20	26/6/20
<input checked="" type="checkbox"/> 100%	Realizar análisis de riesgo	19/6/20	30/6/20
▼ Tratamiento de medidas preventivas (100%)		1/7/20	31/7/20
<input checked="" type="checkbox"/> 100%	Entrenar a los trabajadores en procedimientos para detectar req...	1/7/20	31/7/20
<input checked="" type="checkbox"/> 100%	Organizar y llevar a cabo capacitaciones referidas a las tecnolog...	1/7/20	31/7/20
<input checked="" type="checkbox"/> 100%	Se realizará un estudio detallado e investigación para conocer ...	1/7/20	31/7/20
<input checked="" type="checkbox"/> 100%	Se llevará un control coordinado del desarrollo de funciones, re...	1/7/20	31/7/20
<input checked="" type="checkbox"/> 100%	Se da prioridad a la tarea que genera el retraso, con asistencia ...	1/7/20	31/7/20
<input checked="" type="checkbox"/> 100%	Generar una comunidad que respalde el proyecto y ayude a dar...	1/7/20	31/7/20

Figura 2. 2 Lista de Planificación.

▼ Definición de Requerimientos (100%)		16/3/20	21/4/20
<input checked="" type="checkbox"/> 100%	Investigar aplicaciones similares	16/3/20	20/3/20
<input checked="" type="checkbox"/> 100%	Relevar y analizar en general sistemas similares	23/3/20	15/4/20
<input checked="" type="checkbox"/> 100%	Detectar funciones e interfaces en sistemas similares	23/3/20	15/4/20
<input checked="" type="checkbox"/> 100%	Detectar tecnologías en sistemas similares	23/3/20	15/4/20
<input checked="" type="checkbox"/> 100%	Elaborar modelo lógico de sistemas relevados	6/4/20	21/4/20
<input checked="" type="checkbox"/> 100%	Elaborar comparación de funcionalidades de sistemas similares	6/4/20	21/4/20
<input checked="" type="checkbox"/> 100%	Justificar proyecto a partir de necesidades encontradas	30/3/20	19/4/20
<input checked="" type="checkbox"/> 100%	Definir objetivos y alcances preliminares del nuevo sistema	23/3/20	3/4/20
<input checked="" type="checkbox"/> 100%	Preparación entrega 1: "Etapa de Definición de Requerimientos"	4/5/20	4/5/20
	Entrega 1  1		5/5/20

Figura 2. 3 Lista de Definición de Requerimientos.

▼ Diseño (100%)		13/4/20	21/5/20
<input checked="" type="checkbox"/> 100%	Definir objetivos y alcances definitivos del sistema	13/4/20	21/4/20
<input checked="" type="checkbox"/> 100%	Definir módulos del sistema	13/4/20	21/4/20
<input checked="" type="checkbox"/> 100%	Definir salida del sistema	27/4/20	21/5/20
<input checked="" type="checkbox"/> 100%	Definir interfaces de usuario	27/4/20	20/5/20
<input checked="" type="checkbox"/> 100%	Definir métodos de comunicación entre partes del sistema	27/4/20	20/5/20
<input checked="" type="checkbox"/> 100%	Definir historias de usuario	27/4/20	20/5/20
<input checked="" type="checkbox"/> 100%	Definir Sprint	27/4/20	5/5/20
<input checked="" type="checkbox"/> 100%	Definir modelos de datos	13/4/20	13/5/20
<input checked="" type="checkbox"/> 100%	Preparación entrega 2: "Etapa de Diseño"	8/6/20	8/6/20
	Entrega Trabajo Práctico Integrador N° 1		19/5/20
	Entrega 2		9/6/20
<input checked="" type="checkbox"/> 100%	Correcciones generales de las etapas anteriores	10/6/20	10/7/20
<input checked="" type="checkbox"/> 100%	Trazabilidad general de los elementos del sistema	13/7/20	2/8/20

Figura 2. 4 Lista de Diseño.

▾ Desarrollo (100%)	7/4/20	25/10/20
<input checked="" type="checkbox"/> 100% Definir tecnologías de desarrollo	7/4/20	14/4/20
<input checked="" type="checkbox"/> 100% Capacitarse en las tecnologías elegidas	20/4/20	12/5/20
<input checked="" type="checkbox"/> 100% Planificar implementación de módulos del sistema	27/4/20	5/5/20
▾ Planificar pruebas del sistema (100%)	3/8/20	24/8/20
<input checked="" type="checkbox"/> 100% Planificar pruebas de Validación de Ingresos de Datos	3/8/20	5/8/20
<input checked="" type="checkbox"/> 100% Planificar pruebas de Lógica de los Módulos	6/8/20	8/8/20
<input checked="" type="checkbox"/> 100% Planificar pruebas de integración entre los módulos del sistema	9/8/20	14/8/20
<input checked="" type="checkbox"/> 100% Planificar pruebas de carga	15/8/20	19/8/20
<input checked="" type="checkbox"/> 100% Planificar pruebas de seguridad por niveles de usuario	20/8/20	24/8/20
▾ Programar módulos del sistema (100%)	21/7/20	14/10/20
▾ Módulos del sistema back-end (100%)	21/7/20	28/9/20
<input checked="" type="checkbox"/> 100% Módulo de Registro y Actividad de Usuario	21/7/20	5/8/20
<input checked="" type="checkbox"/> 100% Módulo de Catálogo y Búsqueda	10/8/20	28/8/20
<input checked="" type="checkbox"/> 100% Módulo de Visualización y Seguimiento	5/8/20	24/8/20
<input checked="" type="checkbox"/> 100% Módulo de Reseñas	5/8/20	10/8/20
<input checked="" type="checkbox"/> 100% Módulo de Suscripción y Donación	25/8/20	14/9/20

Figura 2. 5 Lista de Desarrollo (a).

<input checked="" type="checkbox"/> 100%	Módulo de Publicación de Obras	21/7/20	5/8/20
<input checked="" type="checkbox"/> 100%	Módulo de Notificaciones	15/9/20	24/9/20
<input checked="" type="checkbox"/> 100%	Módulo de Contrato	25/8/20	11/9/20
<input checked="" type="checkbox"/> 100%	Módulo de Reportes	14/9/20	22/9/20
<input checked="" type="checkbox"/> 100%	Módulo de Configuración	15/9/20	28/9/20
▼ Módulos del sistema front-end (100%)		24/8/20	5/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Registro y Actividad de Usuario	24/8/20	8/9/20
<input checked="" type="checkbox"/> 100%	Módulo de Catálogo y Búsqueda	1/9/20	19/9/20
<input checked="" type="checkbox"/> 100%	Módulo de Visualización y Seguimiento	14/9/20	26/9/20
<input checked="" type="checkbox"/> 100%	Módulo de Reseñas	7/9/20	12/9/20
<input checked="" type="checkbox"/> 100%	Módulo de Suscripción y Donación	14/9/20	24/9/20
<input checked="" type="checkbox"/> 100%	Módulo de Publicación de Obras	7/9/20	22/9/20
<input checked="" type="checkbox"/> 100%	Módulo de Notificaciones	25/9/20	5/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Contrato	12/9/20	30/9/20
<input checked="" type="checkbox"/> 100%	Módulo de Reportes	21/9/20	5/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Configuración	21/9/20	4/10/20
<input checked="" type="checkbox"/> 100%	Documentación Módulos del Sistema	12/10/20	14/10/20

Figura 2. 6 Lista de Desarrollo (b).

▼ Revisión de módulos del sistema (100%)		8/9/20	9/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Registro y Actividad de Usuario	8/9/20	9/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Catálogo y Búsqueda	19/9/20	9/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Visualización y Seguimiento	26/9/20	9/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Reseñas	12/9/20	9/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Suscripción y Donación	24/9/20	9/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Publicación de Obras	22/9/20	9/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Notificaciones	5/10/20	9/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Contrato	30/9/20	9/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Reportes	5/10/20	9/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Configuración	4/10/20	9/10/20

Figura 2. 7 Lista de Desarrollo – Revisión de módulos del sistema (c).

▼ Ejecución de pruebas unitarias (100%)		8/9/20	6/10/20
<input checked="" type="checkbox"/> 100%	Ejecutar pruebas unitarias Módulo de Registro y Actividad de Us...	8/9/20	9/9/20
<input checked="" type="checkbox"/> 100%	Ejecutar pruebas unitarias Módulo de Catálogo y Búsqueda	19/9/20	20/9/20
<input checked="" type="checkbox"/> 100%	Ejecutar pruebas unitarias Módulo de Visualización y Seguimiento	26/9/20	27/9/20
<input checked="" type="checkbox"/> 100%	Ejecutar pruebas unitarias Módulo de Reseñas	12/9/20	13/9/20
<input checked="" type="checkbox"/> 100%	Ejecutar pruebas unitarias Módulo de Suscripción y Donación	24/9/20	25/9/20
<input checked="" type="checkbox"/> 100%	Ejecutar pruebas unitarias Módulo de Publicación de Obras	22/9/20	23/9/20
<input checked="" type="checkbox"/> 100%	Ejecutar pruebas unitarias Módulo de Notificaciones	5/10/20	6/10/20
<input checked="" type="checkbox"/> 100%	Ejecutar pruebas unitarias Módulo de Contrato	30/9/20	1/10/20
<input checked="" type="checkbox"/> 100%	Ejecutar pruebas unitarias Módulo de Reportes	5/10/20	6/10/20
<input checked="" type="checkbox"/> 100%	Ejecutar pruebas unitarias Módulo de Configuración	4/10/20	5/10/20
<input checked="" type="checkbox"/> 100%	Resultados de Pruebas Unitarias	6/10/20	6/10/20
<input checked="" type="checkbox"/> 100%	Revisión completa de Pruebas unitarias	12/10/20	13/10/20

Figura 2. 8 Lista de Desarrollo – Ejecución de pruebas unitarias (d).

▼ Ejecución de pruebas (100%)		28/8/20	11/10/20
▼ Ejecución pruebas de Validación de Ingresos de Datos (100%)		5/9/20	10/9/20
✓ 100%	Usuario crea cuenta en la plataforma 01	5/9/20	6/9/20
✓ 100%	Usuario crea cuenta en la plataforma 02	7/9/20	8/9/20
✓ 100%	Usuario quiere registrarse en la plataforma 03	9/9/20	10/9/20
▼ Ejecución pruebas de Lógica de los Módulos Principales (100%)		17/9/20	6/10/20
✓ 100%	Usuario paga suscripción 01	19/9/20	20/9/20
✓ 100%	Autor adjunta la portada al publicar una obra 02	17/9/20	18/9/20
✓ 100%	Administrar aprueba publicación 03	5/10/20	6/10/20
▼ Ejecución pruebas de integración entre los módulos del sistema (100%)		14/9/20	19/9/20
✓ 100%	Colecciones y Categorías existentes 01	14/9/20	15/9/20
✓ 100%	Ver estadística de publicación al dar like 02	16/9/20	17/9/20
✓ 100%	Publicar y visualizar el contenido de la obra 03	18/9/20	19/9/20
▼ Ejecución pruebas de carga (100%)		23/9/20	6/10/20
✓ 100%	Peticiones por segundo de la API REST al endpoint de usuari...	23/9/20	24/9/20
✓ 100%	Peticiones por segundo de la API REST al endpoint de catálo...	28/9/20	29/9/20
✓ 100%	Peticiones por segundo de la API REST al endpoint de catálo...	30/9/20	1/10/20
✓ 100%	Peticiones por segundo de la API REST al endpoint de catálo...	5/10/20	6/10/20

Figura 2. 9 Lista de Desarrollo – Ejecución de pruebas (e).

▾ Ejecución pruebas de seguridad por niveles de usuario (100%)		28/8/20	11/10/20
<input checked="" type="checkbox"/> 100%	Ingreso de Usuario Administrador 01	28/8/20	29/8/20
<input checked="" type="checkbox"/> 100%	Ingresar a una ruta sin permisos 02	8/9/20	9/9/20
<input checked="" type="checkbox"/> 100%	Autor puede visualizar reportes y detalles de sus publicacion...	10/10/20	11/10/20
<input checked="" type="checkbox"/> 100%	Resultados de Ejecución pruebas de Validación de Ingresos de Dat...	10/9/20	10/9/20
<input checked="" type="checkbox"/> 100%	Resultados de Ejecución pruebas de Lógica de los Módulos Princip...	6/10/20	6/10/20
<input checked="" type="checkbox"/> 100%	Resultados de Ejecución pruebas de integración entre los módulos ...	19/9/20	19/9/20
<input checked="" type="checkbox"/> 100%	Resultados de Ejecución pruebas de carga	7/10/20	7/10/20
<input checked="" type="checkbox"/> 100%	Resultados de Ejecución pruebas de seguridad por niveles de usua...	12/10/20	12/10/20
<input checked="" type="checkbox"/> 100%	Preparar demo 1 del sistema	18/9/20	21/9/20
<input type="checkbox"/>	Entrega Trabajo Práctico Integrador N° 2		15/9/20
<input type="checkbox"/>	Entregar demo 1		22/9/20
<input checked="" type="checkbox"/> 100%	Preparar demo 2 del sistema	9/10/20	12/10/20
<input type="checkbox"/>	Entregar demo 2		13/10/20

Figura 2. 10 Lista de Desarrollo – Ejecución de pruebas y Resultados (f).

▾ Revisión de módulos del sistema (100%)		17/5/20	25/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Registro y Actividad de Usuario	14/10/20	15/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Catálogo y Búsqueda	15/10/20	16/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Visualización y Seguimiento	16/10/20	17/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Reseñas	17/5/20	18/5/20
<input checked="" type="checkbox"/> 100%	Módulo de Suscripción y Donación	18/10/20	19/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Publicación de Obras	19/10/20	21/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Notificaciones	21/10/20	22/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Contrato	22/10/20	23/10/20
<input checked="" type="checkbox"/> 100%	Módulo de Reportes	23/10/20	Yesterday
<input checked="" type="checkbox"/> 100%	Módulo de Configuración	Yesterday	Today

Figura 2. 11 Lista de Desarrollo – Revisión de módulos del sistema (g).



▼ Implementación (100%)		5/10/20	25/10/20
<input checked="" type="checkbox"/> 100%	Planificación de las tareas	5/10/20	7/10/20
<input checked="" type="checkbox"/> 100%	Asignar los responsables	7/10/20	8/10/20
<input checked="" type="checkbox"/> 100%	Especificaciones de Servidores	8/10/20	11/10/20
<input checked="" type="checkbox"/> 100%	Equipos de Conectividad	11/10/20	12/10/20
<input checked="" type="checkbox"/> 100%	Comunicaciones	8/10/20	9/10/20
<input checked="" type="checkbox"/> 100%	Puestos de Trabajo	7/10/20	9/10/20
<input checked="" type="checkbox"/> 100%	Infraestructura	12/10/20	13/10/20
<input checked="" type="checkbox"/> 100%	Carga Inicial de Datos	13/10/20	16/10/20
<input checked="" type="checkbox"/> 100%	Migración	16/10/20	19/10/20
<input checked="" type="checkbox"/> 100%	Métodos de Replicación	19/10/20	21/10/20
<input checked="" type="checkbox"/> 100%	Backups	20/10/20	21/10/20
<input checked="" type="checkbox"/> 100%	Recuperación	21/10/20	22/10/20
<input checked="" type="checkbox"/> 100%	Etapas del Método de Conversión	22/10/20	24/10/20
<input checked="" type="checkbox"/> 100%	Instalación del Sistema	19/10/20	21/10/20
<input checked="" type="checkbox"/> 100%	Verificación de Instalación del Sistema	23/10/20	Yesterday
<input checked="" type="checkbox"/> 0%	Preparación de entrega 3: "Etapa de Desarrollo e Implementación"	Today	Today
<input type="checkbox"/> 0%	Entrega 3		Tomorrow

Figura 2. 12 Lista de Implementación.

▼ Realización del Póster (100%)		23/9/20	13/10/20
<input checked="" type="checkbox"/> 100%	Diseñar Póster	23/9/20	25/9/20
<input checked="" type="checkbox"/> 100%	Preparar entrega 1 póster	28/9/20	28/9/20
<input type="checkbox"/> 0%	Entregar Póster (Entrega 1)		29/9/20
<input checked="" type="checkbox"/> 100%	Corregir Póster	7/10/20	11/10/20
<input checked="" type="checkbox"/> 100%	Preparar entrega 2 póster	12/10/20	13/10/20
<input type="checkbox"/> 0%	Entregar Póster (Entrega 2)		13/10/20

Figura 2. 13 Lista Realización del Póster.

<div style="display: flex; justify-content: space-between;"> <span>▼ Capacitación (100%)</span> <span>21/9/20</span> <span>26/10/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 100%</span> <span>Realizar manual de usuario</span> <span>30/9/20</span> <span>15/10/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 100%</span> <span>Realizar tutoriales</span> <span>21/9/20</span> <span>1/10/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 100%</span> <span>Diseñar plan de capacitación</span> <span>28/9/20</span> <span>9/10/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 100%</span> <span>Entrega de material en la plataforma para usuarios finales</span> <span>17/10/20</span> <span>19/10/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>▼ Capacitar usuarios internos de la organización (100%)</span> <span>19/10/20</span> <span>21/10/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 100%</span> <span>Capacitación de Administradores del Sistema</span> <span>19/10/20</span> <span>19/10/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 100%</span> <span>Capacitación de Gestor de Contenido</span> <span>20/10/20</span> <span>20/10/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 100%</span> <span>Capacitación de Usuario (Autor y Lector)</span> <span>21/10/20</span> <span>21/10/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 100%</span> <span>Evaluación de las capacitaciones</span> <span>22/10/20</span> <span>Tomorrow</span> </div>		

Figura 2. 14 Lista de Capacitación.

<div style="display: flex; justify-content: space-between;"> <span>▼ Revisión de Módulos del Sistema (0%)</span> <span>28/10/20</span> <span>8/11/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 0%</span> <span>Módulo de Registro y Actividad de Usuario</span> <span>Wednesday</span> <span>Thursday</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 0%</span> <span>Módulo de Catálogo y Búsqueda</span> <span>Thursday</span> <span>Friday</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 0%</span> <span>Módulo de Visualización y Seguimiento</span> <span>Friday</span> <span>Saturday</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 0%</span> <span>Módulo de Reseñas</span> <span>Saturday</span> <span>1/11/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 0%</span> <span>Módulo de Suscripción y Donación</span> <span>1/11/20</span> <span>3/11/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 0%</span> <span>Módulo de Publicación de Obras</span> <span>3/11/20</span> <span>4/11/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 0%</span> <span>Módulo de Notificaciones</span> <span>4/11/20</span> <span>5/11/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 0%</span> <span>Módulo de Contrato</span> <span>5/11/20</span> <span>6/11/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 0%</span> <span>Módulo de Reportes</span> <span>6/11/20</span> <span>7/11/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 0%</span> <span>Módulo de Configuración</span> <span>7/11/20</span> <span>8/11/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>✓ 0%</span> <span>Correcciones generales del sistema</span> <span>9/11/20</span> <span>9/11/20</span> </div>		
<div style="display: flex; justify-content: space-between;"> <span>📅 14° Exposición Anual de Proyectos de Sistemas</span> <span></span> <span></span> <span>10/11/20</span> </div>		

Figura 2. 15 Entregables y revisiones finales.

### 3. Definición de Sprint

Se usan los Sprint para determinar cada uno de los ciclos o iteraciones que se obtienen dentro de este proyecto Scrum. Permite tener un ritmo de trabajo con un tiempo prefijado. En cada Sprint se consigue un entregable o incremento del proyecto que aporte valor al cliente.

Antes de empezar, se comienza con la reunión de planificación del Sprint. Allí se decide lo que se va a hacer, cómo se va a concretar y el número de tareas. Para este proyecto se decide realizar Sprint cada 2 semanas, obteniendo 18 Sprint en total desde el inicio del Proyecto hasta finalizarlo. Se contempla el inicio del primer Sprint en la fecha 16/03/2020 hasta el último que termina el 22/11/2020.

<b>Número de Sprint</b>	<b>Fecha</b>	<b>Tareas</b>
1	16/03/2020 al 29/03/2020	<p><b>Planificación</b></p> <ul style="list-style-type: none"> <li>• Definir herramientas de gestión de proyecto.</li> <li>• Definir métodos de comunicación formal.</li> <li>• Gestión de configuración de software.</li> </ul> <p><b>Definición de Requerimientos</b></p> <ul style="list-style-type: none"> <li>• Investigar aplicaciones similares.</li> <li>• Relevar y analizar en general sistemas similares.</li> <li>• Detectar funciones e interfaces en sistemas similares.</li> <li>• Detectar tecnologías en sistemas similares.</li> <li>• Definir objetivos y alcances preliminares del nuevo sistema.</li> </ul>
2	30/03/2020 al 12/04/2020	<p><b>Planificación</b></p> <ul style="list-style-type: none"> <li>• Definir actividades.</li> <li>• Estimar tiempos de actividades.</li> <li>• Definir equipo de trabajo y funciones principales.</li> <li>• Configurar herramientas de gestión de proyecto.</li> <li>• Gestión de configuración del software.</li> </ul> <p><b>Definición de Requerimientos</b></p> <ul style="list-style-type: none"> <li>• Relevar y analizar en general sistemas similares.</li> <li>• Detectar funciones e interfaces en sistemas similares.</li> </ul>

		<ul style="list-style-type: none"> <li>• Detectar tecnologías en sistemas similares.</li> <li>• Elaborar modelo lógico de sistemas relevados.</li> <li>• Elaborar comparación de funcionalidades de sistemas similares.</li> <li>• Justificar proyecto a partir de necesidades encontradas.</li> <li>• Definir objetivos y alcances preliminares del nuevo sistema.</li> </ul> <p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>• Definir tecnologías de desarrollo.</li> </ul>
3	13/04/2020 al 26/04/2020	<p><b>Planificación</b></p> <ul style="list-style-type: none"> <li>• Estimar tiempos de actividades.</li> <li>• Definir equipo de trabajo y funciones principales.</li> <li>• Configurar herramientas de gestión de proyecto.</li> </ul> <p><b>Definición de Requerimientos</b></p> <ul style="list-style-type: none"> <li>• Relevar y analizar en general sistemas similares.</li> <li>• Detectar funciones e interfaces en sistemas similares.</li> <li>• Detectar tecnologías en sistemas similares.</li> <li>• Elaborar modelo lógico de sistemas relevados.</li> <li>• Elaborar comparación de funcionalidades de sistemas similares.</li> <li>• Justificar proyecto a partir de necesidades encontradas.</li> </ul> <p><b>Diseño</b></p> <ul style="list-style-type: none"> <li>• Definir objetivos y alcances definitivos del sistema.</li> <li>• Definir módulos del sistema.</li> <li>• Definir modelo de datos.</li> </ul> <p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>• Definir tecnologías de desarrollo.</li> <li>• Capacitarse en las tecnologías elegidas.</li> </ul>
4	27/04/2020 al 10/05/2020	<p><b>Planificación</b></p> <ul style="list-style-type: none"> <li>• Realizar análisis de factibilidad.</li> </ul> <p><b>Definición de Requerimientos</b></p> <ul style="list-style-type: none"> <li>• Preparación entrega 1: "Etapa Definición de Requerimientos".</li> </ul> <p><b>Diseño</b></p> <ul style="list-style-type: none"> <li>• Definir salida del sistema.</li> <li>• Definir interfaces de usuario.</li> </ul>

		<ul style="list-style-type: none"> <li>Definir métodos de comunicación entre partes del sistema.</li> <li>Definir historias de usuario.</li> <li>Definir Sprint.</li> <li>Definir modelo de datos.</li> </ul> <p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>Capacitarse en las tecnologías elegidas.</li> <li>Planificar implementación de módulos del sistema.</li> </ul>
5	11/05/2020 al 24/05/2020	<p><b>Planificación</b></p> <ul style="list-style-type: none"> <li>Realizar análisis de factibilidad.</li> </ul> <p><b>Diseño</b></p> <ul style="list-style-type: none"> <li>Definir salida del sistema.</li> <li>Definir interfaces de usuario.</li> <li>Definir métodos de comunicación entre partes del sistema.</li> <li>Definir historias de usuario.</li> <li>Definir modelo de datos.</li> </ul> <p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>Capacitarse en las tecnologías elegidas.</li> <li>Revisión de módulos del sistema.</li> </ul> <p><b>Entrega Trabajo Práctico Integrador N.º 1</b></p>
6	25/05/2020 al 7/06/2020	<p><b>Planificación</b></p> <ul style="list-style-type: none"> <li>Realizar análisis de factibilidad.</li> </ul> <p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>Revisión de módulos del sistema.</li> </ul>
7	8/06/2020 al 21/06/2020	<p><b>Planificación</b></p> <ul style="list-style-type: none"> <li>Realizar análisis de factibilidad.</li> <li>Realizar análisis de riesgo.</li> </ul> <p><b>Diseño</b></p> <ul style="list-style-type: none"> <li>Preparación entrega 2: "Etapa de Diseño".</li> <li>Correcciones generales de las etapas anteriores.</li> </ul> <p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>Revisión de módulos del sistema.</li> </ul>
8	22/06/2020 al 05/07/2020	<p><b>Planificación</b></p> <ul style="list-style-type: none"> <li>Realizar análisis de factibilidad.</li> <li>Realizar análisis de riesgo.</li> <li>Tratamiento de medidas preventivas.</li> </ul> <p><b>Diseño</b></p> <ul style="list-style-type: none"> <li>Correcciones generales de las etapas anteriores.</li> </ul> <p><b>Desarrollo</b></p>

		<ul style="list-style-type: none"> <li>• Revisión de módulos del sistema.</li> </ul>
9	06/07/2020 al 19/07/2020	<p><b>Planificación</b></p> <ul style="list-style-type: none"> <li>• Tratamiento de medidas preventivas.</li> </ul> <p><b>Diseño</b></p> <ul style="list-style-type: none"> <li>• Correcciones generales de las etapas anteriores.</li> <li>• Trazabilidad general de los elementos del sistema.</li> </ul> <p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>• Revisión de módulos del sistema.</li> </ul>
10	20/07/2020 al 02/08/2020	<p><b>Planificación</b></p> <ul style="list-style-type: none"> <li>• Tratamiento de medidas preventivas.</li> </ul> <p><b>Diseño</b></p> <ul style="list-style-type: none"> <li>• Trazabilidad general de los elementos del sistema.</li> </ul> <p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>• Programar módulos del sistema.</li> <li>• Revisión de módulos del sistema.</li> </ul>
11	03/08/2020 al 16/08/2020	<p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>• Planificar pruebas del sistema.</li> <li>• Programar módulos del sistema.</li> <li>• Revisión de módulos del sistema.</li> </ul>
12	17/08/2020 al 30/08/2020	<p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>• Planificar pruebas del sistema.</li> <li>• Programar módulos del sistema.</li> <li>• Ejecución de pruebas.</li> <li>• Revisión de módulos del sistema.</li> </ul>
13	31/08/2020 al 13/09/2020	<p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>• Programar módulos del sistema.</li> <li>• Revisión de módulos del sistema.</li> <li>• Ejecución de pruebas unitarias.</li> <li>• Ejecución de pruebas.</li> <li>• Resultados de Ejecución pruebas de Validación de Ingresos de Datos.</li> <li>• Revisión de módulos del sistema.</li> </ul>
14	14/09/2020 al 27/09/2020	<p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>• Programar módulos del sistema.</li> <li>• Revisión de módulos del sistema.</li> <li>• Ejecución de pruebas unitarias.</li> <li>• Ejecución de pruebas.</li> </ul>

		<ul style="list-style-type: none"> <li>• Resultados de Ejecución pruebas de integración entre los módulos del sistema.</li> <li>• Preparar demo 1 del sistema.</li> <li>• Entregar demo 1.</li> <li>• Revisión de módulos del sistema.</li> </ul> <p><b>Capacitación</b></p> <ul style="list-style-type: none"> <li>• Realizar tutoriales.</li> </ul> <p><b>Entrega Trabajo Práctico Integrador N.º 2</b></p>
15	28/09/2020 al 11/10/2020	<p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>• Programar módulos del sistema.</li> <li>• Revisión de módulos del sistema.</li> <li>• Ejecución de pruebas unitarias.</li> <li>• Resultados de Pruebas Unitarias.</li> <li>• Ejecución de pruebas.</li> <li>• Resultados de Ejecución pruebas de Lógica de los Módulos Principales.</li> <li>• Resultados de Ejecución pruebas de carga.</li> <li>• Preparar demo 2 del sistema.</li> <li>• Revisión de módulos del sistema.</li> </ul> <p><b>Implementación</b></p> <ul style="list-style-type: none"> <li>• Planificación de las tareas.</li> <li>• Asignar los responsables.</li> <li>• Especificaciones de Servidores.</li> <li>• Equipos de Conectividad.</li> <li>• Comunicaciones.</li> <li>• Puestos de Trabajo.</li> </ul> <p><b>Realización del Póster</b></p> <ul style="list-style-type: none"> <li>• Preparar entrega 1 póster.</li> <li>• Corregir póster.</li> </ul> <p><b>Capacitación</b></p> <ul style="list-style-type: none"> <li>• Realizar manual de usuario.</li> <li>• Realizar tutoriales.</li> <li>• Diseñar plan de capacitación.</li> </ul>
16	12/10/2020 al 25/10/2020	<p><b>Desarrollo</b></p> <ul style="list-style-type: none"> <li>• Programar módulos del sistema.</li> <li>• Revisión completa de Pruebas Unitarias.</li> <li>• Resultados de Ejecución pruebas de seguridad por niveles de usuario.</li> <li>• Preparar demo 2 del sistema.</li> <li>• Entregar demo 2.</li> <li>• Revisión de módulos del sistema.</li> </ul> <p><b>Implementación</b></p>

		<ul style="list-style-type: none"> <li>• Equipos de Conectividad.</li> <li>• Infraestructura.</li> <li>• Carga Inicial de Datos.</li> <li>• Migración.</li> <li>• Métodos de Replicación.</li> <li>• Backups.</li> <li>• Recuperación.</li> <li>• Etapas del Método de Conversión.</li> <li>• Instalación del Sistema.</li> <li>• Verificación de Instalación del Sistema.</li> </ul> <p><b>Realización del Póster</b></p> <ul style="list-style-type: none"> <li>• Preparar entrega 2 póster.</li> </ul> <p><b>Capacitación</b></p> <ul style="list-style-type: none"> <li>• Realizar manual de usuario.</li> <li>• Entrega de material en la plataforma para usuarios finales.</li> <li>• Capacitar usuarios internos de la organización.</li> <li>• Evaluación de las capacitaciones.</li> </ul>
17	26/10/2020 al 08/11/2020	<p><b>Implementación</b></p> <ul style="list-style-type: none"> <li>• Preparación de entrega 3: "Etapa de Desarrollo e Implementación".</li> <li>• Entrega 3.</li> </ul> <p><b>Capacitación</b></p> <ul style="list-style-type: none"> <li>• Evaluación de las capacitaciones.</li> </ul> <p><b>Revisión de Módulos del Sistema</b></p> <p><b>Correcciones generales del sistema</b></p>
18	09/11/2020 al 22/11/2020	<p><b>Correcciones generales del sistema</b></p> <p><b>14°Exposición Anual de Proyectos de Sistemas</b></p>

Tabla 2. 1 División del proyecto en Sprints.



# CAPÍTULO II: Organización para la Ejecución del Proyecto

## 1. Metodología Ágil

La metodología ágil (Fundación Wikipedia, 2020b) surge como sustituto a los métodos clásicos de gestión. La flexibilidad, la calidad y la necesidad de entregar proyectos y productos en cortos espacios de tiempo son una prioridad. Se decide utilizar para el desarrollo del Sistema *Omics* esta metodología porque el enfoque que brinda permite desarrollar el sistema, de tal forma que cada vez que el equipo finaliza un sprint se tiene un conjunto de funcionalidades listas para entregar.

Para realizar la Metodología Ágil se decide utilizar la técnica de Framework Scrum. Es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto.

En Scrum se realizan entregas parciales y regulares del producto final. Además, Scrum está indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos como en el caso de un Sistema de Cómics, del cual no se tiene experiencia previa, ni muchos conocimientos sobre la temática de cómics.

Ventajas de utilizar la Metodología Ágil:

- ❖ Aumento de la productividad del equipo de trabajo.
- ❖ Mejora la motivación e implicación del equipo, porque con el uso de la metodología ágil se permite a todo el equipo conocer el estado del proyecto en todo momento.
- ❖ Las ideas de todos los miembros se tienen en cuenta. Antes de tomar una decisión definitiva todo se negocia y se llega a un acuerdo entre todos los miembros del equipo.
- ❖ Fomenta el trabajo en equipo, y por tanto, aumenta la implicación de los miembros del mismo.
- ❖ Mayor velocidad y eficacia, debido a que la metodología ágil permite hacer entregas parciales del producto, por lo que se puede entregar una versión funcional del mismo en breve intervalo de tiempo.

- ❖ Mejora la calidad del producto. Las entregas parciales del proyecto permiten detectar los fallos o errores que tenga, y aplicar soluciones antes de su presentación definitiva.
- ❖ La satisfacción de los clientes es mayor, debido a que estarán involucrados en cada etapa del proceso viendo la progresión del mismo, aportando sugerencias, modificaciones, mejoras y opiniones.

## 1.1. Técnicas para aplicar la Metodología Ágil:

### 1.1.1. Tablero Kanban

El Tablero Kanban (Fundación Wikipedia, 2020d) es un método sencillo de aplicar y muy útil. Consiste en dividir un tablero en columnas que indican los estados por los que debe pasar un proyecto.

Lo más común es dividir el tablero en cuatro columnas. En la primera columna se puede poner "tareas que hacer" y en distintos post-it indicar las tareas. En la segunda columna se deberán indicar qué tareas están en proceso. En la tercera se incluirán las que están en fase de prueba y en la última columna todas las tareas finalizadas.

Los post-it se mueven de izquierda a derecha en función de cómo progresa el proyecto. Pero no solo es válido ese movimiento. Si se detecta algún error, las tareas también se puede mover hacia atrás hasta que este esté solventado. En la imagen se puede observar un ejemplo de cómo sería un Tablero Kanban.

Este tablero también introduce el concepto WIP (Work In Progress), que hace referencia a un número que se introduce al lado del título de cada tarea y que indica cuántas tareas, máximas o mínimas, pueden desarrollarse de forma simultánea en una misma etapa.



*Figura 2. 16 Ejemplo de un Tablero Kanban*

### **1.1.2. SCRUM**

SCRUM (Palacio, 2015) es un método de organización en las empresas para el desarrollo de proyectos. Es una metodología ágil en la que un grupo de trabajo tiene tareas que ha de desarrollar en un período de tiempo determinado, denominados Sprint o iteraciones. Los Sprint o iteraciones deben entregarse al cliente en un tiempo limitado, normalmente en el máximo de un mes.

Al inicio de esta metodología se redacta una lista de deseos sobre qué funcionalidades debe tener el producto. Ésta se ordena por orden de prioridad, de mayor a menor importancia para el negocio permitiendo que las tareas se realicen de forma más ágil.

En SCRUM se trabaja con ciclos cortos de trabajo, y se caracteriza porque se realizan reuniones de forma periódica con el cliente, el cual está muy involucrado y es conocedor de cómo avanza todo el proceso.

A nivel de organización interno también se hacen diariamente las conocidas como Stand-up Meeting o Daily Scrum que son reuniones breves al inicio de la jornada para hacer una revisión de lo que se hizo el día anterior, qué problemas surgieron y qué se va a hacer a lo largo del día.

#### Proceso de Scrum

En Scrum un proyecto se ejecuta en ciclos temporales cortos y de duración fija (iteraciones que normalmente son de 2 semanas, aunque en algunos equipos son de 3 y hasta 4 semanas, límite máximo de feedback de producto real y reflexión). Cada iteración tiene que proporcionar un resultado completo, un incremento de producto final que sea susceptible de ser entregado con el mínimo esfuerzo al cliente cuando lo solicite.

El proceso parte de la lista de objetivos/requisitos priorizada del producto, que actúa como plan del proyecto. En esta lista el cliente (Product Owner) prioriza los objetivos balanceando el valor que le aportan respecto a su coste (que el equipo estima considerando la Definición de Hecho) y quedan repartidos en iteraciones y entregas.

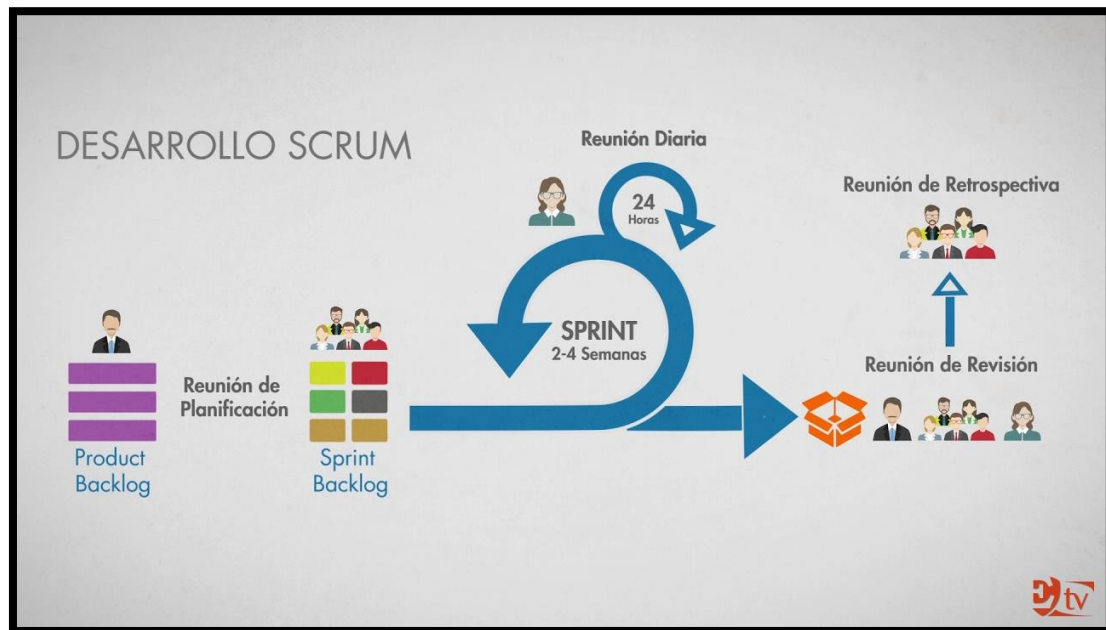


Figura 2. 17 Proceso de Scrum

Las actividades que se llevan a cabo en Scrum son las siguientes (los tiempos indicados son para iteraciones de 2 semanas):

### Planificación de la iteración

El primer día de la iteración se realiza la reunión de planificación de la iteración. Tiene dos partes:

- ❖ *Selección de requisitos* (2 horas). El cliente presenta al equipo la lista de requisitos priorizada del producto o proyecto. El equipo pregunta al cliente las dudas que surgen y selecciona los requisitos más prioritarios que prevé que podrá completar en la iteración, de manera que puedan ser entregados si el cliente lo solicita.
- ❖ *Planificación de la iteración* (2 horas). El equipo elabora la lista de tareas de la iteración necesarias para desarrollar los requisitos seleccionados. La estimación de esfuerzo se hace de manera conjunta y los miembros del equipo se autoasignan las tareas, se autoorganizan para trabajar, incluso en parejas (o grupos mayores), con el fin de compartir conocimiento (creando un equipo más resiliente) o para resolver juntos objetivos especialmente complejos.

### Ejecución de la iteración

Cada día el equipo realiza una reunión de sincronización (15 minutos), normalmente delante de un tablero físico o pizarra ( *Scrum Taskboard* ) – *Proyectos Ágiles*, s/f). El equipo inspecciona el trabajo que el resto está realizando (dependencias entre tareas, progreso hacia el objetivo de la iteración, obstáculos que pueden impedir este

objetivo) para poder hacer las adaptaciones necesarias que permitan cumplir con la previsión de objetivos a mostrar al final de la iteración. En la reunión cada miembro del equipo responde a tres preguntas:

- ❖ ¿Qué he hecho desde la última reunión de sincronización para ayudar al equipo a cumplir su objetivo?
- ❖ ¿Qué voy a hacer a partir de este momento para ayudar al equipo a cumplir su objetivo?
- ❖ ¿Qué impedimentos tengo o voy a tener que nos impiden conseguir nuestro objetivo?

Durante la iteración el Facilitador (Scrum Master) se encarga de que el equipo pueda mantener el foco para cumplir con sus objetivos.

- ❖ Elimina los obstáculos que el equipo no puede resolver por sí mismo.
- ❖ Protege al equipo de interrupciones externas que puedan afectar el objetivo de la iteración o su productividad.

Durante la iteración, el cliente junto con el equipo refinan la lista de requisitos (para prepararlos para las siguientes iteraciones) y, si es necesario, cambian o vuelven a planificar nuevamente los objetivos del proyecto (10%-15% del tiempo de la iteración) con el objetivo de maximizar la utilidad de lo que se desarrolla y el retorno de inversión.

### **Inspección y adaptación**

El último día de la iteración se realiza la reunión de revisión de la iteración. Tiene dos partes:

- ❖ *Revisión* (demostración) (1,5 horas). El equipo presenta al cliente los requisitos completados en la iteración, en forma de incremento de producto preparado para ser entregado con el mínimo esfuerzo. En función de los resultados mostrados y de los cambios que haya habido en el contexto del proyecto, el cliente realiza las adaptaciones necesarias de manera objetiva, ya desde la primera iteración, replanificando el proyecto.
- ❖ *Retrospectiva* (1,5 horas). El equipo analiza cómo ha sido su manera de trabajar y cuáles son los problemas que se pueden impedir para progresar adecuadamente, mejorando de manera continua su productividad. El Facilitador se va a encargar de eliminar o escalar los obstáculos identificados que estén más allá del ámbito de acción del equipo.

### 1.1.3. Diagrama Burndown

El diagrama Burndown, también denominado diagrama de trabajo pendiente permite conocer el tiempo que falta para terminar todas las tareas pendientes en un sprint.

Son fáciles de diseñar, muy visuales para conocer cuánto tiempo falta para acabar y su mantenimiento no es costoso.

Elaboración de un diagrama Burndown:

- ❖ Primero se divide el trabajo en iteraciones o Sprint, indicando la duración de cada una de ellas.
- ❖ A continuación, se realiza una estimación de las tareas a desarrollar en los días ideales.
- ❖ Finalmente, se acuerda qué funcionalidad va a desarrollar cada participante del equipo. Esto dependerá del número de participantes del equipo, y de la duración del sprint.

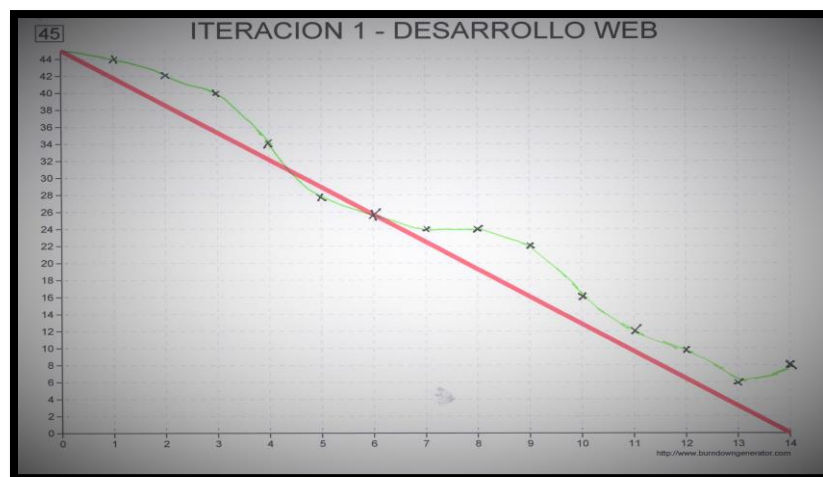


Figura 2. 18 Diagrama de Burndown

#### Los beneficios que nos otorga el utilizar Scrum son:

- ❖ *Gestión regular de las expectativas del cliente (los docentes):* Los docentes comprueban de manera regular si se van cumpliendo los ítems de la Guía anual, da feedback, desde el inicio del Sistema hasta el finalizado, evitando así tener errores al terminar el sistema.
- ❖ *Resultados anticipados ("time to market"):* Los docentes tienen avances del proyecto, sin que esté terminado.

- ❖ *Flexibilidad y adaptación:* Respecto a los cambios solicitados por los docentes, y la necesidad de modificación de los requisitos del sistema.
- ❖ *Mitigación de riesgos:* Desde la primera iteración el equipo tiene que gestionar los problemas que pueden aparecer en una entrega del proyecto. Al hacer patentes estos riesgos, es posible iniciar su mitigación de manera anticipada. "Si hay que equivocarse o fallar, mejor hacerlo lo antes posible". El feedback temprano permite ahorrar esfuerzo y tiempo en errores técnicos.
- ❖ *Productividad y calidad:* De manera regular el equipo va mejorando y simplificando su forma de trabajar. Los miembros del equipo sincronizamos el trabajo diariamente y nos ayudamos a resolver los problemas que pueden impedir que se avance. La comunicación y la adaptación a las diferentes necesidades entre los miembros del equipo son máximas para evitar que se realicen tareas innecesarias o doble trabajo.
- ❖ *Alineación entre el cliente (docentes) y el equipo de desarrollo:* Los resultados del proyecto se miden en forma de objetivos y entregas realizadas a los docentes. El Sistema se enriquece con el aporte de los conocimientos de todos los miembros del equipo.
- ❖ *Equipo motivado:* Los integrantes están más motivados cuando pueden mostrar el trabajo realizado.

## 2. Equipo de Trabajo (estructura, puestos, perfiles, cantidades)

Como equipo de trabajo para este proyecto, se sabe que participan varias personas con diferentes formas de pensar y de actuar pero que existe una meta en común. Se han propuesto algunas reglas o normas de comportamiento que hay que respetar para generar orden al momento de realizar actividades. En forma general, se nombrarán las más importantes:

- ❖ Claridad en los objetivos
- ❖ Claridad en los roles para cada integrante
- ❖ Procedimientos definidos
- ❖ Buena comunicación
- ❖ Confianza y responsabilidad
- ❖ Participación de todos
- ❖ Normas de comportamiento (ya que esto va a ir determinando las consecuencias en el trabajo, que pueden ser positivas o negativas)

Hay que tener en cuenta, que cada miembro del equipo tiene que ir cumpliendo sus objetivos, ya que la suma de los objetivos de cada uno dará como resultado el objetivo final del proyecto. Se tiene que ir analizando cada uno de ellos y si se observa que no se están cumpliendo, es necesario hacer reuniones y ver las causas. Esto es muy importante porque se pueden ver los indicadores y revisiones del trabajo continuo que es esencial para llevar un control del mismo y poder rectificar.

Como equipo se definen algunos procesos para realizar las tareas y se dialoga sobre cómo se toman las decisiones con la utilización de recursos e intercambio de información.



### 3. Funciones Principales de los Miembros del Equipo de Trabajo

Las funciones principales asignadas para cada miembro del equipo de trabajo se detallan a continuación:

<i>Miembros/ Roles</i>	Amin	Boglioli	Haarth	Lemos	Muñoz Velázquez	Profesores
<b>Scrum Master</b>	X					
<b>Product Manager</b>				X		
<b>Product Owner</b>						X
<b>Dev Front-End</b>					X	
<b>Dev Back-End</b>		X				
<b>Tester</b>			X			
<b>Arquitecto de Sw</b>		X				

Tabla 2. 2 Roles y asignaciones de integrantes del equipo.

La asignación de roles se realiza en base a las capacidades individuales de cada miembro del equipo, teniendo en cuenta sus fortalezas y debilidades. Es necesario conocer las tareas que realiza cada uno para tener un orden, y plantear las dudas al responsable a cargo. Tras finalizar las tareas asignadas, se colabora con el compañero para lograr avanzar con el trabajo en equipo.

A continuación, se describen los roles para ampliar las actividades que debe realizar cada integrante del equipo:

- ❖ **Product Manager:** Es el que se encarga de guiar cómo hay que trabajar, los procedimientos de coordinación y de trabajo entre el equipo. Es la persona que

tiene el rol de comunicador, y quien tiene que fluir la comunicación del equipo para, por ejemplo, la resolución de conflictos, la toma de decisiones, organización de las reuniones, abordamiento de los conflictos, entre otros.

- ❖ **Product Owner:** Representa la voz del consumidor final (puede ser interno o externo). Define la visión del producto, y respecto a la elaboración de la estrategia. Colabora con el equipo en la definición de la estrategia de negocio y técnica. El Product Owner escribe historias de usuario, las prioriza, y las coloca en el Product Backlog.
- ❖ **Scrum Master:** El Scrum es facilitado por un Scrum Master, cuyo trabajo primario es eliminar los obstáculos que impiden que el equipo alcance el objetivo del sprint. El Scrum Master no es el líder del equipo (porque ellos se autoorganizan), sino que actúa como una protección entre el equipo y cualquier influencia que le distraiga. El Scrum Master se asegura de que el proceso Scrum se utiliza como es debido. El Scrum Master es el que hace que las reglas se cumplan.
- ❖ **Equipo Scrum (Team):** El equipo tiene la responsabilidad de entregar incrementos del producto al final de cada Sprint. Es recomendable que el equipo tenga habilidades transversales necesarias para realizar el trabajo (análisis, diseño, desarrollo, pruebas, documentación, etc.):
  - **Dev Front-end:** Es el encargado de desarrollar la parte visual, y los elementos necesarios para que el usuario interactúe con el sistema. También es responsable de realizar, definir e implementar el diseño del mismo.
  - **Dev Back-end:** Es el encargado de desarrollar la parte lógica para responder a las solicitudes de base de datos requeridas desde el front-end.
  - **Tester:** Los probadores de software son quienes planifican y llevan a cabo pruebas de software, y realizan las pruebas de software para comprobar si funcionan correctamente. Además, se encargan de identificar el riesgo de sufrir errores de un software, detectan errores, y los comunican al equipo. Evalúan el funcionamiento general del software y sugieren formas de mejorarlo. Pueden evaluar el correcto funcionamiento de la siguiente forma:
    - Si el software coincide con la especificación original, o qué hay que hacer para que coincida.

- Si cuando un usuario hace clic en un botón o enlace para que suceda algo, por ejemplo, para ir a una nueva pantalla, se abre la pantalla correcta.
  - Si el usuario tiene que introducir algunos datos, el software actúa de forma correcta y los datos se almacenan de forma segura.
  - Si está diseñado para que varios usuarios puedan abrirlo al mismo tiempo sin que el programa reduzca su velocidad de procesamiento.
- **Arquitecto de Software:** Es la persona con amplios conocimientos técnicos, gran experiencia en programación y liderazgo, el cual ejerce las siguientes funciones:
- Gestión de los requisitos no funcionales y definición de la Arquitectura de Software.
  - Selección de la Tecnología.
  - Mejora continua de la arquitectura.
  - Facilitador.
  - Líder y formador.
  - Aseguramiento de la calidad.

## 4. Métodos de Comunicación Formal, Control de Avances, Retroalimentación y Decisiones

Tener una buena comunicación es muy importante. Si bien en algunos momentos se pueden enfrentar con algunos problemas, hay que seguir un procedimiento para poder estar coordinados, ya que de lo contrario se pueden provocar retrasos en el equipo. Debe fluir la comunicación, siendo clara y abierta para el correcto desempeño del equipo. Para lo anterior mencionado, se eligieron dos herramientas:

- ❖ **Discord:** Es utilizado para la comunicación formal, ya que la app permite enviar documentos, enlaces, imágenes y videos. Se adecua a nuestra necesidad. También, da la posibilidad de organizar reuniones vídeo llamada, en caso de que sea necesario. Se puede compartir pantalla con el resto de los miembros. Otras de sus características es que se puede dividir la comunicación en canales según el área que se necesite.

➤ [HTTPS://DISCORDAPP.COM/](https://discordapp.com/)



Figura 2. 19 Logo de Discord.

- ❖ **Telegram:** Es empleado para realizar la comunicación informal, además que permite poder usarse sin necesidad que el móvil esté conectado a Internet, lo cual es una ventaja a la hora de tener una comunicación rápida.

➤ [HTTPS://WEB.TELEGRAM.ORG](https://web.telegram.org)

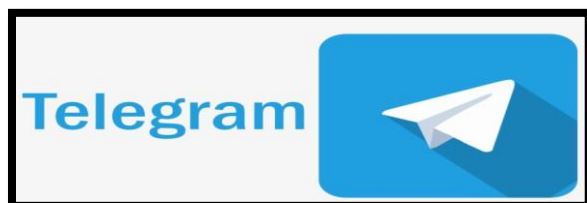


Figura 2. 20 Logo de Telegram.

- ❖ **Trello:** Es la herramienta utilizada para realizar el control de avance del proyecto. Empleando el sistema Kanban para el registro de actividades con tarjetas virtuales, organiza tareas, permite agregar listas, adjuntar archivos, etiquetar eventos, agregar comentarios y compartir tableros. Trello es un tablón virtual en

el que se pueden colgar ideas, tareas, imágenes o enlaces. Es versátil y fácil de usar pudiendo realizar cualquier tipo de tarea que requiera organizar información.

➤ [HTTPS://TRELLO.COM/ES](https://trello.com/es)



Figura 2. 21 Logo de Trello

❖ **TeamGantt y Microsoft Project:** Es la herramienta utilizada para realizar el control de avance de las actividades. TeamGantt presenta integración con la herramienta Trello, permitiendo generar automáticamente las actividades en la segunda, ofreciendo un desarrollo controlado y ágil de las tareas. Además, el seguimiento de actividades podemos visualizarlo de manera precisa, en gráfico de barra por cada cabecera de grupo y mostrando el porcentaje de realización. TeamGantt ofrece la posibilidad de sincronización y trabajo en simultáneo sobre el mismo proyecto, y Microsoft Project su potente agregado de herramientas para realizar estimaciones en cuanto a recursos, costos y utilización de estos, facilitando, a su vez, la visualización. Sus principales atractivos son que se pueden asignar recursos a tareas, dar seguimiento al progreso, administrar presupuesto y analizar cargas de trabajo.

➤ [HTTPS://WWW.TEAMGANTT.COM](https://www.teamgantt.com)

➤ [HTTPS://WWW.MICROSOFT.COM/ES-AR/MICROSOFT-365/PROJECT/PROJECT-MANAGEMENT-SOFTWARE](https://www.microsoft.com/es-ar/microsoft-365/project/project-management-software)



Figura 2. 22 Logo de TeamGantt y Project de Microsoft.

La retroalimentación es un proceso de comunicación entre el Product Manager y cada uno del equipo de trabajo para poder realizar una revisión del desempeño que se ha estado haciendo, mostrando el cumplimiento de las actividades esperadas. Permite conocer y fomentar las fortalezas de cada uno y establecer pautas para trabajar sobre las debilidades. De esta manera, la retroalimentación (feedback) se convierte en una

herramienta fundamental de apoyo para que se pueda encontrar la motivación y contribuir para que se vayan superando en las actividades que se realizan, mejorar el rendimiento y el impacto que va causando en el proyecto.

Para la toma de decisiones todos deben participar en cada tema, dando opiniones y estableciendo compromisos. Una vez que se elija una idea a realizar, hay que trabajar sobre la misma.

Como equipo se van a ir guiando con una serie de pasos que hay que tener en cuenta:

- ❖ Establecer un objetivo.
- ❖ Identificar alternativas.
- ❖ Evaluar alternativas.
- ❖ Seleccionar la mejor alternativa.
- ❖ Implantar y controlar la decisión.
  - Desarrollar un plan.
  - Comunicar la decisión.
  - Seguimiento de la ejecución.
  - Controlar los progresos.
  - Actuar sobre las desviaciones.

## 5. Gestión de Configuración del Software: Método de gestión del versionado durante todo el proyecto

**Para el versionado se consideran los dos productos principales del proyecto:**

- ❖ Documentación.
- ❖ Implementación.

**La implementación estará compuesta por:**

- ❖ Código fuente del sistema.
- ❖ Test automatizados unitarios, de integración, y demás pruebas.
- ❖ Herramientas generadas.
  - Migraciones de base de datos.
- ❖ Configuración.
- ❖ De entornos: Testing, desarrollo, producción
  - De despliegue y puesta en producción
  - De servicios utilizados: Sistema de gestión de base de datos, servicios externos (APIs).
- ❖ Scripts.

**Para la documentación del proyecto:**

- ❖ En una primera etapa, se utiliza el versionado proporcionado por la herramienta Google Docs. [HTTPS://DOCS.GOOGLE.COM](https://docs.google.com).  
En una segunda etapa, debido a la carga que supone la herramienta de la primera trabajada, se decide utilizar OneDrive para que el manejo de datos y el exportar a PDF sea de fácil mantenimiento y sostenida en el tiempo.
- ❖ Para back-end, se documenta utilizando la herramienta brindada por Postman.
- ❖ Del tiempo de tareas y desarrollo de las mismas, se utiliza las gráficas proporcionadas por HacknPlan.

En cuanto a todo lo correspondiente a documentación del código:

- ❖ Se utiliza *Git* como herramienta de versionado.
- ❖ Junto a la anterior, se usa un servicio de almacenamiento de repositorios como *GitHub*. Esta herramienta maneja tags (*etiquetas*) y Branch (*ramas*). Las etiquetas se usan para definir versiones con el sistema *SEM Ver*. Las ramas permiten mantener múltiples *instantáneas* del código fuente, incluso haciendo referencia a distintas versiones. Así, con el uso de ramas y etiquetas se mantiene el workflow (*flujo de trabajo*) sobre el código.
- ❖ Las etiquetas indican las versiones, las ramas y las etapas de desarrollo:

- **Develop:** Desarrollo
- **Staging:** Versiones listas para testing
- **Release:** Salida a producción
  - Los cambios son confirmados por *commits*. Incluyen mensajes simples e indican qué cambios se realizan en esa *snapshot* (instantánea) en el código fuente.
  - Para los mensajes de los commits se respeta el formato determinado por una convención altamente utilizada: [WWW.CONVENTIONALCOMMITTS.ORG](http://www.conventionalcommits.org).



## CAPÍTULO III: Factibilidad

### 1. Definición y Descripción de los Recursos para cada una de las Actividades

NOMBRE DE LA TAREA	NOMBRE DE LOS RECURSOS
<b><i>Planificación</i></b>	
Definir actividades.	Scrum Master [25%]; Product Manager [25%].
Estimar tiempos de actividades.	Product Manager [25%].
Definir equipo de trabajo y funciones principales.	Product Manager [25%].
Definir herramientas de gestión de proyectos.	Arquitecto de Software [25%].
Configurar herramientas de gestión de proyectos.	Arquitecto de Software [25%].
Definir métodos de comunicación formal.	Product Manager [25%]; Scrum Master [25%].
Gestión de configuración del software.	Arquitecto de Software [25%]
Realizar análisis de factibilidad.	Product Manager [25%]; Scrum Master [25%].
Realizar análisis de riesgo.	Product Manager [25%].
<b><i>Tratamiento de Medidas Preventivas</i></b>	

<p>Entrenar a los trabajadores en procedimientos para detectar requerimientos. Desarrollar análisis de Seguridad de la Tarea y procedimientos seguros de trabajo.</p>	<p>Product Manager [10%].</p>
<p>Organizar y llevar a cabo capacitaciones referidas a las tecnologías que se van a usar en el proyecto. La persona con más habilidad en la tecnología puede capacitar al miembro del equipo que lo necesite. También se puede contar con la compra de cursos webs.</p>	<p>Product Manager [10%].</p>
<p>Realizar un estudio detallado e investigación para conocer mejor el negocio. Hay contactos que pueden brindar asesoramiento más detallado en la industria que se plantea para utilizar en el proyecto.</p>	<p>Product Manager [10%].</p>
<p>Llevar un control coordinado del desarrollo de funciones, replanteo de alguna librería o tecnología a usar y una vez que finalice la función realizada por alguno de los integrantes se pasará a otro del equipo para que lo revise. Realizar refactoring en las partes críticas del sistema regularmente.</p>	<p>Product Manager [10%].</p>
<p>Dar prioridad a la tarea que genera el retraso, con asistencia al miembro que lo necesite para acelerar su finalización o visualizar una solución alternativa que no cause consecuencia en cascada.</p>	<p>Product Manager [10%].</p>

<p>Generar una comunidad que respalde el proyecto y ayude a darse a conocer el mismo. Establecer tiempos a largo plazo coordinando las tareas fundamentales. Considerar mercados potenciales basado en el contenido que se vaya generando en los períodos de prueba en desarrollo.</p>	<p>Product Manager [10%].</p>
<p><b>Definición de Requerimientos</b></p>	
<p>Investigar aplicaciones similares.</p>	<p>Arquitecto de Software [50%]; Scrum Master [50%].</p>
<p>Relevar y analizar sistemas generales en forma global.</p>	<p>Product Manager [25%].</p>
<p>Detectar funciones e interfaces en sistemas similares.</p>	<p>Tester.</p>
<p>Detectar tecnologías en sistemas similares.</p>	<p>Product Manager.</p>
<p>Elaborar modelo lógico de sistemas relevados.</p>	<p>Arquitecto de Software [25%].</p>
<p>Elaborar comparación de funcionalidades de sistemas similares.</p>	<p>Scrum Master [25%]; Product Manager [25%].</p>
<p>Justificar proyecto a partir de necesidades.</p>	<p>Scrum Master [25%].</p>
<p>Definir objetivos y alcances preliminares del nuevo sistema.</p>	<p>Arquitecto de Software [25%]; Scrum Master [25%]; Product Manager [25%].</p>
<p>Preparación de entrega 1: "Etapa de Definición de Requerimientos".</p>	<p>Arquitecto de Software [25%]; Scrum Master [25%]; Product Manager [25%].</p>

<b>Entrega 1</b>	
<b>Diseño</b>	
Definir objetivos y alcances definitivos del nuevo sistema.	Arquitecto de Software [25%].
Definir módulos del sistema.	Dev Front-end [25%]; Dev Back-end [25%].
Definir salida del sistema.	Arquitecto de Software [25%]; Dev Front-end [25%]; Dev Back-end [25%].
Definir interfaces de usuario.	Dev Front-end [25%].
Definir métodos de comunicación entre partes del sistema.	Dev Back-end [25%].
Definir historias de usuario.	Tester [25%].
Definir Sprint.	Scrum Master [25%].
Definir modelo de datos.	Scrum Master [25%].
Preparación entrega 2: "Etapa de Diseño".	Arquitecto de Software [25%]; Scrum Master [25%]; Product Manager [25%]; Dev Front-end [25%]; Dev Back-end [25%].
<b>Entrega Trabajo Práctico Integrador N.º 1</b>	
<b>Entrega 2</b>	
Correcciones generales de las etapas anteriores.	Product Manager [40%]; Scrum Master; Tester; Arquitecto de Software; Dev Back-end; Dev Front-end.
Trazabilidad general de los elementos del	Scrum Master [90%]; Tester [90%]; Arquitecto de Software; Dev Back-

sistema.	end [70%]; Dev Front-end; Product Manager [5%].
<b>Desarrollo</b>	
Definir tecnologías de desarrollo.	Arquitecto de Software [25%].
Capacitarse en las tecnologías elegidas.	Dev Front-end [25%]; Dev Back-end [25%]; Tester [25%].
Planificar implementación de módulos del sistema.	Arquitecto de Software [25%]; Product Manager [10%].
<b>Planificar Pruebas del Sistema</b>	
Planificar pruebas de Validación de Ingresos de Datos.	Product Manager; Scrum Master; Tester.
Planificar pruebas de Lógica de los Módulos Principales.	Product Manager; Scrum Master; Tester.
Planificar pruebas de integración entre los módulos del sistema.	Product Manager; Scrum Master; Tester.
Planificar pruebas de carga.	Scrum Master; Product Manager; Tester.
Planificar pruebas de seguridad por niveles de usuario.	Scrum Master; Product Manager; Tester.
<b>Programar Módulos del Sistema</b>	
<b>Módulos del Sistema Back-end</b>	
Módulo de Registro y Actividad de Usuario.	Dev Back-end [10%].
Módulo de Catálogo y Búsqueda.	Dev Back-end [10%].
Módulo de Visualización y Seguimiento.	Dev Back-end [10%].
Módulo de Reseñas.	Dev Back-end [10%].
Módulo de Suscripción y Donación.	Dev Back-end [10%].
Módulo de Publicación de Obras.	Dev Back-end [10%].
Módulo de Notificaciones.	Dev Back-end [10%].

Módulo de Contrato.	Dev Back-end [10%].
Módulo de Reportes.	Dev Back-end [10%].
Módulo de Configuración.	Dev Back-end [10%]
<b>Módulos del Sistema Front-end</b>	
Módulo de Registro y Actividad de Usuario.	Dev Front-end [10%].
Módulo de Catálogo y Búsqueda.	Dev Front-end [10%].
Módulo de Visualización y Seguimiento.	Dev Front-end [10%].
Módulo de Reseñas.	Dev Front-end [10%].
Módulo de Suscripción y Donación.	Dev Front-end [10%].
Módulo de Publicación de Obras.	Dev Front-end [10%].
Módulo de Notificaciones.	Dev Front-end [10%].
Módulo de Contrato.	Dev Front-end [10%].
Módulo de Reportes.	Dev Front-end [10%].
Módulo de Configuración.	Dev Front-end [10%].
Documentación Módulos del Sistema.	Arquitecto de Software [10%].
<b>Revisión de Módulos del Sistema</b>	
Módulo de Registro y Actividad de Usuario.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Catálogo y Búsqueda.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Visualización y Seguimiento.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Reseñas.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%]
Módulo de Suscripción y Donación.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Publicación de Obras.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].

Módulo de Notificaciones.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Contrato.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Reportes.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Configuración.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
<b>Ejecución de Pruebas Unitarias</b>	
Ejecutar pruebas unitarias Módulo de Registro y Actividad de Usuario.	Tester [1%].
Ejecutar pruebas unitarias Módulo de Catálogo y Búsqueda.	Tester [1%].
Ejecutar pruebas unitarias Módulo de Visualización y Seguimiento.	Tester [1%].
Ejecutar pruebas unitarias Módulo de Reseñas.	Tester [1%].
Ejecutar pruebas unitarias Módulo de Suscripción y Donación.	Tester [1%].
Ejecutar pruebas unitarias Módulo de Publicación de Obras.	Tester [1%].
Ejecutar pruebas unitarias Módulo de Notificaciones.	Tester [1%].
Ejecutar pruebas unitarias Módulo de Contrato.	Tester [1%].
Ejecutar pruebas unitarias Módulo de Reportes.	Tester [1%].
Ejecutar pruebas unitarias Módulo de Configuración.	Tester [1%].
<b>Resultados de Pruebas Unitarias</b>	
Revisión completa de Pruebas unitarias.	Tester [80%].
<b>Ejecución de Pruebas</b>	

<b>Ejecución Pruebas de Validación de Ingresos de Datos</b>	
Usuario crea cuenta en la plataforma 01.	Tester.
Usuario crea cuenta en la plataforma 02.	Tester.
Usuario quiere registrarse en la plataforma 03.	Tester.
<b>Ejecución Pruebas de Lógica de los Módulos Principales</b>	
Usuario paga suscripción 01.	Tester [50%].
Autor adjunta la portada al publicar una obra 02.	Tester.
Administrador aprueba publicación 03.	Tester [50%].
<b>Ejecución Pruebas de Integración entre los Módulos del Sistema</b>	
Colecciones y Categorías existentes 01.	Tester.
Ver estadística de publicación al dar like 02.	Tester.
Publicar y visualizar el contenido de la obra 03.	Tester.
<b>Ejecución Pruebas de Carga</b>	
Peticiones por segundo de la API REST al endpoint de usuarios 01.	Scrum Master [10%].
Peticiones por segundo de la API REST al endpoint de catálogo 02.	Scrum Master [10%].
Peticiones por segundo de la API REST al endpoint de catálogo 03.	Scrum Master [10%].
Peticiones por segundo de la API REST al endpoint de catálogo 04.	Scrum Master [10%].
<b>Ejecución Pruebas de Seguridad por Niveles de Usuario</b>	
Ingreso de Usuario Administrador 01.	Product Manager [10%].
Ingresar a una ruta sin permisos 02.	Product Manager [10%].
Autor puede visualizar reportes y detalles de sus publicaciones 03.	Product Manager [10%].
<b>Resultados de Ejecución pruebas de</b>	



<b>Validación de Ingresos de Datos</b>	
<b>Resultados de Ejecución pruebas de Lógica de los Módulos Principales</b>	
<b>Resultados de Ejecución pruebas de integración entre los módulos del sistema</b>	
<b>Resultados de Ejecución pruebas de carga</b>	
<b>Resultados de Ejecución pruebas de seguridad por niveles de usuario</b>	
Preparar Demo 1 del sistema.	Arquitecto de Software [25%]; Dev Front-end [25%]; Dev Back-end [25%].
<b>Entrega Trabajo Práctico Integrador N.º 2</b>	
<b>Entregar Demo 1</b>	
Preparar Demo 2 del sistema.	Arquitecto de Software [25%]; Product Manager [25%].
<b>Entregar Demo 2</b>	
<b>Revisión de Módulos del Sistema</b>	
Módulo de Registro y Actividad de Usuario.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Catálogo y Búsqueda.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%]
Módulo de Visualización y Seguimiento.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Reseñas.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Suscripción y Donación.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Publicación de Obras.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Notificaciones.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].

Módulo de Contrato.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Reportes.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
Módulo de Configuración.	Dev Back-end [10%]; Dev Front-end [10%]; Arquitecto de Software [5%].
<b>Implementación</b>	
Planificación de las tareas	Arquitecto de Software[25%]
Asignar los responsables	Arquitecto de Software[25%]
Especificaciones de Servidores	Arquitecto de Software[25%]
Equipos de Conectividad	Arquitecto de Software[25%]
Comunicaciones	Product Manager[10%]
Puestos de Trabajo	Product Manager[10%]
Infraestructura	Arquitecto de Software[25%]
Carga Inicial de Datos	Arquitecto de Software[25%]
Migración	Arquitecto de Software[25%]
Métodos de Replicación	Arquitecto de Software[25%]
Backups	Arquitecto de Software[10%]
Recuperación	Arquitecto de Software[25%]
Etapas del Método de Conversión	Arquitecto de Software[25%]
Instalación del Sistema	Arquitecto de Software[25%]
Verificación de Instalación del Sistema	Arquitecto de Software[25%]

Preparación de entrega 3: "Etapa de Desarrollo e Implementación" y Sistema	Arquitecto de Software[25%]; Product Manager[25%]; Scrum Master[25%]
<b>Entrega 3</b>	
<b>Realización del Póster</b>	
Diseñar Póster.	Product Manager [25%]; Scrum Master [25%]; Dev Front-end [25%].
Preparar entrega 1 póster.	Scrum Master [25%]; Dev Front-end [25%].
<b>Entregar Póster (Entrega 1)</b>	
Corregir Póster.	Product Manager [25%]; Scrum Master [25%]; Dev Front-end [25%].
Preparar entrega 2 Póster.	Scrum Master; Dev Front-end [50%].
<b>Entregar Póster (Entrega 2)</b>	
<b>Capacitación</b>	
Realizar manual de usuario.	Product Manager [25%]; Scrum Master [25%]; Tester [25%].
Realizar tutoriales.	Product Manager [25%]; Scrum Master [25%]; Tester [25%].
Diseñar plan de capacitación.	Product Manager [25%]; Scrum Master [30%].

Entrega de material en la plataforma para usuarios finales.	Product Manager [25%]; Scrum Master [25%].
<b>Capacitar Usuarios Internos de la Organización</b>	
Capacitación de Administradores del Sistema.	Product Manager [25%]; Scrum Master [25%].
Capacitación de Gestor de Contenido.	Product Manager [25%]; Scrum Master [25%].
Capacitación de Usuario (Autor y Lector).	Product Manager [25%]; Scrum Master [25%].
Evaluación de las capacitaciones.	Product Manager [25%]; Scrum Master [25%].
<b>Revisión de Módulos del Sistema</b>	
Módulo de Registro y Actividad de Usuario.	Dev Back-end; Dev Front-end; Product Manager; Scrum Master; Tester; Arquitecto de Software
Módulo de Catálogo y Búsqueda.	Dev Back-end; Dev Front-end; Product Manager; Scrum Master; Tester; Arquitecto de Software
Módulo de Visualización y Seguimiento.	Dev Back-end; Dev Front-end; Product Manager; Scrum Master; Tester; Arquitecto de Software

Módulo de Reseñas.	Dev Back-end; Dev Front-end; Product Manager; Scrum Master; Tester; Arquitecto de Software
Módulo de Suscripción y Donación.	Dev Back-end; Dev Front-end; Product Manager; Scrum Master; Tester; Arquitecto de Software
Módulo de Publicación de Obras.	Dev Back-end; Dev Front-end; Product Manager; Scrum Master; Tester; Arquitecto de Software
Módulo de Notificaciones.	Dev Back-end; Dev Front-end; Product Manager; Scrum Master; Tester; Arquitecto de Software
Módulo de Contrato.	Dev Back-end; Dev Front-end; Product Manager; Scrum Master; Tester; Arquitecto de Software
Módulo de Reportes.	Dev Back-end; Dev Front-end; Product Manager; Scrum Master; Tester; Arquitecto de Software
Módulo de Configuración.	Dev Back-end; Dev Front-end; Product Manager; Scrum Master; Tester; Arquitecto de Software
<b>Correcciones generales del sistema</b>	
<b>14° Exposición Anual de Proyectos de Sistemas</b>	

Tabla 2. 3 Recursos para cada una de las actividades.

## 2. Diagrama de Recursos

Recurso	Mar	Abr	May	Jun	Jul	Ago	Sep	Oct	Nov	Cantidad Total de Horas
<b>Arquitecto de Sw</b>	81.5	230.5	48	168.5	224	16	31.75	124.85	57	982.1
<b>Product Manager</b>	134	230.8	8.8	144.8	186.75	137.6	34.05	113.85	57	1047.65
<b>Scrum Master</b>	65.25	238.75	83.5	168.5	208.9	150.4	38.2	111.4	57	1121.9
<b>Dev Front-end</b>	0	195	98.75	168.5	224	22.3	147.1	104.8	57	1017.45
<b>Dev Back-end</b>	0	195	98.75	168.5	196.1	62.4	104.3	95.7	57	977.75
<b>Tester</b>	71	219	60.5	167	208.9	150.4	82.05	70.15	57	1086

Tabla 2. 4 Cantidad de horas mensuales y totales por recurso.

Recurso	Tarea	Cantidad de horas por tarea
<b>Arquitecto de Software</b>	Definir herramientas de gestión de proyectos	10
	Gestión de configuración del software	30
	Configurar herramientas de gestión de proyecto	30
	Preparación entrega 1: "Etapa de Definición de Requerimientos"	1.5
	Definir objetivos y alcances preliminares del nuevo sistema	22
	Elaborar modelo lógico de sistemas relevados	30
	Investigar aplicaciones similares	20
	Definir salida del sistema	63.25
	Definir objetivos y alcances definitivos del sistema	64
	Preparación entrega 2: "Etapa de Diseño"	1.5
	Definir tecnologías de desarrollo	56
	Planificar implementación de módulos del sistema	33.25
	Preparación de entrega 3: "Etapa de Desarrollo e Implementación" y Sistema	2.5
	Preparar demo 2 del sistema	8.5
	Preparar demo 1 del sistema	1.7
Documentación Módulos del Sistema	2.4	

Preparar entrega 2 póster	2
Módulo de Registro y Actividad de Usuario	12.4
Módulo de Catálogo y Búsqueda	8
Módulo de Visualización y Seguimiento	5.2
Módulo de Reseñas	10.8
Módulo de Suscripción y Donación	6
Módulo de Publicación de Obras	6.8
Módulo de Notificaciones	1.6
Módulo de Contrato	3.6
Módulo de Reportes	1.6
Módulo de Configuración	2
Módulo de Registro y Actividad de Usuario	0.4
Módulo de Catálogo y Búsqueda	0.4
Módulo de Visualización y Seguimiento	0.4
Módulo de Reseñas	0.4
Módulo de Suscripción y Donación	0.4
Módulo de Publicación de Obras	0.8
Módulo de Notificaciones	0.4
Módulo de Contrato	0.4
Módulo de Reportes	0.4
Módulo de Configuración	0.4
Correcciones generales de las etapas anteriores	240
Trazabilidad general de los elementos del sistema	167
Backups	2
Métodos de Replicación	1.6
Migración	6
Carga Inicial de Datos	6.5
Infraestructura	2
Equipos de Conectividad	2
Especificaciones de Servidores	6
Asignar los responsables	2
Planificación de las tareas	4
Instalación del Sistema	4
Recuperación	2
Verificación de Instalación del Sistema	4
Módulo de Registro y Actividad de Usuario	8

	Módulo de Catálogo y Búsqueda	8
	Módulo de Visualización y Seguimiento	8
	Módulo de Reseñas	8
	Módulo de Suscripción y Donación	16
	Módulo de Publicación de Obras	8
	Módulo de Notificaciones	8
	Módulo de Contrato	8
	Módulo de Reportes	8
	Módulo de Configuración	8
	Etapas del Método de Conversión	4
<b>Product Manager</b>	Estimar tiempos de actividades	22
	Realizar análisis de factibilidad	56
	Definir métodos de comunicación formal	20
	Diseñar Póster	6
	Realizar manual de usuario	30
	Corregir Póster	10.5
	Diseñar plan de capacitación	22
	Realizar tutoriales	20
	Definir actividades	10
	Definir equipo de trabajo y funciones principales	30
	Relevar y analizar en general sistemas similares	46
	Elaborar comparación de funcionalidades de sistemas similares	30
	Definir objetivos y alcances preliminares del nuevo sistema	22
	Preparación entrega 1: "Etapas de Definición de Requerimientos"	1.5
	Preparación entrega 2: "Etapas de Diseño"	1.5
	Preparar demo 2 del sistema	3.4
	Preparación de entrega 3: "Etapas de Desarrollo e Implementación" y Sistema	2.5
Generar una comunidad que respalde el proyecto y ayude a darse a conocer el mismo. Establecer tiempos a largo plazo coordinando las tareas fundamentales. Considerar mercados potenciales basado en el contenido que se vaya generando en los períodos de prueba	25	
Se da prioridad a la tarea que genera el retraso. con asistencia al miembro que lo necesite para acelerar su	25	



finalización o visualizar una solución alternativa que no cause consecuencia en cascada.	
Se llevará un control coordinado del desarrollo de funciones. replanteo de alguna librería o tecnología a usar y una vez que finalice la función realizada por alguno de los integrantes se pasará a otro del equipo para que lo revise. Realizar refactoring	25
Se realizará un estudio detallado e investigación para conocer mejor el negocio. Tenemos contactos que pueden brindarnos asesoramiento más detallado en la industria que planteamos para utilizar en el proyecto.	25
Organizar y llevar a cabo capacitaciones referidas a las tecnologías que vamos a usar en el proyecto. La persona con más habilidad en la tecnología puede capacitar al miembro del equipo que lo necesite. También se puede contar con la compra de cursos webs	25
Entrenar a los trabajadores en procedimientos para detectar requerimientos. Desarrollar análisis de Seguridad de la Tarea y procedimientos seguros de trabajo.	25
Planificar implementación de módulos del sistema	4.1
Detectar tecnologías en sistemas similares	184
Realizar análisis de riesgo	24.5
Entrega de material en la plataforma para usuarios finales	4
Planificar pruebas de integración entre los módulos del sistema	40
Correcciones generales de las etapas anteriores	96
Planificar pruebas de carga	32
Planificar pruebas de seguridad por niveles de usuario	32
Capacitación de Usuario (Autor y Lector)	1.5
Capacitación de Gestor de Contenido	1.5
Capacitación de Administradores del Sistema	1.5
Evaluación de las capacitaciones	10
Planificar pruebas de Validación de Ingresos de Datos	16
Planificar pruebas de Lógica de los Módulos Principales	16
Autor puede visualizar reportes y detalles de sus publicaciones 03	0.8
Ingresar a una ruta sin permisos 02	0.8
Ingreso de Usuario Administrador 01	0.8
Trazabilidad general de los elementos del sistema	8.35
Puestos de Trabajo	1.6

	Comunicaciones	0.8
	Módulo de Registro y Actividad de Usuario	8
	Módulo de Catálogo y Búsqueda	8
	Módulo de Visualización y Seguimiento	8
	Módulo de Reseñas	8
	Módulo de Suscripción y Donación	16
	Módulo de Publicación de Obras	8
	Módulo de Notificaciones	8
	Módulo de Contrato	8
	Módulo de Reportes	8
	Módulo de Configuración	8
	<b>Scrum Master</b>	Definir actividades
Justificar proyecto a partir de necesidades encontradas		40
Elaborar comparación de funcionalidades de sistemas similares		30
Definir modelos de datos		161.25
Definir Sprint		26.75
Preparar entrega 1 póster		1.5
Definir métodos de comunicación formal		20
Realizar análisis de factibilidad		56
Investigar aplicaciones similares		20
Definir objetivos y alcances preliminares del nuevo sistema		22
Preparación entrega 1: "Etapa de Definición de Requerimientos"		1.5
Preparación entrega 2: "Etapa de Diseño"		1.5
Preparación de entrega 3: "Etapa de Desarrollo e Implementación" y Sistema		2.5
Diseñar Póster		6
Corregir Póster		10.5
Realizar manual de usuario		30
Realizar tutoriales		20
Diseñar plan de capacitación		26.4
Entrega de material en la plataforma para usuarios finales		4
Planificar pruebas de carga		32
Planificar pruebas de seguridad por niveles de usuario	32	

	Planificar pruebas de integración entre los módulos del sistema	40
	Correcciones generales de las etapas anteriores	240
	Trazabilidad general de los elementos del sistema	150.3
	Planificar pruebas de Validación de Ingresos de Datos	16
	Planificar pruebas de Lógica de los Módulos Principales	16
	Peticiones por segundo de la API REST al endpoint de catálogo 04	0.8
	Peticiones por segundo de la API REST al endpoint de catálogo 03	0.8
	Peticiones por segundo de la API REST al endpoint de catálogo 02	0.8
	Peticiones por segundo de la API REST al endpoint de usuarios 01	0.8
	Capacitación de Administradores del Sistema	1.5
	Capacitación de Gestor de Contenido	1.5
	Capacitación de Usuario (Autor y Lector)	1.5
	Evaluación de las capacitaciones	10
	Módulo de Registro y Actividad de Usuario	8
	Módulo de Catálogo y Búsqueda	8
	Módulo de Visualización y Seguimiento	8
	Módulo de Reseñas	8
	Módulo de Suscripción y Donación	16
	Módulo de Publicación de Obras	8
	Módulo de Notificaciones	8
	Módulo de Contrato	8
	Módulo de Reportes	8
	Módulo de Configuración	8
<b>Dev Front-end</b>	Definir interfaces de usuario	63.25
	Capacitarse en las tecnologías elegidas	103.25
	Definir módulos del sistema	64
	Definir salida del sistema	63.25
	Preparación entrega 2: "Etapa de Diseño"	1.5
	Módulo de Registro y Actividad de Usuario	12
	Módulo de Configuración	10.4
	Módulo de Reportes	9.6
	Módulo de Contrato	14.4

Módulo de Notificaciones	8
Módulo de Publicación de Obras	12
Módulo de Suscripción y Donación	8
Módulo de Reseñas	4
Módulo de Visualización y Seguimiento	9.6
Módulo de Catálogo y Búsqueda	14.4
Módulo de Registro y Actividad de Usuario	24.8
Módulo de Catálogo y Búsqueda	16
Módulo de Visualización y Seguimiento	10.4
Módulo de Reseñas	21.6
Módulo de Suscripción y Donación	12
Módulo de Publicación de Obras	13.6
Módulo de Notificaciones	3.2
Módulo de Contrato	7.2
Módulo de Reportes	3.2
Módulo de Configuración	4
Módulo de Registro y Actividad de Usuario	0.8
Módulo de Catálogo y Búsqueda	0.8
Módulo de Visualización y Seguimiento	0.8
Módulo de Reseñas	0.8
Módulo de Suscripción y Donación	0.8
Módulo de Publicación de Obras	1.6
Módulo de Notificaciones	0.8
Módulo de Contrato	0.8
Módulo de Reportes	0.8
Módulo de Configuración	0.8
Correcciones generales de las etapas anteriores	240
Trazabilidad general de los elementos del sistema	167
Módulo de Registro y Actividad de Usuario	8
Módulo de Catálogo y Búsqueda	8
Módulo de Visualización y Seguimiento	8
Módulo de Reseñas	8
Módulo de Suscripción y Donación	16
Módulo de Publicación de Obras	8
Módulo de Notificaciones	8
Módulo de Contrato	8

	Módulo de Reportes	8
	Módulo de Configuración	8
<b>Dev Back-end</b>	Definir métodos de comunicación entre partes del sistema	63.25
	Definir módulos del sistema	64
	Definir salida del sistema	63.25
	Preparación entrega 2: "Etapa de Diseño"	1.5
	Capacitarse en las tecnologías elegidas	103.25
	Módulo de Registro y Actividad de Usuario	12
	Módulo de Configuración	10.4
	Módulo de Reportes	6.4
	Módulo de Contrato	14.4
	Módulo de Notificaciones	8
	Módulo de Publicación de Obras	12
	Módulo de Suscripción y Donación	16
	Módulo de Reseñas	4
	Módulo de Visualización y Seguimiento	15.2
	Módulo de Catálogo y Búsqueda	14.4
	Módulo de Configuración	4
	Módulo de Reportes	3.2
	Módulo de Contrato	7.2
	Módulo de Notificaciones	3.2
	Módulo de Publicación de Obras	13.6
	Módulo de Suscripción y Donación	12
	Módulo de Reseñas	21.6
	Módulo de Visualización y Seguimiento	10.4
	Módulo de Catálogo y Búsqueda	16
	Módulo de Registro y Actividad de Usuario	24.8
	Módulo de Configuración	0.8
	Módulo de Reportes	0.8
	Módulo de Contrato	0.8
	Módulo de Notificaciones	0.8
	Módulo de Publicación de Obras	1.6
Módulo de Suscripción y Donación	0.8	
Módulo de Reseñas	0.8	
Módulo de Visualización y Seguimiento	0.8	

	Módulo de Catálogo y Búsqueda	0.8
	Módulo de Registro y Actividad de Usuario	0.8
	Correcciones generales de las etapas anteriores	240
	Trazabilidad general de los elementos del sistema	116.9
	Módulo de Configuración	8
	Módulo de Reportes	8
	Módulo de Contrato	8
	Módulo de Notificaciones	8
	Módulo de Publicación de Obras	8
	Módulo de Suscripción y Donación	16
	Módulo de Reseñas	8
	Módulo de Visualización y Seguimiento	8
	Módulo de Catálogo y Búsqueda	8
	Módulo de Registro y Actividad de Usuario	8
<b>Tester</b>	Definir historias de usuario	63.25
	Capacitarse en las tecnologías elegidas	103.25
	Realizar manual de usuario	30
	Realizar tutoriales	20
	Detectar funciones e interfaces en sistemas similares	184
	Ejecutar pruebas unitarias Módulo de Configuración	0.08
	Ejecutar pruebas unitarias Módulo de Reportes	0.08
	Ejecutar pruebas unitarias Módulo de Contrato	0.08
	Ejecutar pruebas unitarias Módulo de Notificaciones	0.08
	Ejecutar pruebas unitarias Módulo de Publicación de Obras	0.08
	Ejecutar pruebas unitarias Módulo de Suscripción y Donación	0.08
	Ejecutar pruebas unitarias Módulo de Reseñas	0.08
	Ejecutar pruebas unitarias Módulo de Visualización y Seguimiento	0.08
	Ejecutar pruebas unitarias Módulo de Catálogo y Búsqueda	0.08
	Ejecutar pruebas unitarias Módulo de Registro y Actividad de Usuario	0.08
	Planificar pruebas de integración entre los módulos del sistema	40
	Planificar pruebas de carga	32
	Planificar pruebas de seguridad por niveles de usuario	32
	Correcciones generales de las etapas anteriores	240
	Trazabilidad general de los elementos del sistema	150.3
Planificar pruebas de Validación de Ingresos de Datos	16	

Planificar pruebas de Lógica de los Módulos Principales	16
Usuario quiere registrarse en la plataforma 03	8
Usuario crea cuenta en la plataforma 02	8
Usuario crea cuenta en la plataforma 01	8
Administrador aprueba publicación 03	4
Autor adjunta la portada al publicar una obra 02	8
Usuario paga suscripción 01	4
Publicar y visualizar el contenido de la obra 03	8
Ver estadística de publicación al dar like 02	8
Colecciones y Categorías existentes 01	8
Revisión completa de Pruebas unitarias	6.4
Módulo de Registro y Actividad de Usuario	8
Módulo de Catálogo y Búsqueda	8
Módulo de Visualización y Seguimiento	8
Módulo de Reseñas	8
Módulo de Suscripción y Donación	16
Módulo de Publicación de Obras	8
Módulo de Notificaciones	8
Módulo de Contrato	8
Módulo de Reportes	8
Módulo de Configuración	8

Tabla 2. 5 Cantidad de horas por recurso y tarea.

Nombre del Recurso	Tasa	Capacidad Máxima	Acumular	Calendario Base
<b>Arquitecto de Software</b>	\$ 350,00/hora	100%	Prorratio	Estándar
<b>Product Manager</b>	\$ 300,00/hora	100%	Prorratio	Estándar
<b>Scrum Master</b>	\$ 275,00/hora	100%	Prorratio	Estándar
<b>Dev Front-end</b>	\$ 210,00/hora	100%	Prorratio	Estándar
<b>Dev Back-end</b>	\$ 210,00/hora	100%	Prorratio	Estándar
<b>Tester</b>	\$ 110,00/hora	100%	Prorratio	Estándar

Tabla 2. 6 Utilización de los recursos

## 2.1. Gráficos de barras que muestran la carga horaria por mes:

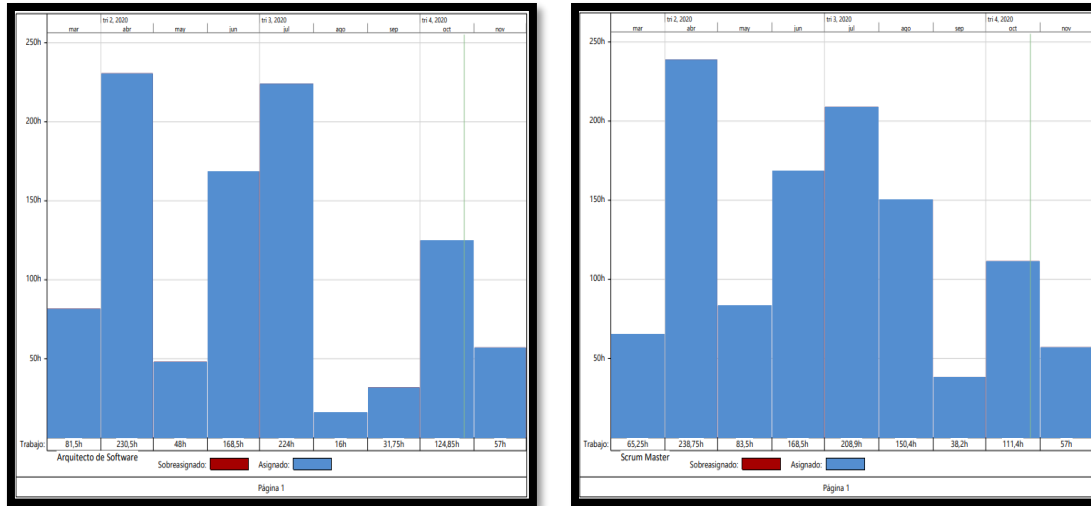


Figura 2. 23 Gráfico de recursos Arquitecto de Software y Scrum Master

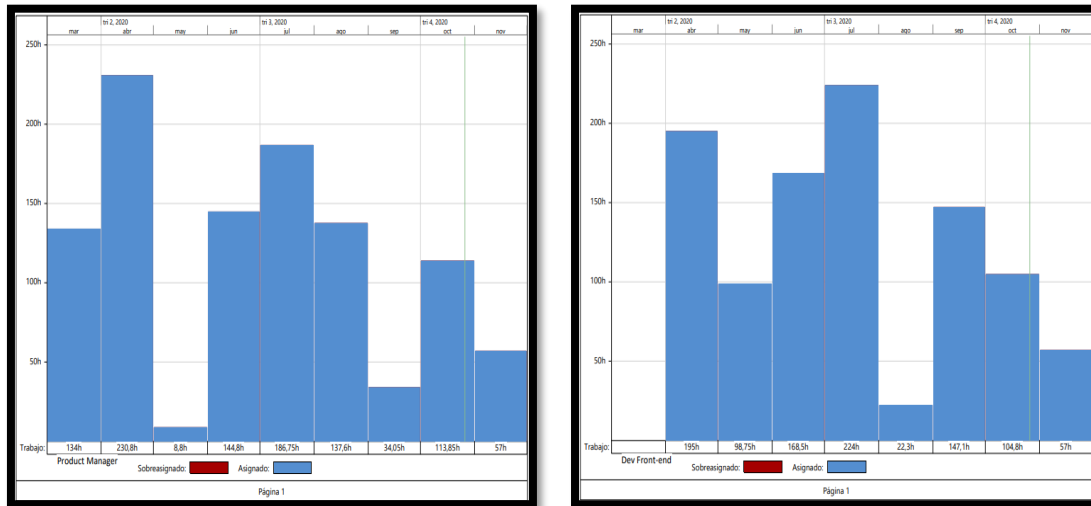


Figura 2. 24 Gráfico de recursos Product Manager y Dev Front-end



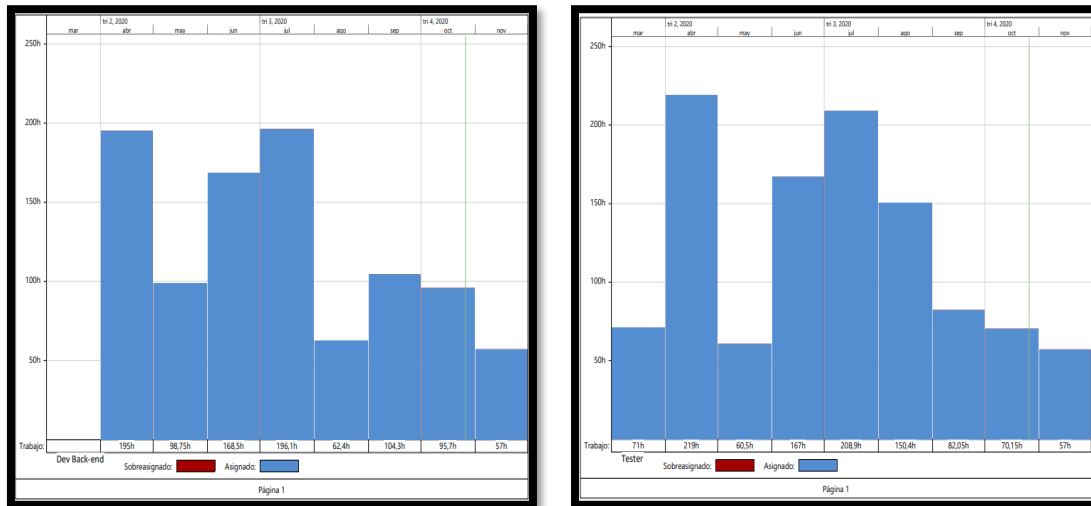


Figura 2. 25 Gráfico de recursos Dev Back-end y Tester

Para la realización del diagrama de recursos, se considera días de trabajo de lunes a viernes con carga horaria de 8 horas reloj. En el desarrollo del mismo, se encuentran sobreasignaciones y superposiciones horarias, lo que ocasiona una pérdida de trabajo efectivo. Esto se arregla para maximizar la colaboración en equipo y su consecuente redistribución.

Se tiene en cuenta por mes que cada recurso no puede exceder las 240 horas. Además, el recurso que más ganancia tiene, en términos de hora, se reduce levemente para que el proyecto no sea tan caro, como así también se dispuso al de menor costo/hora con carga mayor para solventar la sobreasignación.

Para una visualización más detallada del diagrama de recursos, la asignación de los mismos, y su utilización en función de horas mensuales. (**VER ANEXO N.º 2: DIAGRAMA DE RECURSOS**).

### 3. Análisis de Factibilidad

Para guiar la toma de decisiones en la evaluación del proyecto, se hace un estudio de factibilidad para identificar las posibilidades de éxito o fracaso del mismo. De esta manera, se puede decidir si se procede o no a la implementación. En base a esto se producen recomendaciones que se tienen en cuenta para mejorar el diseño.

Se analizan las siguientes factibilidades:

1. **Factibilidad Técnica:** Indica si se dispone de los conocimientos y habilidades en el manejo métodos, procedimientos, y funciones requeridas para el desarrollo e implementación del proyecto.
2. **Factibilidad Legal:** Requerimientos legales del proyecto para su operación, y aprobación respectiva.
3. **Factibilidad Económica:** Comparación entre beneficios vs costos.
4. **Factibilidad Operativa:** Comprende una determinación de la probabilidad de que un nuevo sistema se use como se supone.

Al planificar, se presta especial atención en el análisis de la factibilidad técnica y la factibilidad legal.

#### 3.1. Factibilidad Técnica

El núcleo del proyecto y los beneficios que puedan derivar de su realización y ejecución dependen por completo de un correcto desenvolvimiento técnico. A pesar de que el resultado final es un producto, en términos de negocio lo que se ofrece es un servicio. Concretamente, un servicio de lectura y publicación de cómics.

Para brindar este servicio a la comunidad se requiere de una infraestructura adecuada que soporte toda la plataforma, su arquitectura y el tráfico generado por las interacciones.

##### 3.1.1. Infraestructura general

Se destaca la infraestructura de servidores, la que da soporte y sirve de base al funcionamiento del servicio. Se pone énfasis en la tasa de transferencia, y capacidad de almacenamiento y procesamiento.

Se utilizan dos servidores para correr el back-end, uno principal y otro como respaldo. En éste corren los sistemas de gestión de bases de datos utilizados y servicios de procesamiento. Se cuenta, también, con un servidor enfocado en el almacenamiento de archivos, del estilo servidor CDN (Content Delivery Network).

Se consideran los siguientes parámetros y promedios, de utilización en el "peor" de los casos o de mayor carga requerida, en un período mensual:

- ❖ Cantidad de usuarios: 1000.
- ❖ Tamaño de imagen: 300 KB.
- ❖ Cantidad de imágenes por episodio: 30.
- ❖ Cantidad de cómics leídos por usuario: 20 (considerando usuarios que retoman lecturas no finalizadas).
- ❖ Cantidad de cómics en la plataforma: 100.
- ❖ Cantidad de episodios por cómic: 10.

En promedio, mensualmente se considera un volumen de transferencia de:

$$Tr = 1000 \text{ usuarios} * 20 \text{ lecturas} * 30 \text{ imágenes} * 300 \text{ KB} = 171 \text{ GB}$$

Así, en un rápido análisis, se considera una comunidad ya establecida que mantiene gran cantidad de lecturas mensuales, la tasa de transferencia mínima, requerida para servir todas las consultas, es de aproximadamente 171 GB mensuales.

También, se consideran 10 GB de almacenamiento requeridos, aproximadamente, para la base de datos que dará soporte a la estructura e interrelación de datos. Se puede estimar la capacidad de almacenamiento mínima como:

$$AlmacImágenes = 100 \text{ cómics} * 10 \text{ episodios} * 30 \text{ imágenes} * 300 \text{ KB} = 8.58 \text{ GB}$$

$$AlmacTotal = AlmacBD + AlmacImágenes = 10 \text{ GB} + 8.58 \text{ GB} = 18.58 \text{ GB}$$

El volumen de almacenamiento para los parámetros propuestos no supera los 20 GB. La contratación de servicios y servidores de almacenamiento es extremadamente económica en la actualidad, por lo tanto, no representa un riesgo o necesidad de gran inversión.

Recabando información de proveedores de renombre de servidores privados virtuales (VPS) y servicios de almacenamiento (que dan soporte a toda la arquitectura) se observa que, de acuerdo a los requerimientos especificados anteriormente, la contratación de infraestructura no representa una gran inversión de capital, ni un riesgo financiero para la organización.

En la siguiente tabla se especifican los precios de servidores privados virtuales ofrecidos por DigitalOcean. [VER SECCIÓN 3.1.7. ESCALABILIDAD Y SEGURIDAD](#), para entender el impacto en caso de que el procesamiento y almacenamiento requeridos se incrementen drásticamente.

Memoria	CPU	Transferencia mensual	Almacenamiento (SSD)	Precio
1 GB	1 CPU	1 TB	25 GB	U\$S 5 / mes
2 GB	1 CPU	2 TB	50 GB	U\$S 10 / mes
3 GB	1 CPU	3 TB	60 GB	U\$S 15 / mes
2 GB	2 CPU	3 TB	60 GB	U\$S 15 / mes
1 GB	3 CPU	3 TB	60 GB	U\$S 15 / mes
4 GB	2 CPU	4 TB	80 GB	U\$S 20 / mes

Tabla 2. 7 Capacidades y precios de VPS (servidor privado virtual) de DigitalOcean.

Recuperado de: [DIGITALOCEAN.COM/PRICING](https://digitalocean.com/pricing)

Fecha: 14/04/2020

Es apreciable que estos primeros seis planes ofrecidos por DigitalOcean no varían demasiado en precio. Además, se puede optar por una alternativa más enfocada en memoria u otra más enfocada en procesamiento, según se requiera, sin incurrir en gastos extras.

### 3.1.2. Volumen, tipos de datos y captura de los mismos

En la sección anterior se detalla que se requiere como mínimo, en vistas de que la arquitectura soporta un gran número de usuarios y de contenido, 20 GB de espacio de almacenamiento. A su vez, la transferencia promedio de datos resulta en 171 GB mensuales.

Los tipos de datos que se manejan consisten en:

- ❖ Datos planos: Datos y estructuras de datos, interrelación entre los mismos, que representan la información para la organización, almacenados en un sistema de gestión de base de datos.

- ❖ Imágenes: Son datos estructurados de forma particular, y como tales, deben tratarse acorde haciendo uso del procesamiento necesario para optimizar el almacenamiento sin perder calidad. Contenido multimedia.

Los datos planos, es decir aquellos que se representan como datos primitivos (texto, números, *booleanos*) y el esquema que los organiza (relaciones entre estos, esquema de la base de datos) están soportados óptimamente por el motor de base de datos a utilizar.

Las imágenes son el eje central y crucial en lo que respecta a almacenamiento de archivos para la plataforma. No existen herramientas que se destaquen para manipularlas, es decir, su procesamiento está sujeto a los requerimientos de la aplicación, y posibilidades técnicas de acuerdo a las capacidades de cómputo disponibles.

La captura o ingreso de datos planos se realiza a partir de formularios sencillos, mediante una comunicación directa entre cliente (aplicación de lectura y panel de gestión web) y el servidor (back-end). Por otro lado, la subida de imágenes representa un reto en cuanto a tecnología a implementar para asegurar una correcta transferencia de grandes cantidades de datos.

### 3.1.3. Frecuencia y volumen de ingreso de datos

La carga de imágenes representa el mayor volumen y frecuencia de datos transferidos al servidor.

Al considerar los parámetros descritos en la [SECCIÓN 3.1.1. INFRAESTRUCTURA GENERAL](#), y 5% de usuarios de la plataforma como autores, si se sube 1 obra por mes, no se sobrepasa los 400 GB de transferencia mensual. Esto es, un 40% del límite establecido en los planes más económicos de servidores de la mayoría de las compañías, es decir que no hay un riesgo.

En el desarrollo de la arquitectura del back-end, se separa los diversos servicios que la componen en distintos servidores de acuerdo a requerimientos de almacenamiento y procesamiento.

### 3.1.4. Tecnologías a utilizar

La tecnología que se utiliza no representa un problema de obsolescencia, o dificultar el mantenimiento a lo largo del tiempo.

Las plataformas y herramientas se han elegido con detenimiento, poniendo atención a la evolución de éstas a lo largo del tiempo y la estabilidad que proporcionan a la hora de soportar una arquitectura como la que se presenta en la plataforma *OmicS*.

Para el desarrollo del back-end se utiliza RUST, es un lenguaje de programación compilado, de propósito general y multiparadigma que está siendo desarrollado por Mozilla. Ha sido diseñado para ser un lenguaje seguro, concurrente y práctico. Es un lenguaje de programación que soporta programación funcional pura, por procedimientos, imperativa y orientada a objetos.

El balanceo de carga se realiza mediante API Proxy.

El front-end se realiza con Angular 9, el cual es un framework para aplicaciones web desarrollado en TypeScript, de código abierto mantenido por Google, que se utiliza para crear y mantener aplicaciones web, con capacidad de Modelo Vista Controlador (MVC).

### 3.1.5. Comunicación entre partes del sistema

La comunicación entre las partes en este modelo cliente-servidor es importante ya que debe soportar tanto tipos de datos planos como imágenes del formato utilizado. La comunicación se realiza mediante una API REST, bajo el protocolo HTTP.

Aquellas partes que requieren gran optimalidad y tiempos de respuestas ínfimos en comparación con los datos transferidos es posible que utilicen gRPC y GraphQL. **gRPC** (*gRPC Remote Procedure Calls*) el cual es un sistema de llamada a procedimiento remoto de código abierto desarrollado en Google. Utiliza como transporte HTTP/2 y *Protocol Buffers* como lenguaje de descripción de interfaz. Proporciona características como autenticación, transmisión bidireccional y control de flujo, enlaces bloqueantes o no bloqueantes, cancelaciones y tiempos de espera. Genera enlaces multiplataforma entre cliente y servidor para muchos lenguajes.

### 3.1.6. Nivel de automatización de las funciones

Las automatizaciones a pequeña escala representan todo lo que gira en torno al desarrollo de la plataforma, como formateo y validación del código, generación de documentación de forma automática, correr *scripts* para realizar diversas tareas, realizar migraciones, actualización de dependencias del proyecto, testing automatizado, el cual incluye las pruebas de validación de ingreso de datos, prueba de lógica de los módulos principales, pruebas de integración entre los módulos del sistema, pruebas de carga, y pruebas de seguridad por niveles de usuario. Está contemplado como parte del soporte a la etapa de desarrollo.

Como principal automatización se tiene la integración continua mediante la cual la aplicación se pone en producción. Existen gran cantidad de servicios que ofrecen una capa gratuita para manejar sistemas de la magnitud de la plataforma *OmicS*. Solamente se requiere una adecuada configuración de las ramas del proyecto en el repositorio.

Se automatiza el balanceo de carga mediante el uso de un API Proxy para prevenir sobrecargas.

También forma parte de la automatización las migraciones, tanto de esquemas de la base de datos como de los mismos datos.

### 3.1.7. Escalabilidad y seguridad

Depende completamente de la arquitectura del sistema y las metodologías adoptadas, que permiten migrar partes de la plataforma a otros servidores sin afectar nada.

Es imperativo que la plataforma escale correctamente al crecer el número de usuarios, y requerimientos de transferencia de datos. Como primera opción y solución temporal en caso de que se alcancen los límites previstos, se propone el uso de un balanceador de carga entre distintas instancias de la aplicación corriendo. Todo esto mientras se prepara un despliegue de servicios separados de la misma plataforma en distintos servidores para que trabajen en conjunto, adoptando una arquitectura de servicios distribuidos, con vistas a microservicios.

En cuanto a seguridad son tenidos en cuenta los siguientes aspectos:

- ❖ Encriptación de contraseñas de usuario.
- ❖ Virtualización de la red en la que se comunican los servicios del back-end para impedir fugas de datos.
- ❖ Utilización de Docker como tecnología de contenedores para abrir, en cada servicio, únicamente aquellos puntos de acceso (puertos) relevantes.
- ❖ Único punto de acceso desde el exterior del servidor mediante API REST, previamente testada bajo distintos puntos de vista.
- ❖ Usuarios, accesos, y privilegios acordes al rol dentro de la organización de la persona interesada.

### 3.1.8. Funcionamiento ininterrumpido del sistema

El funcionamiento ininterrumpido del sistema es asegurado por los distintos proveedores de servicios, y recursos que se utilizan, asegurando un 99% de disponibilidad de la plataforma.

Sin embargo, también hay que considerar posibles interrupciones de la plataforma por errores propios del desarrollo o errores lógicos. Deben estar cubiertas todas las posibilidades de fallas mediante el testing automatizado de todas las pruebas antes mencionadas. Las tecnologías a utilizar aseguran estabilidad en el funcionamiento con años de trayectoria.

Para mitigar el error humano que conlleva todo desarrollo de aplicación, desde la codificación, el diseño, y la implementación de módulos independientes para su funcionamiento en conjunto, se automatiza todo el proceso mediante métodos de *Continuous Integration / Continuous Deployment* (¿Qué son la integración/distribución continuas (CI/CD)?, s/f).

La integración continua es un proceso de automatización para los desarrolladores. Si la CI tiene éxito, los cambios del código nuevo en una aplicación se diseñan, se prueban y se combinan periódicamente en un repositorio compartido. Esto soluciona el problema de que se desarrollen demasiadas divisiones de una aplicación al mismo tiempo, porque podrían entrar en conflicto entre sí. La implementación continua hace referencia a la liberación automática de los cambios que implementa el desarrollador desde el repositorio hasta la producción, para que los clientes puedan usarlos.

Esta automatización se realiza a nivel de codificación, construcción, *testing* y despliegue. Antes de la integración continua se contempla el siguiente proceso:

- ❖ **Build:** Todo lo que implica la construcción del software desde funcionalidades específicas, módulos, y la aplicación como un todo. La automatización acá depende de los scripts generados para construir la plataforma en su totalidad, y sus dependencias, con un simple comando de terminal. Incluye la misma plataforma y lo referente a la base de datos, como migraciones de tablas y datos.
- ❖ **Testing:** Antes de realizar cualquiera de estas construcciones se ejecutan todas las pruebas (antes mencionadas) pertinentes para asegurar que la lógica del código cumpla con la funcionalidad requerida según las historias de usuario. En el próximo apartado se detallan las metodologías de *testing* aplicadas.

En lo referido a implementación (despliegue) continua se automatiza, mediante servicios desarrollados para tal fin, la puesta en producción desde el repositorio donde se encuentra el código hasta el servidor que pondrá en funcionamiento la plataforma.

De aquí surge la idea de *DevOps*, es decir, desarrolladores con capacidades de operaciones. La puesta en producción, o incluso en otros entornos (como entorno de *testing* o entorno de pruebas *alfa* para usuarios finales), es realizada de forma automática con el conocimiento y respectiva experiencia de los mismos desarrolladores. No se requiere de personal capacitado específicamente para esta actividad.

De lo anterior se desprende la estrategia de automatización adoptada para poner la plataforma en producción.

Con lo planteado se asegura una disponibilidad (*uptime*) del 99% del tiempo para la plataforma.



### 3.1.9. Métodos de desarrollo, testing, implementación y mantenimiento

Se sigue metodología ágil, lo que representa una rápida respuesta a posibles problemas, y avances con periodicidad semanal. En cuanto a desarrollo, los desarrolladores cuentan con habilidades de operación, siendo considerados bajo el término de *DevOps*.

Todas las etapas están cubiertas por técnicas de automatización que facilitan las actividades. Los desarrolladores centran su trabajo en ampliar, o mejorar las funcionalidades que surgen de los requerimientos de la aplicación. Además, futuros cambios en el código de la plataforma no deben retrasar al equipo de desarrollo, ni al de *testing* haciendo que estos vuelvan a probar las funcionalidades que en el pasado ya se aprobaron.

Por esto, el testing continuo del código, las funciones y la integración acompaña a todo el proceso de desarrollo, este se divide en los siguientes tipos de pruebas:

- ❖ Pruebas de Validación de Ingreso de Datos: Se realizan para validar que se ha especificado es lo que el usuario realmente quería, además se comprueba si los valores que ha introducido el usuario cumplen las restricciones impuestas por la aplicación.
- ❖ Pruebas de Lógica de los Módulos Principales: Se realizan para Lograr combinar módulos para que trabajen correctamente de forma conjunta.
- ❖ Pruebas de Integración entre los Módulos del Sistema: Se realizan para comprobar que distintos módulos funcionando en conjunto, cumplan su cometido y generen los resultados esperados. Para esto se ejecutan los módulos de acuerdo a los establecido por los distintos casos de uso, esperando que se satisfagan de forma adecuada
- ❖ Pruebas de Seguridad por Niveles de Usuario: Se realizan para Comprobar que cada usuario sólo puede acceder a las funcionalidades pertinentes para ese rol, evitando accesos no deseados a funciones que no debe poder ejecutar. Y solo permitir al usuario administrador ejecutar funciones propias de la configuración general del sistema.
- ❖ Pruebas de Carga: Se realizan para validar el rendimiento de funciones críticas en términos de complejidad de procesamiento, utilización de recurso o cantidad de usuarios en línea.

Cabe aclarar que los tests forman parte del código, cada test es código encargado de ejecutar porciones de la aplicación para constatar que se comportan de la forma esperada, de manera automática.

Estos tests utilizan aserciones lógicas para las comprobaciones, si éstas se cumplen aseguran que el comportamiento de la plataforma es el requerido. Y de aquí se destaca su mayor importancia, evitan la regresión: los desarrolladores y los testers no tienen que volver a probar cada parte de la aplicación cada vez que se va a poner nueva versión de la misma en producción.

Para cubrir la implementación, y según lo descrito en el apartado anterior, se utiliza una metodología de integración continua. Ya que los tests cubren gran parte de la funcionalidad de la plataforma de forma automática, se pueden ejecutar en cualquier servidor, sin la intervención de una persona. Así, una vez que nuevo código se sube al repositorio central de la aplicación, un servicio externo clona este repositorio, ejecuta los tests de forma autónoma, y si estos "pasan" (se cumplen las aserciones y no se encuentran errores) este mismo servicio se encarga de hacer una nueva *build* de la aplicación, y se realiza la puesta en producción.

Lo anterior se puede resumir mediante el siguiente flujo:

- Se desarrolla la plataforma, agregando, modificando, o eliminando código.
- Se realiza un *merge* del código con el existente en el repositorio local del desarrollador.
- El desarrollador sube los cambios al repositorio central de la aplicación.
- Un servicio de integración continua, externo, detecta que se han realizado nuevos cambios por lo que clona el repositorio localmente.
- Ejecuta los tests de forma automática.
- Si los tests no arrojan errores, se construye (*build*) de la plataforma desde el código haciendo uso de scripts de automatización.
- La plataforma construida se pone en producción (o en un entorno de *testing*, según se requiera) al subirlo al servidor correspondiente y ejecutar las rutinas para detener la versión anterior de la aplicación que ya estaba corriendo, y así correr la nueva versión.

El último punto trae a consideración la última etapa de automatización que se requiere: detener la vieja aplicación en ejecución y levantar la nueva que se ha construido a partir de los cambios generados en el código. Se recuerda que toda la arquitectura de la aplicación se basa en contenedores de Docker. Por lo tanto, cuando se construye la aplicación los artefactos resultantes se agrupan e insertan en una imagen de Docker para subirla al servidor, y así que desde el servidor se cree un nuevo contenedor a partir de esta imagen, esto es, ponerla en ejecución. Docker automatiza este proceso de forma eficiente, deteniendo el contenedor anterior y ejecutando el nuevo en cuestión de segundos.

Así, resulta imperceptible el cambio entre una versión de la aplicación, y otra al poner la nueva en ejecución. Con esto se asegura que los desarrolladores centren su tiempo en lo requerido en los casos de uso e historias de usuario.

### 3.1.10. Backups, recuperación y migraciones

De la base de datos se realizan backups diarios a un servidor secundario para posibles recuperaciones en caso de pérdidas accidentales, o por ataques externos.

Los backups de contenido multimedia no requieren atención por parte de quien mantiene la aplicación en producción. Estos utilizan servicios externos como CDN (Content Delivery Network) que aseguran la integridad y disponibilidad del contenido.

Se utiliza una base de datos relacional por la ventaja que aporta en cuanto a la integridad de los datos, ya que proveen un esquema que se debe respetar a la hora de insertar o actualizar datos. Pero esto también requiere una adecuada gestión en cuanto a la evolución de los esquemas de datos, haciendo uso de migraciones y la aplicación adecuada de las mismas. Se automatizan, tanto para esquemas y evolución, como para los mismos datos.

El backup de la base de datos es automático: se corre un proceso mediante cron, un servicio de los sistemas GNU/Linux encargado de ejecutar tareas (*scripts*) de forma periódica. Así se realiza una copia, todos los días a las 2:00 AM, de la base de datos completa, y se sube a un servicio de almacenamiento de archivos. Esto asegura que, en caso de errores críticos e inesperados a nivel de datos, se mantiene una copia de la base de datos con una vejez de a lo sumo de 1 día.

Esta actividad sigue un criterio determinado para nombrar los archivos que se almacenan en el servicio de archivos: el nombre del archivo SQL resultante de ejecutar la herramienta que genera el backup será ***nombre\_de\_la\_bd-fecha.sql***. Así, en el servicio de almacenamiento de archivos, se mantienen todas las copias realizadas hasta el momento, donde cada una, en su nombre de archivo, especifique la fecha de realización.

Otro punto de interés son las migraciones, éstas permiten que se modifiquen los esquemas de tablas y estructuras de datos que mantiene el SGBD (Sistema de Gestión de Base de Datos), para adaptarlos a los nuevos cambios de código. De esta forma, se pueden agregar, modificar o quitar campos de las tablas de una base de datos relacional de acuerdo a lo requerido por el código de la plataforma. Es importante porque estos esquemas son fijos y adecuados a los datos en una primera instancia de desarrollo, pero el código del sistema tiende a cambiar con el tiempo y, más aún, los modelos de datos, por lo que deben adaptarse nuevamente estas estructuras a nivel de base de datos.

Las migraciones se llevan a cabo con la herramienta Knex.js, la cual permite automatizar todo el proceso mediante simples scripts.

Cada vez que se realizan cambios sobre las estructuras de datos, se genera un nuevo script en una carpeta del proyecto especificada con anterioridad donde reside el código a ejecutar por el servidor en producción para adaptar la tabla del SGBD a los nuevos requerimientos.

Como última acotación, el código de la plataforma no requiere de backup ya que reside en un repositorio, mediante el cual se mantiene registro por fechas de cada cambio realizado, así como también el historial de versiones.

### 3.1.11. Integración con otros sistemas

La plataforma es modular, por lo que pueden agregarse módulos a la arquitectura sin interferir con los ya existentes. También presenta las abstracciones pertinentes para reemplazar servicios externos por otros, en caso de ser necesario.

Se integra con la API de Mercado Pago para el cobro de donaciones, y suscripciones de los lectores.

La ejecución de los módulos independientes de la plataforma se apoya en contenedores con tecnología Docker. Cada instancia de módulos en ejecución es independiente de las demás, sólo depende de una adecuada comunicación entre estos.

La comunicación asíncrona se realiza mediante un bus de datos que soporte grandes flujos de datos.

La comunicación síncrona es mediante una API REST (gRPC en caso de ser necesario) definida para tal fin.

### 3.1.12. Crecimiento estimado de la cantidad de usuarios: Comunidad

Se espera un crecimiento exponencial en el número de usuarios lectores, pero no representa un problema para la carga general de la plataforma. También, se espera un crecimiento lineal del número de usuarios autores, los cuales representan la mayoría de la carga de subida de contenido a la plataforma.

La aplicación se desarrolla teniendo presente todo el tiempo la escalabilidad horizontal necesaria para las siguientes iteraciones de su ejecución. Dicho de otra forma, el crecimiento exponencial de usuarios esperados no afecta la *performance* del servidor ya que, en caso de ser necesario, un servicio intermedio (a modo de *proxy*) ejecutará más instancias de la plataforma para responder en tiempo y forma todas las consultas generadas desde los clientes.

Se realizan algunas estimaciones del incremento de usuarios de forma mensual. Para este análisis se ha tenido en cuenta un total de 100 usuarios iniciales, y un porcentaje de adhesión de nuevos usuarios. Así, se comienza con un incremento del 40% en la

cantidad de usuarios gracias a la publicidad y el esfuerzo del área de Marketing para atraer nuevos usuarios. Durante el primer año este porcentaje representativo del incremento disminuye un 2%. A partir del primer año del lanzamiento de la plataforma se considera un incremento del 5%, considerando así un escenario poco favorable para la organización. Sin embargo, como se puede apreciar en la siguiente tabla, se alcanzan los 3700 usuarios, aproximadamente, en el segundo año.

Ya establecido esto, los esfuerzos de Marketing con el objetivo de dar a conocer la plataforma acompañan a la organización durante toda su existencia, por lo que se puede asegurar un crecimiento estable del número de usuarios tanto lectores como de autores, a lo largo del tiempo.

También se supone que el total de usuarios lectores de la plataforma realizan aportes. Para un análisis más realista [VER SECCIÓN 3.3 FACTIBILIDAD ECONÓMICA](#), donde se pone énfasis en la cantidad de meses requeridos para recuperar la inversión en función del número de usuarios.

Además, se establece el modelo económico que adopta la organización para asegurar el ingreso constante de dinero. Así, dado un escenario de poco incremento, de todas formas, se recupera la inversión antes del segundo año de operación de *Omics*.

Usuarios Iniciales		100
Meses	Porcentaje de Nuevos Usuarios	Cantidad de Usuarios
1	40.00%	140
2	38.00%	193
3	36.00%	262
4	34.00%	351
5	32.00%	463
6	30.00%	602
7	28.00%	771
8	26.00%	971
9	24.00%	1204
10	22.00%	1469
11	20.00%	1763
12	18.00%	2080

13	5.00%	2184
14	5.00%	2293
15	5.00%	2408
16	5.00%	2528
17	5.00%	2654
18	5.00%	2787
19	5.00%	2926
20	5.00%	3072
21	5.00%	3226
22	5.00%	3387
23	5.00%	3556
24	5.00%	3734

Tabla 2. 8 Porcentaje de incremento de usuarios mensualmente

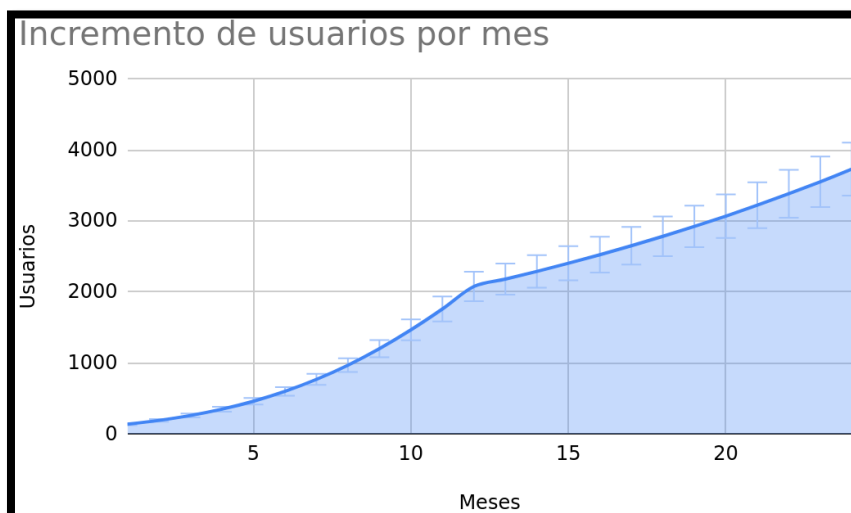


Figura 2. 26 Incremento del número de usuarios mensualmente.

**Conclusión:**

En base al análisis que se ha realizado, se puede concluir que el proyecto es factible técnicamente ya que para la conducción de métodos, funciones y procedimientos que se pretenden para el desarrollo y la implementación del proyecto, se cuenta con conocimientos varios, prácticas y habilidades. También se dispone del equipo y las herramientas para llevarlo a cabo.

Se mantienen en consideración los siguientes aspectos:

- ❖ Recursos de servidores y servicios, tanto para almacenamiento como para procesamiento, son económicos.
- ❖ Se presta especial atención en el volumen y transferencia de contenido multimedia.
- ❖ Modelo cliente-servidor: front-end y back-end. Comunicación con tecnología estable y relevante.
- ❖ Se desarrolla la arquitectura con escalabilidad en mente.
- ❖ Se automatizan tareas de desarrollo, el testing y el deployment.
- ❖ Se consideran todos los aspectos de seguridad influyentes.
- ❖ Se utiliza migración de esquemas de datos y de datos.
- ❖ Contenido multimedia seguro por uso de servicios externos. Backups a la base de datos automática y periódicamente.

## 3.2. Factibilidad Legal

Este apartado es muy importante para el uso de la plataforma, y conlleva ser muy específico tanto para los lectores, como para los autores propietarios de las obras publicadas en *OmicS*.

### 3.2.1. Privacidad y tratamiento de datos

Cuando una persona crea su usuario para acceder y hacer uso de la plataforma, tanto de lectura como de publicación, se solicitan datos que son facilitados por el usuario que deben garantizar su veracidad, exactitud, autenticidad y vigencia (*Ley 25.326 Protección de los Datos*, 2008).

### 3.2.2. Propiedad intelectual de la obra

Esta es una de las características más importantes, debido a que en el momento en que el autor suba contenido para que sea publicado en la plataforma *OmicS*, debe dar validez a los administradores de la plataforma, que cuenta con la autorización de el/los dueños de la propiedad intelectual de dicha obra para su publicación. La responsabilidad sobre lo publicado y su uso en la plataforma corre a cargo del usuario autor, sin responsabilizar a *OmicS* en caso de que haya uso indebido (*Ley 11723 del Régimen Legal de La Propiedad Intelectual*, 2009).

### 3.2.3. Duración y terminación de continuidad en la plataforma

Cada autor que esté en la plataforma tiene dos tipos de duración que se aclaran al momento de publicar su contenido en la plataforma que son:

- ❖ Indefinido: El mismo queda en la plataforma hasta que *Omic*s decida o se vea obligado a sacarlo (por motivos de infringir cambios legales futuros).
- ❖ Por período: Cada cierto plazo, previamente estipulado, la obra queda en la plataforma y el autor con un mes de antelación de finalizar dicho período, puede realizar y confirmar cambios en el acuerdo del contenido. En caso de que lo mencionado no se realice, el período se actualiza con lo pactado anteriormente de manera automática.

### 3.2.4. Distribución del contenido y derecho de uso

Al ingresar un cómic a la plataforma *Omic*s, al mismo se le otorga un derecho de comunicación pública el cual concede a que un grupo amplio de usuarios puedan acceder, y visualizar su obra sin haber distribución de por medio. Esto puede darse de modo compartido en otras plataformas o exclusivo.

Se tiene en cuenta que todo lo anterior mencionado entra bajo ciertos límites de los derechos de autor, dejando a *Omic*s el poder utilizar fragmentos, y citar el contenido para publicitar la misma. Aunque aún no contemplados en el desarrollo actual, puede darse el uso de otorgar no sólo el derecho de uso y distribución de la obra, sino también el de propiedad que haría a *Omic*s dueño de dicho contenido, con un acuerdo previo en caso de haber regalías o límites en la modificación de la obra, estos casos serían muy específicos.

### 3.2.5. Obligaciones y responsabilidades del usuario en la plataforma

En la plataforma *Omic*s, el usuario se compromete a:

Hacer un uso adecuado y lícito de la plataforma, así como de los contenidos y servicios, de conformidad con:

- La legislación aplicable en cada momento y región.
- Las condiciones generales de uso de la plataforma.
- El no infringir normas de propiedad intelectual de las obras publicadas.
- No distribuir de forma comercial capturas de las imágenes de cualquier contenido parcial de las obras.

### 3.2.6. Cookies

La aplicación *Omic*s se reserva el derecho de utilizar la tecnología de "cookies" en la plataforma, a fin de reconocerlo como usuario frecuente y personalizar el uso que realice mediante la preselección de categorías favoritas, y notificaciones. Las cookies de la plataforma se asocian únicamente con un usuario anónimo y su dispositivo, y no proporcionan los datos personales del mismo.



Debido al uso de las cookies, le es posible a *Omic*s reconocer el navegador y dispositivo utilizado por el usuario para facilitar contenido, mostrar sus preferencias de navegación, y avisos publicitarios. También sirve para reconocer diferentes perfiles de usuario, medir visitas y parámetros del tráfico para el control del número de entradas a la plataforma.

### Conclusión:

Al tener asesoría legal en el apartado de propiedad intelectual y detallado correctamente los usos de datos de los usuarios regidos bajo las leyes mencionadas, el proyecto es factible legalmente. Se han realizado varios pasos y evaluaciones que demuestran que el proyecto puede ponerse en marcha y mantenerse, mostrando evidencias de que se ha planeado cuidadosamente. Además, se puede asegurar de que el proyecto no infringe ninguna norma o ley establecida ya sea Municipal o Nacionalmente. Se garantiza el respeto a los acuerdos, convenios y reglamentos internos.

### 3.3. Factibilidad Económica

La factibilidad económica hace referencia a la disposición del capital en efectivo o de los créditos de financiamiento necesario para invertir en el desarrollo de la plataforma *Omic*s, el cual prueba que los beneficios a obtener son superiores a los costos en que se incurre al desarrollar e implementar la plataforma. Para esto se tiene en cuenta la recesión económica y la inflación de Argentina, para poder así determinar los costos a futuro.

Para conocer esto con más detalle se realiza un análisis de los costos totales de la plataforma, detallado a continuación:

#### 3.3.1. Gestión de costos

<b>Recurso</b>	<b>Costo</b>	<b>Período</b>
<b>1. Personal</b>		
1.1 Personal pagado	\$ 1.429.564,50	Único
1.2 Asesores pagados	\$ 8.000	Único (duración: 9 meses)
1.3 Asesores voluntarios	\$ 0.00	3 veces al mes (duración: 5 meses)
<b>2. Servidores y equipos</b>	\$ 9.142.49	Único (duración: 7 meses)

<b>3. Otros</b>	\$ 9000	Único (duración: últimos 2 meses)
<b>Total</b>	<b>\$1.455.706,99</b>	

Tabla 2. 9 Costos recurrentes durante 9 meses.

Durante los 9 meses de desarrollo de la plataforma se incurren en un costo de \$1.455.706,99. **VER SECCIÓN 4. COSTOS DESAGREGADOS POR RECURSOS** para un análisis más detallado.

### 3.3.2. Ingresos

Se plantea el modelo que adopta la gestión de la plataforma para generar ingresos de dinero por parte del usuario, cabe resaltar que este modelo se piensa como dinámico.

El primer paso importante para la plataforma, y sobre todo para la organización, es generar comunidad atrayendo nuevos usuarios en lugar de enfocarse inicialmente en la generación de ingresos constantes. De esta forma, se decide adoptar inicialmente un modelo de Donación, donde los usuarios aportan a los autores directamente, recibiendo la plataforma un porcentaje por ello.

Una vez que se genera suficiente comunidad, se pasa a un modelo de Suscripción, donde los usuarios pueden adherirse a una suscripción mensual para recibir contenido premium de los autores. Sin embargo, se conserva una cantidad determinada de contenido gratuito accesible para todos los usuarios registrados en la plataforma.

De esta forma, los autores reciben, además de donaciones, un porcentaje del ingreso total de la plataforma en función de la cantidad de visualizaciones que generan sus publicaciones.

Así, podemos considerar tanto las donaciones como las suscripciones de forma similar para realizar las estimaciones pertinentes. Sólo un porcentaje del total de usuarios de la plataforma realizan donaciones, y se suscriben.

### 3.3.3. Recuperación de la inversión

En una primera instancia, los usuarios de la plataforma retribuyen a los autores con un mínimo de \$100 mensualmente. De ese importe, *OmicS* se queda con el 70%, es decir \$70.

Con esto en mente, se realiza un análisis de la cantidad de usuarios donando, se construye un gráfico que muestra la cantidad de meses en los cuales se recupera la inversión que se realiza al inicio, para la plataforma. Así, con una cantidad de 1.300

usuarios donando \$100 mensualmente, en 12 meses aproximadamente se recupera la inversión inicial.

$$\text{Retorno} = \$ 1.455.706 / (70\% * \$100 * 1700 \text{ usuarios}) = 12.2 \text{ meses}$$

En un escenario más crítico, teniendo 850 usuarios donantes mensualmente, la inversión se recupera en un período de 24 meses.

El valor de las donaciones/suscripciones influye también en el tiempo necesario para recuperar la inversión. Por esto, en la figura anterior se han realizado estimaciones para donaciones/suscripciones con un valor de: \$30, \$50, \$75, \$100 y \$150.

Es inmediato notar que a mayor cantidad de usuarios las curvas convergen, asegurando recuperar la inversión en pocos meses. Por esto, hay que asegurar que al menos 2.000 usuarios que realizan aportes garantizan la recuperación de la inversión inicial en un período de 2 años, independientemente del valor de las donaciones/suscripciones. Con 3.200 usuarios, en un período de 1 año, realizando los aportes mínimos (curva azul en [FIGURA 2. 27](#)).

En la siguiente [TABLA 2. 10](#) se observa una aproximación a la cantidad de meses que se necesitan para recuperar la inversión de acuerdo a la cantidad de usuarios que aportan, y al valor de este aporte, ya sea en forma de donación o de suscripción mensual. El verdadero interés está en visualizar la cantidad de usuarios que permiten recuperar la inversión por ejemplo en un año (resaltado en negritas).

<b>RECUPERACIÓN DE INVERSIÓN EN FUNCIÓN DE LA CANTIDAD DE USUARIOS</b>					
	<b>Donación/Suscripción</b>				
	\$30.00	\$50.00	\$75.00	\$100.00	\$150.00
<b>Usuarios que Aportan</b>	<b>Meses</b>	<b>Meses</b>	<b>Meses</b>	<b>Meses</b>	<b>Meses</b>
200	242.62	145.57	97.05	72.79	48.52
400	121.31	72.79	48.52	36.39	24.26
600	80.87	48.52	32.35	24.26	16.17
800	60.65	36.39	24.26	18.20	<b>12.13</b>
1000	48.52	29.11	19.41	14.56	9.70
1200	40.44	24.26	16.17	<b>12.13</b>	8.09
1400	34.66	20.80	13.86	10.40	6.93
1600	30.33	18.20	<b>12.13</b>	9.10	6.07

1800	26.96	16.17	10.78	8.09	5.39
2000	24.26	14.56	9.70	7.28	4.85
2200	22.06	13.23	8.82	6.62	4.41
2400	20.22	<b>12.13</b>	8.09	6.07	4.04
2600	18.66	11.20	7.47	5.60	3.73
2800	17.33	10.40	6.93	5.20	3.47
3000	16.17	9.70	6.47	4.85	3.23
3200	15.16	9.10	6.07	4.55	3.03
3400	14.27	8.56	5.71	4.28	2.85
3600	13.48	8.09	5.39	4.04	2.70
3800	12.77	7.66	5.11	3.83	2.55
4000	<b>12.13</b>	7.28	4.85	3.64	2.43
4200	11.55	6.93	4.62	3.47	2.31
4400	11.03	6.62	4.41	3.31	2.21
4600	10.55	6.33	4.22	3.16	2.11
4800	10.11	6.07	4.04	3.03	2.02
5000	9.70	5.82	3.88	2.91	1.94
5200	9.33	5.60	3.73	2.80	1.87
5400	8.99	5.39	3.59	2.70	1.80
5600	8.66	5.20	3.47	2.60	1.73
5800	8.37	5.02	3.35	2.51	1.67
6000	8.09	4.85	3.23	2.43	1.62

Tabla 2. 10 Meses necesarios para recuperar la inversión en función de los usuarios.

A su vez, se han realizado simulaciones más realistas teniendo en cuenta el total de usuarios de la plataforma en un momento dado, de acuerdo al crecimiento estimado, y la posible popularidad de la misma, el porcentaje mínimo y máximo de usuarios que realizan aportes, y la variabilidad de dichos aportes (valor de donaciones/suscripciones).

Dado lo anterior, podemos estimar la cantidad mínima de usuarios que necesita la plataforma Omics, para recuperar la inversión en 12, 18, 24, y 48 meses.

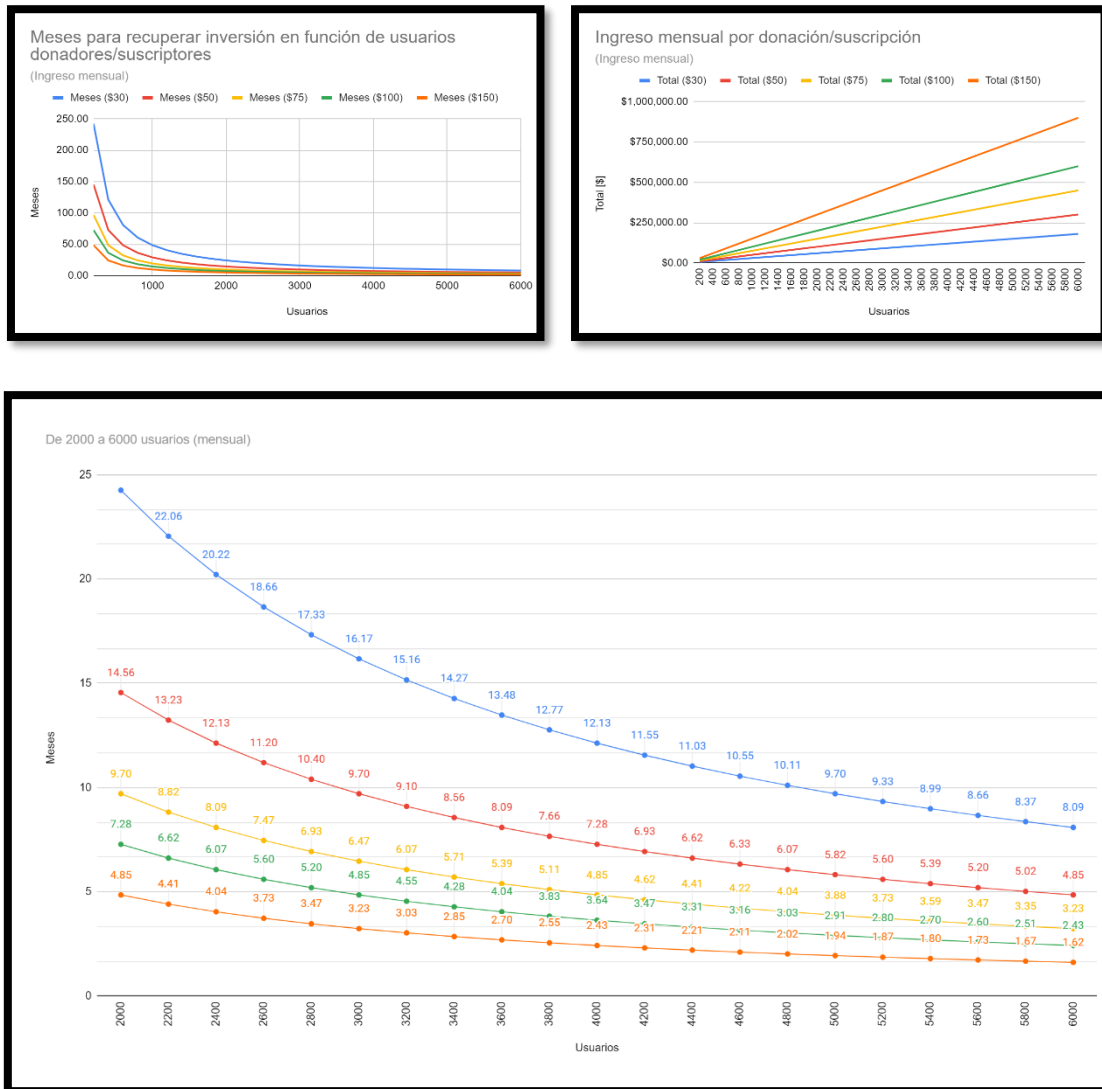


Figura 2. 27 Cantidad de meses en que se recupera la inversión en función del número de usuarios de la plataforma y el monto de la donación/suscripción: \$ 30 (azul), \$50 (rojo), \$75 (amarillo), \$100 (verde), \$150 (naranja).

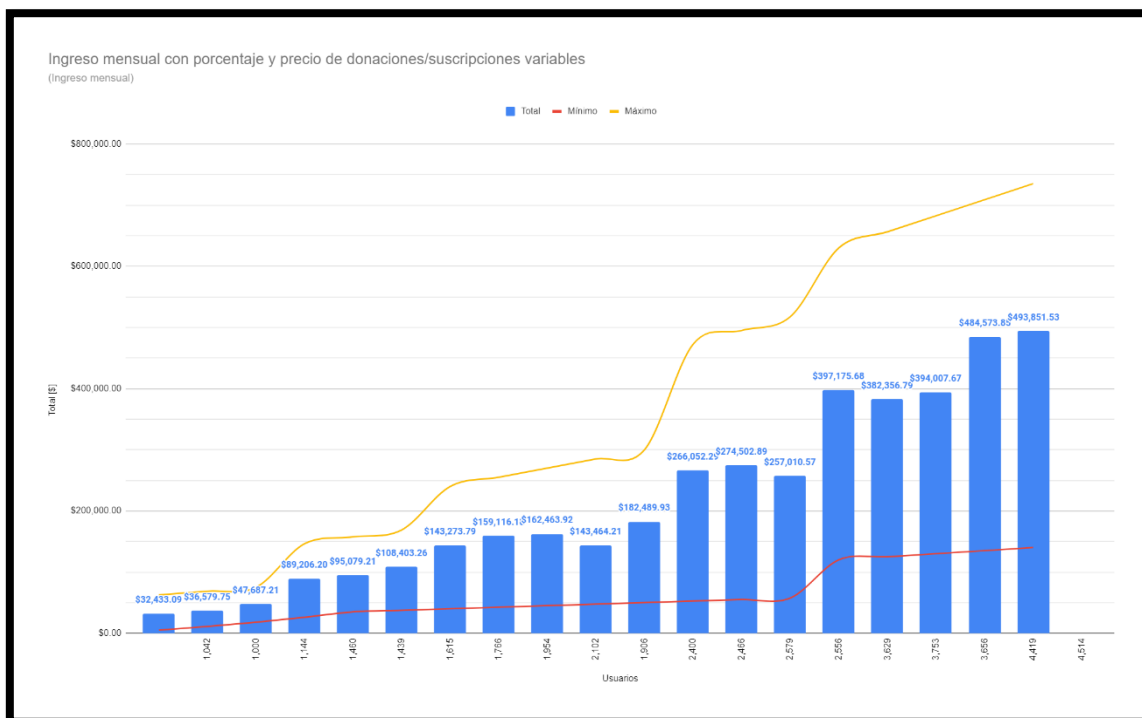


Figura 2. 28 Ingresos mensuales mínimo (rojos), máximos (amarillo) y promedios (azul) teniendo en cuenta la cantidad de usuarios realizando aportes, ya sea por donación o suscripción.

Se han realizado las simulaciones teniendo en cuenta el porcentaje mínimo y máximo de usuarios totales que aportan dinero a la plataforma. También se tuvo en consideración la variabilidad del valor de estos ingresos en forma de donación/suscripción.

Se aprecia el crecimiento lineal, sin embargo, lo importante a destacar es el cambio de un modelo guiado por donaciones a un modelo guiado por suscripciones, donde es imperativo el pago mensual de dicha suscripción.

En una primera instancia, comenzar por recibir un porcentaje de las donaciones, para afianzar una comunidad lo suficientemente grande es una buena idea para luego pasar a un modelo basado en suscripciones de los usuarios, lo que asegura un ingreso mensual a la plataforma.

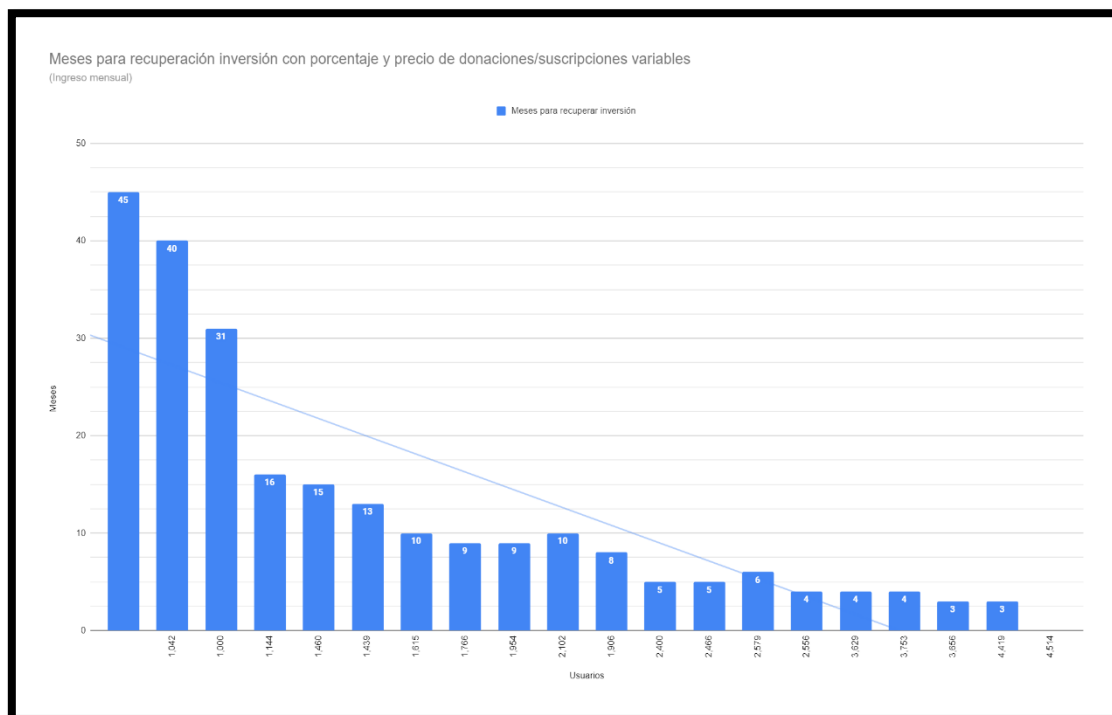


Figura 2. 29 Cantidad de meses necesarios en función del número de usuarios realizando aportes, teniendo en cuenta variabilidad del porcentaje de usuarios y del valor de las donaciones/suscripciones.

La comunidad (cantidad de usuarios) tiende a crecer exponencialmente gracias a las campañas de Marketing para difundir y publicitar la plataforma. difusión, publicidad, además de las recomendaciones de los usuarios a otros. Por esto, según el análisis realizado, podemos concluir que el proyecto es factible económicamente.

### Conclusión:

Con lo que se menciona anteriormente, haciendo referencia a la disposición de capital en efectivo o de los créditos de financiamientos necesarios para invertir en el desarrollo de la plataforma *Omic*s, se llega a la conclusión de que los beneficios a obtener son superiores a los costos realizados al desarrollar e implementar la misma. Con las simulaciones que se han propuesto, teniendo en cuenta el porcentaje mínimo y máximo de usuarios totales que aportan dinero a *Omic*s, y considerando la variabilidad del valor de los ingresos en forma de donación/suscripción, se concluye que el proyecto es factible económicamente. Además, se tiene en consideración para determinar los costos a futuro, la recesión económica y la inflación de Argentina.

Se han incluido análisis de los costos y beneficios asociados con cada alternativa del proyecto para hacer una comparación entre ellos e identificar los casos. Primero se han comparado los costos esperados de cada alternativa con los beneficios esperados

para poder asegurarse de que los beneficios excedan a los costos. Después la proporción costo/beneficio de cada alternativa se compara con las que proporcionan costo/beneficio de las demás opciones para visualizar la que sea más atractiva en su aspecto económico. Además, se han observado los recursos que se necesitan para implementar el proyecto y una evaluación de las posibilidades de éxito.

### 3.4. Factibilidad Operativa

#### 3.4.1. Insumos

En el siguiente cuadro se puede observar cómo se detallan los recursos, tanto hardware como software, con sus respectivas características que son necesarias para llevar a cabo el desarrollo de la plataforma *Omics*.

Hardware	Características
<b>PC</b>	
1	Intel i5 8va generación, 16 GB Ram DDR4, Nvidia Gforce 2070RTX Asus Dual OC, 1TB HDD, 256GB SDD (M2), Sistema operativo Windows 10
1	APU A6 con placa de vídeo integrada, 8 GB de RAM, 500 GB de disco estado sólido, 1 TB disco duro, lecto grabador. Sistema operativo Windows 10
1	Procesador: i7 7700 (Intel) · Memoria: 16 GB de RAM Almacenamiento: 2 TB de disco duro, 256 GB SSD M.2 Placa de vídeo: Asus GTX 1700, 8 GB de memoria de video · Sistema operativo: Arch Linux
<b>Total de PCs</b>	<b>3</b>
<b>Monitor</b>	
2	1 monitor de 19 pulgadas - Samsung 1 monitor de 26 pulgadas - Sentey
1	monitor de 24 pulgadas - LG
3	3 monitores widescreen de 23 pulgadas - Samsung



<b>Total de Monitores</b>	<b>6</b>
<b>Teclado</b>	
1	Inalámbrico, semi mecánico - Logitech
1	Inalámbrico, mecánico - Logitech
1	Inalámbrico, mecánico - Logitech
<b>Total de Teclados</b>	<b>3</b>
<b>Mouse</b>	
3	Inalámbricos ergonómicos - Logitech
1	Inalámbrico ergonómico - Genius
1	Alámbrico ergonómico - Razer
<b>Total de Mouses</b>	<b>5</b>
<b>Notebook</b>	
1	Lenovo, Yoga 520 Core i5 7th Gen
1	Toshiba, Satellite L845 Core i7 6 GB RAM SO Windows 10 Home Single
1	Notebook Lenovo ROG - Procesador Intel i5 7ma generación, 8GB Ram ddr4, Nvidia Gforce 1070, 1 tera HDD, 128GB SDD(M2), SO Windows 10 Professional
1	Asus k53t
<b>Total de Notebooks</b>	<b>4</b>

<p>Servidor local para entorno de desarrollo</p>	<p>Se utiliza Docker como tecnología de contenedores, lo que permite a cada integrante del equipo levantar el entorno en su propia máquina sin conflictos.</p> <p>También, se tiene un servidor dedicado corriendo distintas instancias del servidor, en diferentes etapas de desarrollo para pruebas.</p>
<p><b>Total de Servidores</b></p>	<p><b>1</b></p>

Tabla 2. 11 Equipo disponible para operar.

### 3.4.2. Capacitación del equipo

Para realizar la capacitación de los miembros del equipo se tiene en cuenta, en base al proyecto a desarrollar, las tecnologías utilizadas en la etapa de desarrollo de la plataforma.

A continuación, se definen las tecnologías, y el nivel del equipo:

Tecnologías	Nivel
<b>Front-end</b>	
Angular. Typescript como lenguaje tipado.	Intermedio
<b>Back-end</b>	
RUST	Intermedio
PostgreSQL: Bases de datos.	Intermedio
Redis: Caché, base de datos clave – valor.	Bajo
RabbitMQ: Comunicación asíncrona.	Bajo
Docker: Contenedores, infraestructura, y soporte.	Intermedio
<b>Integración y Soporte</b>	
GIT + GitHub: Repositorios de código.	Avanzado
CircleCI: Integración continua,	Bajo

automatización de builds, deployment.	
Swagger: documentación de API.	Intermedio
<b>Servidores</b>	
VPS (Virtual Private Server): Servidor Linux, administración de servidor.	Intermedio
Amazon S3: Almacenamiento de archivos.	Bajo
<b>Servicios Externos</b>	
Mercado Pago API: Donación y Suscripción a autores.	Intermedio

Tabla 2. 12 Nivel del equipo en cada tecnología.

**Niveles:**

- ❖ Bajo = Requiere que los miembros del equipo se capaciten.
- ❖ Intermedio = Requiere que los miembros del equipo profundicen los conocimientos que ya poseen.
- ❖ Avanzado = Los miembros del equipo ya cuentan con el conocimiento necesario.

**3.4.3. Resolución de conflictos**

Cuando surge un conflicto lo primero a analizar es el motivo, ya que al trabajar en un equipo ágil donde las entregas son continuas y los tiempos acotados, los conflictos son inevitables. Por lo tanto, lo mejor es aceptarlos, solucionarlos para poder avanzar en el desarrollo del proyecto y aprender de la experiencia para no repetir los mismos errores que desembocaron en un conflicto. Además, hay que ver el conflicto como una oportunidad para adaptarnos mejor y crecer como equipo.

Existen herramientas, métodos y gran cantidad de material que aborda la solución de conflictos en equipos.

Se decide utilizar en caso de conflictos lo siguiente:

- ❖ Resolverlo de forma personal, ya que todos los miembros del equipo son responsables. Mediante un café de por medio, en un ambiente agradable, se conversa sobre cuál/es fueron los motivos que llevaron a un conflicto.
- ❖ Si una reunión informal y distendida no logró solucionar el conflicto, se recurre a la contratación de un Agile Coach.

El papel del Agile Coach es enseñar al equipo herramientas, prácticas y técnicas que les permita a los integrantes evitar la aparición de conflictos negativos. En el caso de que surjan, poder abordarlos de una manera constructiva y positiva, viendo los conflictos como una oportunidad para mejorar la interrelación entre los miembros del equipo.

#### **3.4.4. Capacitación usuarios**

Para realizar la capacitación de los usuarios finales, se genera una comunidad de interesados en los cómics. Para ello se utilizan las redes sociales y la visita a eventos de aficionados al manga, cómics, gamers, etcétera.

La plataforma es intuitiva y amigable con el usuario, por lo que no es necesario elaborar un plan de capacitación formal.

#### **Conclusión:**

En base al análisis de factibilidad realizado, se puede concluir que la plataforma *Omics* es factible operativamente, ya que el equipo cuenta con todas las herramientas para lograr la implementación en término de insumos informáticos.

También se cuenta con la capacitación necesaria para el desarrollo de todas las tareas pertinentes a programación, aplicación de servicios, servidores, entre otros. Se aprecia una mejora en la eficiencia de las actividades propuestas para la plataforma, teniendo así, altos índices de productividad. Esto permite mantener a todas las personas involucradas en la evolución de las tareas, como así también, ayuda a tener mayor comprensión de las responsabilidades del trabajo de cada miembro.

En cuanto a la resolución de conflictos entre los miembros del equipo, es bastante eficaz la utilización del método ágil, ya que permite una guía sobre problemas que puedan surgir en el desarrollo de las actividades, proponiendo diálogos abiertos en cierres de Sprint para solucionar los inconvenientes del proceso.

En caso de que se requiera una resolución rápida sobre conflictos, que impiden el desarrollo normal de la actividad principal, se opta por las reuniones diarias (daily scrum meetings) que, en el corto plazo, son aditamentos funcionales para la organización.

#### **Conclusión general:**

Luego del análisis anterior de factibilidad en los cuatros frentes más importantes y relevantes para el proyecto (técnico, legal, económico y operativo) se concluye que el proyecto de la plataforma *Omics*, es factible de acuerdo con estos.

Se pone especial énfasis en los dos primeros, debido a que es importante evaluar la factibilidad del proyecto en términos técnicos y, sobre todo, legales.

*Omics* ofrece un servicio de distribución de contenido, por lo que sus bases y la arquitectura sobre la que subyace la plataforma son de importante consideración. Dicho de otra forma, el componente técnico, las capacidades de procesamiento, de almacenamiento, la disponibilidad del servicio, la recuperación en caso de errores o fallas, y la satisfacción de requerimientos del usuario en tiempo y espacio determinan el éxito de la plataforma como producto y servicio. Así es, que el tratamiento de imágenes y la escalabilidad de la arquitectura son retos técnicos para los involucrados en el diseño, implementación y mantenimiento de la plataforma.

Más allá del aspecto técnico del proyecto, el componente legal es aún más influyente en el éxito de la plataforma. La distribución de contenido, del cual la organización no es completamente dueña, sino más bien intermediaria, puede suponer dificultades legales si algún autor queda en disconformidad con el servicio que *Omics*, como intermediario le brinda.

Tanto técnica como legalmente el proyecto es factible, para esto se considera tecnología de punta y un diseño, acompañado de una adecuada implementación, de una arquitectura escalable que contempla futuros incrementos en requerimientos de carga (velocidad y espacio), teniendo siempre en consideración la recuperación frente a fallas.

Por último, en términos legales, el proyecto no presenta falencias siempre que se le requiera a los distintos usuarios, lectores y autores, que confirmen estar de acuerdo con cada política involucrada: cookies, términos y condiciones, contrato del autor para con la plataforma.

Es necesario proteger a la organización frente a cualquier complicación legal que pudiera surgir. Para ello se ha establecido el requisito de aceptación de acuerdos por parte de los autores.

Finalmente, del aspecto económico analizado en función de múltiples variables, desde el peor al mejor de los escenarios, se puede decir que es factible económicamente, tanto para el proyecto, como para la organización, donde se auto sustentan, así también como para recuperar la inversión inicial en un plazo máximo de 2 años.

Por lo tanto, se concluye que el proyecto es **factible** en su totalidad.

#### 4. Costos desagregados por recursos (personal, tecnología) con periodicidad mensual

RECURSOS				
	Recursos	Costo Total	Subtotal	Proveedor
Servidores	VPS (2 GB, 1 CPU, 2 TB, 50 GB)	\$5,603.92	\$9,142.49	DigitalOcean
	VPS (1 GB, 1 CPU, 1 TB, 25 GB)	\$2,941.82		DigitalOcean
	Servidor de almacenamiento (50 GB)	\$596.75		Amazon: S3
Personal	Product Manager	\$259,095.00	\$1,429,564.50	
	Arquitecto de Software	\$343,735.00		
	Scrum Master	\$308,522.50		
	Desarrollador back-end Sr.	\$205,327.50		
	Desarrollador front-end Sr.	\$213,664.50		
	Tester Jr.	\$99,220.00		
Servicios	Marketing (publicidad, Community manager)	\$9,000.00	\$9,000.00	Facebook ADS
Subcontratación	Asesoría legal	\$4,000.00	\$8,000.00	
	Servicios contables	\$4,000.00		
<b>Total</b>		<b>\$1,455,706.99</b>		

Tabla 2. 13 Costos totales del proyecto.

Recursos/Servicios	Marzo	Abril	Mayo	Junio	Julio
<b>Personal</b>	\$65368,75	\$275231,25	\$90532,50	\$237892,5	\$303072,5
<b>Equipo</b>			\$1300	\$1300	\$1300
<b>Servicios</b>					
<b>Otros</b>					

Tabla 2. 14 Costos desagregados por mes desde Marzo a Julio.

Recursos/Servicios	Agosto	Septiembre	Octubre	Noviembre
<b>Personal</b>	\$122571	\$93652	\$158309	\$82935
<b>Equipo</b>	\$1300	\$1300	\$1300	\$1300
<b>Servicios</b>			\$4500	\$4500
<b>Otros</b>				\$8000

Tabla 2. 15 Costos desagregados por mes desde Agosto a Noviembre.

Recurso	Costo
Personal	\$1,429,564.50
Equipos	\$9142.49
Servicios	\$9000
Otros	\$8000
<b>Total</b>	<b>\$1,455,706.99</b>

Tabla 2. 16 Costo total.

## 5. Análisis de Riesgos

En este apartado se estudian las causas de las posibles amenazas y probables eventos no deseados que se pueden encontrar, así como los daños y consecuencias que éstas pueden producir. Se trata de un proceso sistemático que planifica, identifica, analiza, responde y controla los riesgos de un proyecto.

Las amenazas son actos que pueden afectar negativamente a cualquiera de los elementos del proyecto. Si en estos elementos existe una vulnerabilidad, es el momento en el que la probabilidad de que se ponga en riesgo el proyecto sea más elevada.

Sin embargo, antes de enfrentarnos a situaciones complicadas, se va a realizar el análisis de riesgo enumerando los riesgos que pueden aparecer en el proyecto. Luego se determina la Probabilidad de Ocurrencia (PO) y el impacto que producirá. Una vez que se obtengan estos datos, se procede a calcular la Pérdida Esperada (PE) o Riesgo.

<b>Probabilidad de Ocurrencia</b>
5= Muy alta
4= Alta
3= Intermedia
2= Baja
1= Muy baja

<b>Impacto</b>
5= Muy grave
4= Grave
3= Moderado
2= Leve
1= Muy Leve

<b>Riesgo</b>	<b>Probabilidad de Ocurrencia</b>	<b>Impacto</b>	<b>Pérdida Esperada= PO*Impacto</b>
Error en la planificación.	3	3	9
El equipo pierde interés en el proyecto.	2	2	4
Integrantes del equipo con actitudes negativas.	2	3	6
Un integrante del equipo abandona el proyecto.	1	3	3
Los cambios no son			



controlados y el alcance está en continuo crecimiento.	2	3	6
Un integrante del equipo agrega sus propias características al sistema y éstas no son requerimientos.	2	3	6
Faltan actividades necesarias en la definición del alcance.	3	2	6
Las estimaciones y pronósticos de costo son inexactos.	3	2	6
Falta de identificación de cualquier cambio.	2	3	6
Cuando los requisitos son mal interpretados por el equipo.	2	4	8
Debilidades de los miembros del equipo.	3	2	6
La arquitectura carece de flexibilidad.	2	3	6
La comunicación entre el equipo no funciona bien.	2	5	10
Los componentes técnicos no son adecuados.	2	4	8
Los componentes técnicos tienen vulnerabilidades de seguridad.	3	2	8
Los entornos de prueba e integración no están disponibles.	2	3	6
Incapacidad para integrar componentes.	2	3	6
Los requerimientos están incompletos.	3	2	8

Las decisiones no son las adecuadas para el propósito del proyecto.	2	4	8
La falta de autoridad en el equipo para completar el trabajo y lograr los objetivos.	2	3	6
Capacitación de calidad no disponible.	3	2	6
Error en la definición de los requerimientos.	3	5	15
Falta de conocimiento en la tecnología que hay que utilizar.	3	4	12
Incendio.	1	5	5
Falta de conocimiento en el negocio.	4	2	12
Pérdida de la propiedad intelectual.	1	3	3
Corte de luz.	2	1	2
Manejo incorrecto de la tecnología.	3	3	9
Corte del servicio de internet.	4	2	8
Error en las pruebas.	3	2	6
Interfaz de usuario inadecuada requiere volver a diseñar e implementar.	2	3	6
Funciones de SW inadecuadas requiere volver a diseñarlas e implementarlas.	3	4	12
Falta de tiempo para acostumbrarse a trabajar con herramientas o entornos nuevos.	3	3	9

Pérdida de información importante.	2	5	10
------------------------------------	---	---	----

Tabla 2. 17 Tabla de Riesgos.

### 5.1. Tabla de Riesgos:

- Si el riesgo es menor a 10, entonces No debe ser tratado.
- Si el riesgo es mayor a 10, entonces Si debe ser tratado.

<b>RANKING DE RIESGOS</b>		
<b>Número</b>	<b>Riesgo</b>	<b>PE</b>
1	Error en la definición de los requerimientos.	15
2	Falta de conocimiento en la tecnología que hay que utilizar.	12
2	Falta de conocimiento en el negocio.	12
2	Funciones de software inadecuadas requiere volver a diseñarlas e implementarlas.	12
3	Pérdida de información importante.	10
3	La comunicación entre el equipo no funciona bien.	10

Tabla 2. 18 Ranking de Riesgos.

#### 5.1.1. Tratamiento de Riesgos:

<b>Riesgo</b>	<b>Medida Preventiva</b>
<b>Error en la definición de los requerimientos.</b>	<ul style="list-style-type: none"> <li>❖ Entrenar a los trabajadores en procedimientos para detectar requerimientos.</li> <li>❖ Desarrollar análisis de Seguridad de la Tarea y procedimientos seguros de trabajo.</li> </ul>
<b>Falta de conocimiento en la tecnología que hay que utilizar.</b>	<ul style="list-style-type: none"> <li>❖ Organizar y llevar a cabo capacitaciones referidas a las tecnologías que se van a usar en el proyecto. La persona con más habilidad en la tecnología puede capacitar al miembro del equipo que lo</li> </ul>

	necesite. También se puede contar con la compra de cursos webs.
<b>Falta de conocimiento en el negocio.</b>	<ul style="list-style-type: none"> <li>❖ Realizar un estudio detallado e investigación para conocer mejor el negocio. Se tienen contactos que pueden brindar asesoramiento más detallado en la industria que se plantea para utilizar en el proyecto.</li> </ul>
<b>Funciones de software inadecuadas requiere volver a diseñarlas e implementarlas.</b>	<ul style="list-style-type: none"> <li>❖ Llevar un control coordinado del desarrollo de funciones, replanteo de alguna librería o tecnología a usar y una vez que finalice la función realizada por alguno de los integrantes, se pasa a otro del equipo para que lo revise.</li> <li>❖ Realizar <i>refactoring</i> en las partes críticas del sistema regularmente.</li> </ul>
<b>Pérdida de información importante.</b>	<ul style="list-style-type: none"> <li>❖ Hacer un backup o copia de seguridad de la información importante que se vaya utilizando, ya que hace una copia de los datos originales pudiendo recuperar el sistema y restaurar archivos.</li> <li>❖ Evitar la pérdida utilizando Ondata Disk Control que es un software que ofrece monitorizar el estado del disco duro. Con este programa se busca detectar posibles fallos, de forma que se puedan tomar acciones antes de que se produzcan daños irreparables con la información.</li> <li>❖ Almacenar la información en la nube para poder acceder a la misma desde cualquier lugar o dispositivo y es una buena opción para tener un respaldo de los datos sensibles.</li> <li>❖ Configurar de manera correcta los cortafuegos, aislando a la red de la compañía de los accesos no autorizados.</li> </ul>
<b>La comunicación entre el equipo no</b>	<ul style="list-style-type: none"> <li>❖ Definir de manera clara los canales de</li> </ul>

<p><b><i>funciona bien.</i></b></p>	<p>comunicación para el equipo.</p> <ul style="list-style-type: none"> <li>❖ Establecer más de un canal de comunicación con diferentes propósitos para ordenar la información que se maneja.</li> <li>❖ El modo de comunicación debe ser claro, preciso y con respeto para que no se generen confusiones.</li> </ul>
-------------------------------------	--

Tabla 2. 19 Análisis de riesgos-Medida Preventiva.

Riesgo	Medida Correctiva
<p><b><i>Error en la definición de los requerimientos.</i></b></p>	<ul style="list-style-type: none"> <li>❖ Revisar el requerimiento que falló y analizarlo nuevamente para detectar el problema desde otros puntos de vista junto a asesores externos que faciliten la información más detallada.</li> <li>❖ Si es necesario, cambiar procedimientos para la detección de requerimientos.</li> <li>❖ Mejorar la instrucción en el trabajo.</li> </ul>
<p><b><i>Falta de conocimiento en la tecnología que hay que utilizar.</i></b></p>	<ul style="list-style-type: none"> <li>❖ Establecer horarios extra para capacitar a los integrantes del equipo.</li> <li>❖ Realizar entrega de material importante para conocer la tecnología con que se tenía que trabajar.</li> <li>❖ Contratación de consultores externos en caso de extrema necesidad de adaptación de la tecnología.</li> </ul>
<p><b><i>Falta de conocimiento en el negocio.</i></b></p>	<ul style="list-style-type: none"> <li>❖ Establecer un procedimiento de comunicaciones para que los trabajadores informen a los supervisores de problemas con algún procedimiento.</li> </ul>
<p><b><i>Funciones de sw inadecuadas requiere volver a diseñarlas e implementarlas.</i></b></p>	<ul style="list-style-type: none"> <li>❖ Inducir a los trabajadores a informar potenciales riesgos creados por el diseño del equipo.</li> <li>❖ Revisar los criterios de diseño o calidad a través de códigos, estándares, especificaciones y reglamentaciones.</li> <li>❖ Realizar reuniones extraordinarias para solucionar problemas de diseño e implementación.</li> </ul>

<p><b><i>Pérdida de información importante.</i></b></p>	<ul style="list-style-type: none"> <li>❖ Recopilar nuevamente la información utilizando los backups.</li> <li>❖ Detectar la causa y realizar un seguimiento de lo ocurrido para que no vuelva a suceder.</li> <li>❖ Realizar un control de las acciones y procedimientos para hacer mejoras y garantizar la seguridad de la información.</li> <li>❖ Aplicar actividades como constante monitoreo, valoración, comunicación, entre otros.</li> </ul>
<p><b><i>La comunicación entre el equipo no funciona bien.</i></b></p>	<ul style="list-style-type: none"> <li>❖ Detectar y analizar la causa que originó una confusión o mal entendido en la comunicación para evitar que vuelva a producirse en un futuro.</li> <li>❖ Llegar a un acuerdo o consenso entre los integrantes del equipo para establecer mejoras en los canales de comunicación.</li> <li>❖ Una reunión presencial entre todo el equipo para buscar alternativas en el caso que consideren que un canal específico no funciona.</li> </ul>

*Tabla 2. 20 Análisis de riesgos-Medida Correctiva.*

### **Conclusión:**

De acuerdo con el análisis que se ha realizado, se puede contemplar medidas de prevención para los riesgos que se enumeraron anteriormente. Esto ayuda a identificar varios aspectos que no se han tenido en cuenta y evitar contratiempos. Se utilizan para anticiparse a la causa, y pretenden eliminarla antes de su existencia. Evitan los problemas identificando los riesgos. Cualquier acción que disminuya un riesgo es una acción preventiva.

También se analizaron las medidas correctivas que proponen mejoras como consecuencia de haber estudiado la causa de una no conformidad detectada en la organización. Se trata de una solución que ha sido evaluada y previo a establecer las acciones correctivas se lleva a cabo una acción inmediata que elimine a corto plazo esa situación de incumplimiento que se ha detectado.

## 6. Análisis de Impacto Ambiental

Para el análisis de impacto ambiental se utiliza el método de análisis matricial, el cual se construye a partir de una matriz con dos componentes principales:

- ❖ **Acciones del proyecto.**
- ❖ **Variables Ambientales o Componentes Ambientales.**

Dentro de una Acción se va a tener en cuenta los siguientes ítems:

- **Signo:** Si es positivo y sirve para mejorar el medio ambiente o si es negativo y degrada.
- **Magnitud:** Según la destrucción del ambiente sea total, alta, media o baja.
- **Alcance:** Global, local o restringido.
- **Persistencia:** Se dice que es fugaz si dura menos de 1 año; si dura de 1 a 3 años es temporal y si dura de 4 a 10 años es pertinaz. Si es para siempre es permanente.

A continuación, se listan las acciones que producen un impacto en el medio ambiente y se presenta una tabla con los impactos que se han considerado más importantes:

### Acción: Plataforma “Omics”

El sistema está desarrollado para los lectores que quieran adentrarse en el mundo de los cómics, los cuales tendrán acceso a los ejemplares que ofrece la plataforma virtual, reduciendo de esta manera el uso de papel y tinta. Esto permite que se pueda reciclar el papel ya empleado para los cómics y utilizarlo para otro motivo.

Menos papel significa menos gases tóxicos y una importante reducción de desechos. La distribución por algún medio de transporte sería aún menor ya que los cómics estarían alojados en la plataforma, y los desechos generados por no utilizar papel, producen un beneficio ecológico.

La aplicación está alojada en un servidor, el cual tiene disponibilidad de 24 horas todos los días, lo que conlleva un importante consumo de energía eléctrica.

Componente Ambiental	Signo	Magnitud	Alcance	Persistencia
<b>Reducción de uso de papel y tinta.</b>	+	Alta	Global	Permanente

<b>Consumo energético.</b>	-	Media	Local	Permanente
<b>Reducción de gases tóxicos debido a la no distribución de cómics de manera física.</b>	+	Media	Global	Permanente
<b>Reducción de desechos por bajo consumo de cómics de manera física.</b>	+	Media	Global	Permanente
<b>Impacto social sobre los lectores.</b>	+	Media	Local	Permanente

Tabla 2. 21 Impacto ambiental - Acción: Plataforma "Omics".

**Acción: Módulo publicación de obras**

Este módulo se basa en que los autores puedan subir sus obras a la plataforma de manera virtual para su visualización, reduciendo el uso primario de papel y tinta en su posterior duplicación. Este impacto es significativo también en la tala de árboles y en el reciclaje.

Componente Ambiental	Signo	Magnitud	Alcance	Persistencia
<b>Reducción de uso de papel y tinta.</b>	+	Alta	Global	Permanente
<b>Consumo energético.</b>	-	Media	Local	Permanente
<b>Impacto social sobre los autores.</b>	+	Media	Local	Permanente

Tabla 2. 22 Impacto ambiental - Acción: Módulo publicación de obras.

**Factor negativo de gran impacto al proyecto**

El consumo energético que se menciona anteriormente es un factor que todo sistema basado en tecnología dependiente de electricidad moderno tiene, pero existe la gran posibilidad que en algunas regiones haya cortes importantes de corriente que impidan el acceso (sea dispositivo móvil o PC) a la plataforma. Es un riesgo muy pequeño y generalmente momentáneo, pero de perdurar mucho en el tiempo podría afectarnos negativamente en algunas zonas.



Por esto, el factor mencionado que podría imponer un riesgo crítico para el proyecto ha sido considerado en el análisis de factibilidad técnico y mitigado correctamente: no se utilizan servidores propios para servir los datos, el servicio se contrata a empresas proveedoras de servidores privados virtuales que aseguran una disponibilidad del 100% del tiempo. A pesar de esto, también se consideran servidores de respaldo y balanceadores de carga que, de ser necesario, permiten distribuir las consultas a servidores establecidos en distintas localizaciones geográficas.

Entonces, un corte importante de corriente sólo supone un riesgo mínimo en términos de usuarios afectados que no puedan acceder a la plataforma en un momento determinado. Omics se piensa como una plataforma portátil para el usuario, lo que no supondría un problema al ser utilizada y accedida desde dispositivos móviles que hacen uso de baterías.

El consumo energético, por ende, es significativo, pero se logra reducir al contratar servicios externos para mantener los servidores virtuales. Estos corren sobre una infraestructura que reduce el consumo energético al mínimo para ahorrar recursos y a su vez compartirlos entre múltiples clientes.

## **Conclusión:**

Analizando las variables estipuladas en los cuadros anteriores, se puede concluir que el impacto ambiental generado resulta positivo de acuerdo a los signos expuestos.

El factor que resultó negativo es el consumo energético, debido a la utilización de los servidores y los gastos que demandan.

# TRABAJO PRÁCTICO

# INTEGRADOR N°1:

## *Dirección de Proyectos Informáticos*

## 1. Riesgos del proyecto

A continuación se enumeran los 10 principales riesgos que pueden llegar a aparecer en el proyecto, cuáles son sus consecuencias y qué impacto tendrían esas consecuencias. Además, se detallan las medidas preventivas para cada uno de los riesgos.

<b>Probabilidad de Ocurrencia (PO)</b>	<b>Impacto (I)</b>
5= Muy alta	5= Muy grave
4= Alta	4= Grave
3= Intermedia	3= Moderado
2= Baja	2= Leve
1= Muy baja	1= Muy Leve

Tabla 3. 1 Tabla de referencia de Probabilidad de ocurrencia e impacto de riesgo.

<b>Nº</b>	<b>Riesgo</b>	<b>PO</b>	<b>Impacto</b>	<b>Pérdida Esperada= PO*Impacto</b>
1	Error en la definición de los requerimientos.	3	5	15
2	Falta de conocimiento en la tecnología que hay que utilizar.	3	4	12
3	Falta de conocimiento en el negocio.	4	3	12
4	Funciones de sw inadecuadas requiere volver a diseñarlas e implementarlas.	3	4	12
5	Pérdida de información importante.	2	5	10
6	La comunicación entre el equipo no funciona bien.	2	5	10
7	Error en la planificación.	3	3	9

8	Manejo incorrecto de la tecnología.	3	3	9
9	Los requerimientos están incompletos.	4	2	8
10	Las decisiones no son las adecuadas para el propósito del proyecto.	2	4	8

Tabla 3. 2 Tabla de ponderación riesgos (de mayor a menor consecuencia).

N°	Riesgo	Consecuencia	Medida Preventiva
1	Error en la definición de los requerimientos.	<ul style="list-style-type: none"> <li>❖ Pérdida de tiempo y dinero invertido.</li> <li>❖ Corregir las fallas de la definición podrían volver inviable el proyecto.</li> <li>❖ Frustración en el equipo que podría llegar a hacer que abandonen sus ánimos con seguir el proyecto.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Entrenar a los trabajadores en procedimientos para detectar requerimientos.</li> <li>❖ Desarrollar análisis de Seguridad de la Tarea y procedimientos seguros de trabajo.</li> </ul>
2	Falta de conocimiento en la tecnología que hay que utilizar.	<ul style="list-style-type: none"> <li>❖ Retrasos de tiempos de entrega.</li> <li>❖ Desmotivación de la persona que debe utilizar la tecnología para la tarea que se le encargó.</li> <li>❖ Pérdida de competitividad.</li> <li>❖ Podría acarrear errores muy difíciles de corregir a corto plazo.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Organizar y llevar a cabo capacitaciones referidas a las tecnologías que vamos a usar en el proyecto. La persona con más habilidad en la tecnología puede capacitar al miembro del equipo que lo necesite.</li> <li>❖ También se puede contar con la compra de cursos webs.</li> </ul>

3	Falta de conocimiento en el negocio.	<ul style="list-style-type: none"> <li>❖ Error en la definición de algunos requerimientos.</li> <li>❖ No comunicarse correctamente con los clientes.</li> <li>❖ Pérdida de competitividad frente a otras organizaciones con mayor conocimiento de la industria.</li> <li>❖ Errores en la planificación.</li> <li>❖ No se genera una comunidad alrededor del proyecto que de soporte al mismo.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Se realizará un estudio detallado e investigación para conocer mejor el negocio.</li> <li>❖ Se utiliza asesoramiento más detallado en la industria que planteamos para utilizar en el proyecto.</li> </ul>
4	Funciones de sw inadecuadas requiere volver a diseñarlas e implementarlas.	<ul style="list-style-type: none"> <li>❖ Atraso en la planificación.</li> <li>❖ Frustración de algunos miembros de otras áreas que requerían la funcionalidad.</li> <li>❖ Reorganización de los tiempos de entrega.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Se llevará un control coordinado del desarrollo de funciones, replanteo de alguna librería o tecnología a usar y una vez que finalice la función realizada por alguno de los integrantes se pasará a otro del equipo para que lo revise.</li> <li>❖ Realizar <i>refactoring</i> en las partes críticas del sistema regularmente.</li> </ul>
5	Pérdida de información importante.	<ul style="list-style-type: none"> <li>❖ Atraso en los tiempos de entrega.</li> <li>❖ Desmotivación de miembros del equipo.</li> <li>❖ Clientes frustrados.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Hacer un backup o copia de seguridad de la información importante que se vaya utilizando, ya que hace una copia de los datos</li> </ul>

		<ul style="list-style-type: none"> <li>❖ Circulación de información confidencial.</li> </ul>	<p>originales pudiendo recuperar el sistema y restaurar archivos.</p> <ul style="list-style-type: none"> <li>❖ Evitar la pérdida utilizando Ondata Disk Control que es un software que ofrece monitorizar el estado del disco duro. Con este programa se busca detectar posibles fallos, de forma que se puedan tomar acciones antes de que se produzcan daños irreparables con la información.</li> <li>❖ Almacenar la información en la nube para poder acceder a la misma desde cualquier lugar o dispositivo y es una buena opción para tener un respaldo de los datos sensibles.</li> <li>❖ Configurar de manera correcta los cortafuegos, aislando a la red de la compañía de los accesos no autorizados.</li> </ul>
<p>6</p>	<p>La comunicación entre el equipo no funciona bien.</p>	<ul style="list-style-type: none"> <li>❖ Frustración de todo el equipo porque afecta a la toma de decisiones sobre el proyecto.</li> <li>❖ Pérdida de tiempo si surgen confusiones y se hizo algo mal.</li> <li>❖ Malestar en el ambiente de trabajo.</li> <li>❖ Deterioro de la eficiencia.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Definir de manera clara los canales de comunicación para el equipo.</li> <li>❖ Establecer más de un canal de comunicación con diferentes propósitos para ordenar la información que se maneja.</li> <li>❖ El modo de comunicación debe ser</li> </ul>

		<ul style="list-style-type: none"> <li>❖ Sentimientos negativos.</li> <li>❖ Desconfianza.</li> </ul>	<p>claro, preciso y con respeto para que no se generen confusiones.</p>
7	<p>Error en la planificación.</p>	<ul style="list-style-type: none"> <li>❖ Atrasos de entrega de tareas.</li> <li>❖ Tener que replanificar muchos puntos del proyecto.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Se llevan controles semanales para corregir a tiempo las metas tanto a corto como largo plazo del proyecto.</li> <li>❖ Corroborar que el equipo de desarrollo entienda la planificación y metas a lograr.</li> </ul>
8	<p>Manejo incorrecto de la tecnología.</p>	<ul style="list-style-type: none"> <li>❖ Puede llevar a errores en el proyecto que lleven a perder tiempo esencial para resolverlos.</li> <li>❖ Tener que rehacer algunas funcionalidades del proyecto.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Los desarrolladores encargados de cada área deben acompañar el uso de herramienta los desarrolladores más novatos.</li> <li>❖ Los conocedores de las herramientas de gestión y control realizan talleres sobre las funcionalidades y beneficios de la tecnología usada.</li> <li>❖ Se realiza una cultura en la organización para el uso correcto de la herramienta y ofrecer cursos para su mejor desempeño.</li> </ul>
9	<p>Los requerimientos están incompletos.</p>	<ul style="list-style-type: none"> <li>❖ Problemas de comunicación en el equipo.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Realizarse reuniones semanales donde el Product Owner y Jefe de Proyecto revisen los</li> </ul>

		<ul style="list-style-type: none"> <li>❖ Problemas de comunicación con los clientes.</li> <li>❖ Atraso en los tiempos de entrega al tener que realizarlos nuevamente para completar lo faltante del requerimiento.</li> </ul>	<p>requerimientos con lo planificado, para detallarlos de hacer falta.</p> <ul style="list-style-type: none"> <li>❖ Consultar a los involucrados en el proyecto si están claros los requerimientos específicos que se tienen.</li> </ul>
10	Las decisiones no son las adecuadas para el propósito del proyecto.	<ul style="list-style-type: none"> <li>❖ Atraso de tareas.</li> <li>❖ Problemas de comunicación con los clientes.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Verificar con quién sea encargado del sector que afecta la decisión antes de llevarla a cabo.</li> <li>❖ Tener en claro y transmitir la visión y la misión del proyecto al equipo.</li> </ul>

Tabla 3. 3 Tabla de consecuencia y tratamiento de riesgo.

## 2. Conversión e implementación del sistema

La gestión de la implementación del sistema representa todo un reto, ya que de acuerdo a las características del sistema como plataforma global, requiere de un planteo poco tradicional, más bien debe tener en cuenta múltiples factores. Así es que insertar el sistema en la comunidad de usuarios, aquellos que obtengan beneficio del uso del mismo, no representa una tarea sencilla.

En primer lugar, no se posee control absoluto sobre el proceso de inserción y aceptación, por parte del usuario, para el nuevo sistema. Al tratarse de una plataforma que presta servicio globalmente a cualquier usuario que lo desee, tampoco se puede llevar seguimiento del proceso de implementación por completo. Aunque sí es controlable y cuantificable la implementación técnica del sistema, principalmente las modificaciones incrementales que impactan en la arquitectura de la plataforma.

Sin embargo, sí se pone énfasis en el público objetivo a aquellos usuarios que la organización, detrás de la plataforma, desea atraer. Esto resulta en la elaboración de un método de conversión, y sobre todo de atracción y adaptación del nuevo sistema definido de acuerdo a las expectativas de la organización, o más bien, un plan alineado con las estrategias de negocio adoptadas.



Establecido lo anterior, el sistema se inserta en el mercado para prestar servicio en dos fases. La primera tiene como objetivo la generación de la comunidad en torno a la plataforma. Una vez establecida una comunidad lo suficientemente grande, entra en juego la segunda fase que se encarga de asentar las estrategias de negocio y el modelo económico que se quiere adoptar para generar beneficios en la organización, pero manteniendo también los beneficios que representa para todo tipo de usuario.

Cada fase, a su vez, utiliza distintas estrategias de conversión e implementación. La conversión resulta entonces en una combinación de métodos. También, se requiere tener en cuenta el uso interno del sistema, para los gestores de contenido. Puede que el personal encargado de filtrar el contenido y mantener la plataforma con obras de gran calidad esté acostumbrado a la utilización de sistemas similares.

Es de suma importancia hacer notar que los objetivos y expectativas económicas de la empresa se entrelazan con esta fase de implementación y conversión. La implementación e inserción al mercado son críticas para el éxito de la plataforma. Esta implementación no es tradicional en el sentido de que no conlleva únicamente adaptar a los usuarios al nuevo sistema para mejorar su eficiencia, sino que también se aprovecha la retroalimentación de estos para mejorar el sistema, solucionar fallos y errores inesperados, aumentar la calidad en términos de usabilidad (comodidad) y *performance*.

Los resultados de la implementación alimentan, de forma continua, el crecimiento del sistema. Se divide en etapas acordes al modelo económico. Y para la transición se consideran las plataformas alternativas ya existentes en el mercado, a las cuales no se tiene completo acceso.

1. Primera etapa: generación de comunidad, atracción de usuarios nuevos que ya hayan tenido experiencia con plataformas similares o no. Se enfoca en mejorar la usabilidad para convencer a los usuarios.
  - Piloto: Se les brinda a los usuarios un acceso anticipado a la plataforma, en términos de pruebas *beta*, antes de lanzar oficialmente el sistema al mercado. Tiene como objetivo dar a conocer la plataforma, evaluar el desenvolvimiento de los usuarios en la misma al recibir retroalimentación y preparar el sistema para lanzarlo oficialmente. Pone a prueba el nuevo sistema, tiene bajo costo pero deriva en una etapa crítica para el éxito de la plataforma.
2. Segunda etapa:
  - Directo: No se puede obligar a un usuario a abandonar otras plataformas, incluso hasta podría ser negativo para Omics. Sin embargo, a partir de esta etapa el foco ya no está en mantener funcionalidades similares a otros sistemas para hacer más amena la transición de nuevos usuarios

entrantes, sino en expandir el rango de funcionalidades presentes y planificadas. Así se marca una diferencia que destaque a la plataforma frente a otras.

Por esto, el método de implementación, con el sistema funcionando de forma estable continuamente, se basa en evaluar las expectativas de los usuarios existentes para satisfacerlas mediante nuevas o mejores funcionalidades.

Un usuario satisfecho con el sistema es probable que abandone las demás plataformas para hacer uso completo de *Omic*s.

La primera etapa tiene por objetivo incrementar gradualmente la cantidad de usuarios registrados en el sistema, por lo que deben atenderse sus propuestas de mejora e inquietudes.

Actividades de Marketing y de Community Management actúan de forma constante desde un inicio, acompañando toda la evolución de la plataforma. Inicialmente, lo hacen para atraer la mayor cantidad de usuarios que puedan interesarse en los servicios ofrecidos, y así, posteriormente, mantener y alimentar la comunidad generada. El esfuerzo de quien se desempeña como Community Manager es crucial en las etapas tempranas del sistema, ya que dedican todo su tiempo a la difusión de la organización y la plataforma (junto a sus beneficios) a través de redes sociales y, también, a través de foros donde existe una comunidad formada en torno al mundo del cómic y obras similares.

En una primera implementación se requiere mayor trabajo manual. Las actividades que se tienen en cuenta, de forma general, para efectuar la implementación son:

- ❖ Planificación de las tareas: Establecer las tareas con previsión acerca de las actividades que se deben realizar y estimar los tiempos de cada una.
- ❖ Asignar los responsables: Deben asignarse de forma precisa las tareas a realizar, para que cada responsable pueda identificar su relación con las etapas de puesta en funcionamiento del sistema.
- ❖ Verificación y control de infraestructura: Actividad centrada en evaluar la infraestructura general (servidores, base de datos, servicios externos, seguridad).
- ❖ Instalación de herramientas que soportan la arquitectura del sistema: Instalación y configuración de todas las dependencias del proyecto, siendo estas librerías, frameworks, drivers, instancias de servicios (como bases de datos y bus para comunicación asíncrona).
- ❖ Carga del código del sistema: El código completo del sistema se encuentra en repositorios, por lo que es necesario extraer de las ramas correspondientes los

archivos para poner el sistema en funcionamiento, para cargarlos en el servidor principal donde opera.

- ❖ Ejecución de pruebas: En el mismo servidor donde funciona se ejecutan las pruebas unitarias y de integración para corroborar un funcionamiento acorde a los requerimientos de usuarios sin fallas lógicas. Es imperativo que se pasen todas las pruebas para que la plataforma pueda ponerse en funcionamiento.
- ❖ Ejecución de migraciones: Se ejecutan los scripts iniciales para crear las bases de datos correspondientes y también las migraciones que realizan modificaciones sobre el esquema de datos. Las migraciones permiten, en un futuro, alterar este esquema de datos, ampliando la estructura de la base de datos, según lo plantean los requerimientos.
- ❖ Ejecución de scripts para poblar las bases de datos: Ya establecido el esquema de los datos en las bases de datos relacionales, se ejecutan scripts que completan las tablas con los datos iniciales: inserción de datos globales como categorías, estados, configuraciones y creación de usuarios con niveles de administrador.
- ❖ Configuración de puntos de acceso: Se configuran los puntos de acceso de cada servicio, sea interno o del mismo sistema. Técnicamente, se establecen las redes virtuales mediante las cuales se comunica el sistema con las instancias de servicio como las bases de datos y buses de comunicación, y también se gestionan los puertos en los que opera cada servicio.
- ❖ Capacitación: Se capacita el personal encargado de monitorear el funcionamiento del sistema, los servidores y los servicios de los que depende. A su vez, se capacita al personal interno que hace uso de funcionalidades de administrador para gestionar el contenido.
- ❖ Monitoreo del sistema en funcionamiento: De acuerdo a lo planificado, se monitorea el sistema operando para asegurar su disponibilidad las 24 horas del día.
- ❖ Difusión en redes sociales y foros: En la implementación inicial y futuros incrementos que representan grandes cambios, los cuales pueden aportar nuevos o mejores beneficios a los usuarios finales, el esfuerzo de Marketing es fundamental para dar a conocer estas características.

De lo anterior, se observa que no todas las tareas son ejecutadas en cada iteración, esto depende del contexto. Algunas de estas poseen todas las características para ser automatizadas, como se especifica más adelante.

Las tareas de planificación y asignación de responsabilidades se realizan en un primer inicio y son poco propensas a cambiar con el tiempo, en cada una de las implementaciones incrementales. El último punto también requiere esfuerzo manual de

las áreas encargadas de la difusión de la plataforma, aunque no periódicamente, sino cuando la organización y avances en el sistema lo requieran.

La infraestructura se verifica constantemente para corroborar su correcto funcionamiento, tanto servicios individuales como estos operando en un todo.

### **Aspectos técnicos**

Las actividades que involucran desarrollo (codificación) del sistema, testing, *merging* de cambios, subida al repositorio, descarga del repositorio en servidor principal y puesta en funcionamiento (reemplazando anterior instancia) se automatizan. Requieren que los desarrolladores involucrados posean un perfil de *DevOps*, es decir, que tengan tanto habilidades de desarrollo como de operación (para poner en funcionamiento el sistema y su arquitectura).

Todas las etapas deben estar cubiertas por técnicas de automatización que faciliten las actividades. Los desarrolladores deben centrar su trabajo en ampliar, o mejorar las funcionalidades que surgen de los requerimientos del sistema. Además, futuros cambios en el código del sistema no deben retrasar al equipo de desarrollo, ni al de *testing* haciendo que estos vuelvan a probar las funcionalidades que en el pasado ya se aprobaron.

En primer lugar, es necesario asegurar la calidad del código y su correcto funcionamiento. Por esto, el testing continuo del código, las funciones y la integración acompaña a todo el proceso de desarrollo. Este se divide en:

1. Testing unitario: Se prueba cada función específica del código para corroborar que su lógica sea correcta y solucione de forma adecuada la necesidad para la que fue creada. Se corroboran los resultados derivados de ejecutar una función (o método) con determinados parámetros.
2. Testing de integración: Se comprueba que distintos módulos, funcionando en conjunto, cumplan su cometido, y generen los resultados esperados. Para esto se ejecutan los módulos de acuerdo a los establecido por los distintos casos de uso, esperando que se satisfagan de forma adecuada.
3. Testing manual: Algunas modificaciones o cambios fundamentales en la plataforma requieren de testing manual, es decir, una persona encargada de esta actividad tiene que probar todas las partes del sistema que así lo requieran para asegurar un nivel de calidad aceptable.

Para cubrir la implementación, y según lo descrito en el apartado anterior, se utiliza una metodología de **integración continua**. Esto sucede ya que los tests cubren gran parte de la funcionalidad del sistema de forma automática y se pueden ejecutar en cualquier servidor sin intercesión de una persona. Así, una

vez que nuevo código se sube al repositorio central del sistema, un servicio externo clona este repositorio, ejecuta los tests de forma autónoma, y si estos "pasan" (se cumplen las aserciones y no se encuentran errores), este mismo servicio se encarga de hacer una nueva *build* del sistema y realiza la puesta en producción.

Lo anterior, referido a automatización, puede resumirse mediante el siguiente flujo que expresa un vistazo técnico:

- ❖ Se desarrolla el sistema, agregando, modificando o eliminando código.
- ❖ Se realiza un *merge* de este código con el existente en el repositorio local del desarrollador.
- ❖ El desarrollador sube los cambios al repositorio central del sistema.
- ❖ Un servicio de integración continua, externo, detecta que se han realizado nuevos cambios por lo que clona el repositorio localmente.
- ❖ Ejecuta los tests unitarios y de integración de forma automática.
- ❖ Si los tests no arrojan errores, se construye (*build*) el sistema desde el código haciendo uso de scripts de automatización.
- ❖ Se ejecutan las migraciones para que el esquema de base de datos se corresponda con el modelo de datos que utiliza el sistema.
- ❖ El sistema construido se pone en producción (o en un entorno de *testing*, según se requiera) al subirlo al servidor correspondiente y ejecutar las rutinas para detener la versión anterior del sistema que ya estaba corriendo y correr la nueva versión.

Las ramas de los repositorios (*development*, *testing* y *master*) determinan a qué entorno se hace *deploy* del sistema. Así, si se han realizados cambios grandes en el funcionamiento de la plataforma, se extiende el sistema en un **entorno de testing** (también llamado, *staging*) donde una persona debe probar lo necesario para asegurar el correcto funcionamiento, y una vez logrado esto, el sistema se extiende en producción.

Por último, para facilitar todo el proceso anterior, se utiliza un sistema de contenedores (con tecnología Docker, en este caso) para realizar las *builds* del sistema. Cada *build* es completamente independiente del resto de la arquitectura, cuyas dependencias son únicamente otros contenedores de Docker, los cuales corren instancias de servicios necesarios, como el sistema de gestión de base de datos. Además, gracias a Docker como plataforma, la configuración de puntos de acceso, precisamente las redes virtuales sobre las que operan todos los servicios que hacen a la plataforma, se realiza de forma sencilla y segura mediante archivos de configuración.

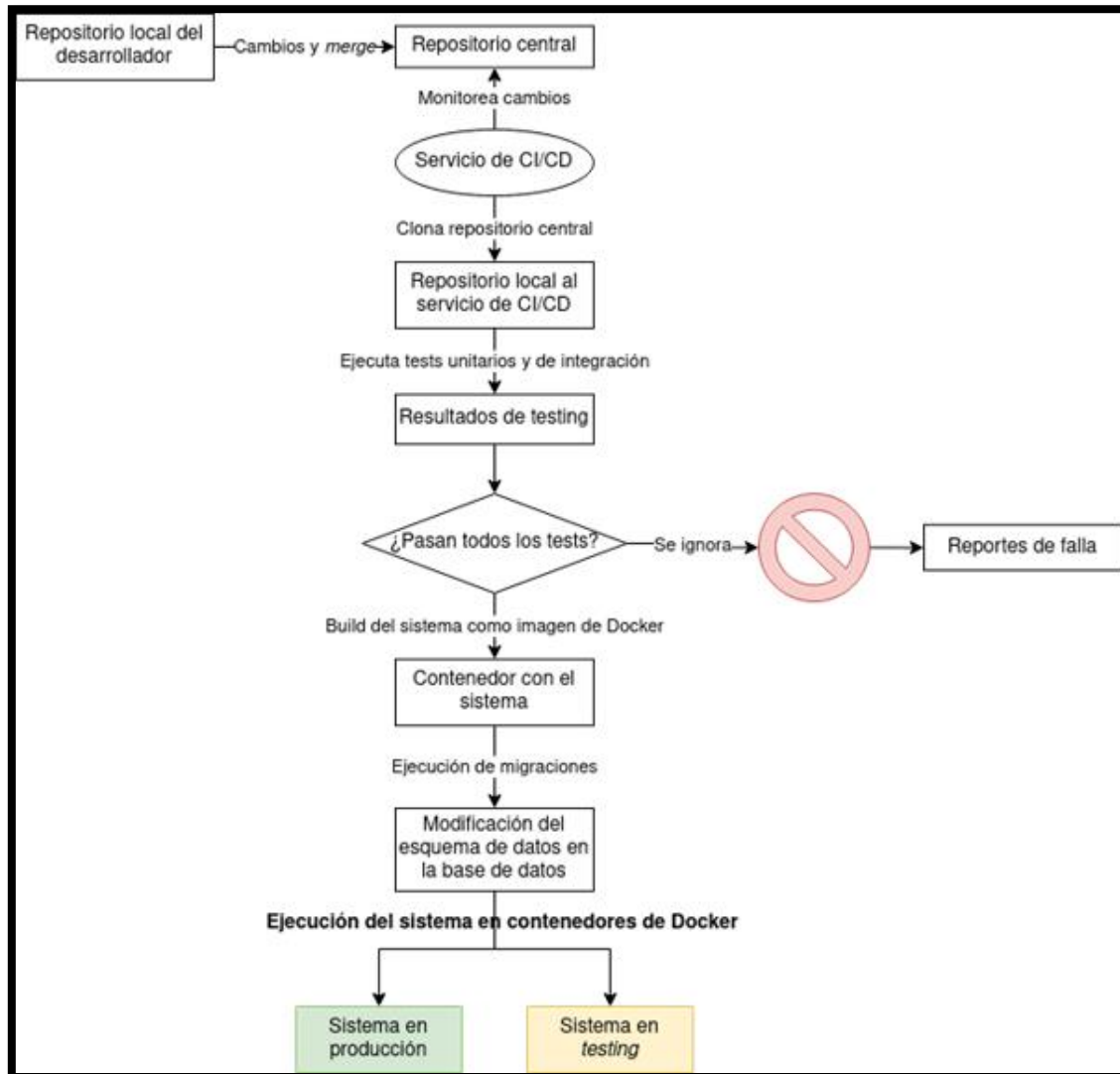


Diagrama 3.1 Proceso de Integración

Todo el proceso puede observarse en el diagrama anterior. Se considera que las migraciones de base de datos se incluyen junto a los cambios que se realizan en el repositorio, por mano del mismo desarrollador.

La tecnología a utilizar, en base a la implementación del sistema es la siguiente:

- Distribución estable, pensada para alojar servidores, de GNU/Linux para correr Docker y los contenedores necesarios.
- Docker: Contenedores basados en la virtualización provista por KVM en Linux.
- CircleCI: Servicio de integración continua, encargado de monitorear el repositorio principal para descargar el código y realizar nuevas *builds*.
- GitHub: Hosting de los repositorios. Utiliza *Git* como herramienta.

- Diversos frameworks para realizar testing unitario, de integración y de seguridad en el sistema.
- Knex: Para crear y ejecutar las migraciones de esquemas de datos en la base de datos.

### 3. Funciones del jefe de proyecto por importancia

Las funciones del Director de Proyecto son las siguientes:

Nº	Función	Breve explicación
1	Toma de decisiones.	<p>No sólo debe tomar decisiones a futuro, con suficiente margen para evaluar sus resultados, sino que, principalmente, debe tomar decisiones de forma proactiva para ejecutar acciones ante el dinamismo del mercado que conlleven al éxito de la plataforma. Además, gestionar posibles riesgos de forma preventiva y, en caso de ser necesario, de forma correctiva.</p> <p>La toma de decisiones de forma eficiente, adelantándose a posibles problemas que puedan aparecer, es importante ya que el rumbo del proyecto depende de esta.</p>
2	Ejercer liderazgo.	<p>Gestionar los recursos. Guiar al personal para cumplir con los objetivos del proyecto. Alinear los objetivos del proyecto con los objetivos de la organización.</p> <p>Comunicar la visión de la organización y la suya de acuerdo al proyecto.</p>
3	Formular el proyecto.	<p>Establecer formalmente las directrices del proyecto, qué se espera y cómo lograrlo. Definir los objetivos e hitos, y la finalidad del proyecto.</p> <p>Una adecuada formulación, y el mantenimiento de la misma mediante actualizaciones, permiten tener un punto de comparación entre esta y la realidad del proyecto.</p>

4	Realizar la planificación y gestionarla.	Planificar los cursos de acción que deben ejecutarse para asegurar el cumplimiento de los objetivos y el desarrollo del mismo. Dividir el proyecto en etapas, tareas, actividades y fases necesarias para que sea factible su desarrollo a corto, mediano y largo plazo.
5	Aplicar retroalimentación y resolución de conflictos.	<p>Gestionar el flujo de retroalimentación en todas las etapas y frentes: del personal de acuerdo a la participación en el equipo y en el proyecto; y del usuario conforme a su desenvolvimiento en el uso de la plataforma y de la percepción que tenga de la usabilidad. La retroalimentación será aquella que le permita modificar el rumbo del desarrollo del sistema para adaptarlo a lo que el usuario o el personal necesite.</p> <p>Resolver conflictos generados entre los participantes de la organización en cada equipo.</p>
6	Asignar tareas y recursos.	Además de planificar el curso de acción y subdividir el proyecto en distintos niveles de actividad, debe asignar los recursos necesarios a cada tarea para asegurar su completitud. Para ejecutarse una tarea, requiere de recursos.
7	Verificar y controlar entregables.	Corroborar que se esté cumpliendo en tiempo y forma con los hitos y entregables. Esto es, cumplir con los tiempos límites para la entrega del material correspondiente a determinadas fechas. Tomar las medidas necesarias para ajustar el desarrollo del proyecto a los tiempos previstos.
8	Analizar, diseñar y aplicar técnicas de motivación individual.	<p>Motivar al personal, teniendo en cuenta sus necesidades, para aumentar la eficiencia en el trabajo de los empleados.</p> <p>De la motivación de los individuos depende el desarrollo del proyecto y las mejoras que se puedan aplicar al mismo.</p>



9	Supervisión y control de cumplimiento.	Corroborar que los objetivos del proyecto se estén cumpliendo y tomar las medidas necesarias para encaminar el proyecto a esto.
10	Aplicar técnicas y métricas de estimación de tiempo y esfuerzo del proyecto.	Basar las estimaciones del tiempo necesario para la ejecución de las tareas, y sus dependencias, en métricas funcionales y formales para verificar el cumplimiento de las mismas. Aplicar técnicas para mantener estas estimaciones realistas y acordes a los requerimientos del proyecto.
11	Administración eficiente de recursos y gestión de presupuestos.	Gestionar coherentemente los recursos limitados de la organización para desarrollar el proyecto de forma eficiente. Asignar los recursos necesarios y suficientes a las partes que lo requieran.
12	Generar informes iniciales, parciales y finales del proyecto.	Formalizar la información que surge del desarrollo continuo del proyecto para realizar mejores estimaciones y mantener métricas significativas para los involucrados. Así, conocer en todo momento el estado del proyecto.
13	Aplicación de diferentes estilos y técnicas de comunicación interpersonal.	El proyecto se desarrolla gracias al esfuerzo realizado por un equipo de personas, por lo que la comunicación entre las partes, incluyendo al líder del proyecto, debe ser rápida, sin ambigüedades, y clara.
14	Ejercer el coaching.	Centrado en el personal y sus necesidades, para alentar sus capacidades y el trabajo en equipo como un todo. Incrementar la sinergia.
15	Realizar tareas técnicas de análisis, diseño, desarrollo, testing e implementación.	El resultado del proyecto es un producto cuyo fin es prestar un servicio, como todo sistema de información. El desarrollo del sistema, según lo planificado a lo largo del proyecto, y su implementación es el fin último de este tipo de proyectos.

Tabla 3. 4 Funciones del Director del Proyecto

#### 4. Principales funciones del Jefe de Proyecto en fase de ejecución

Las 7 principales funciones que cumple el Director de Proyecto durante la fase de ejecución.

N°	Función	Breve explicación
1	Toma de decisiones.	<p>No sólo debe tomar decisiones a futuro, con suficiente margen para evaluar sus resultados, sino que, principalmente, debe tomar decisiones de forma proactiva para ejecutar acciones ante el dinamismo del mercado que conlleven al éxito de la plataforma. Además, gestionar posibles riesgos de forma preventiva y, en caso de ser necesario, de forma correctiva.</p> <p>La toma de decisiones de forma eficiente, adelantándose a posibles problemas que puedan aparecer, es importante ya que el rumbo del proyecto depende de esta.</p>
2	Ejercer liderazgo.	<p>Gestionar los recursos. Guiar al personal para cumplir con los objetivos del proyecto. Alinear los objetivos del proyecto con los objetivos de la organización.</p> <p>Comunicar la visión de la organización y la suya de acuerdo al proyecto.</p>
3	Aplicar retroalimentación y resolución de conflictos.	<p>Gestionar el flujo de retroalimentación en todas las etapas y frentes: del personal de acuerdo a la participación en el equipo y en el proyecto; y del usuario conforme a su desenvolvimiento en el uso de la plataforma y de la percepción que tenga de la usabilidad. La retroalimentación será aquella que le permita modificar el rumbo del desarrollo del sistema para adaptarlo a lo que el usuario o el personal necesite.</p> <p>Resolver conflictos generados entre los participantes de la organización en cada equipo.</p>
4	Verificar y controlar entregables.	<p>Corroborar que se esté cumpliendo en tiempo y forma con los hitos y entregables. Esto es, cumplir con los tiempos límites para la entrega del material correspondiente a determinadas fechas. Tomar las</p>

		medidas necesarias para ajustar el desarrollo del proyecto a los tiempos previstos.
5	Analizar, diseñar y aplicar técnicas de motivación individual.	Motivar al personal, teniendo en cuenta sus necesidades, para aumentar la eficiencia en el trabajo de los empleados.  De la motivación de los individuos depende el desarrollo del proyecto y las mejoras que se puedan aplicar al mismo.
6	Supervisión y control de cumplimiento.	Corroborar que los objetivos del proyecto se estén cumpliendo y tomar las medidas necesarias para encaminar el proyecto a esto.
7	Aplicación de diferentes estilos y técnicas de comunicación interpersonal.	El proyecto se desarrolla gracias al esfuerzo realizado por un equipo de personas, por lo que la comunicación entre las partes, incluyendo al líder del proyecto, debe ser rápida, sin ambigüedades, y clara.

Tabla 3. 5 Funciones principales del Director del Proyecto en fase de ejecución.

## 5. Incorporación de dos personas

La incorporación de 2 personas al equipo del Proyecto se efectúa al inicio de la etapa de Diseño para realizar tareas que sean de publicidad, logotipos de marca y creación de la imagen del proyecto y funciones del sistema. Es importante realizarlo en esta etapa antes de que se empiece con el desarrollo e implementación, ya que en producción se empieza con un proceso crítico donde se debe tener en claro de antemano cuáles son los objetivos y pasos para llevar a cabo el proyecto.

Una persona en el puesto de Community Manager, y la otra en el puesto de Diseñador UX/UI.

Es importante tener en claro los resultados que se desean obtener para contratar a los trabajadores capaces para la tarea que tienen que realizar, considerando las fortalezas y debilidades del equipo ya establecido para que la presencia simultánea entre ellos y los que se van a anexar sea armoniosa. El hecho de contratar a estas 2 personas de forma adecuada es fundamental para lograr que el proyecto se alcance con la calidad deseada. La idea es que los nuevos integrantes del equipo puedan dar lo mejor de sí mismos y que el aporte que ejecuten no sobrecargue a los trabajadores que están desde el comienzo.

Previamente a la decisión de incorporar a estas personas al equipo, se deben considerar las relaciones interpersonales que ya existen en el mismo y las posibles barreras de comunicación que se puedan presentar. Se debe potenciar el trabajo colaborativo, tanto presencial como remoto y facilitar la comunicación e intercambio de información entre los integrantes del equipo. También seleccionar las personas con el mejor perfil laboral teniendo en cuenta las necesidades para el proyecto.

Se contemplan los siguientes roles como prioritarios:

## **1. Community Manager:**

### **1. Perfil:**

- Persona mayor de 18 años de edad y menor a 30 con excelente habilidad de redacción.
- Preferentemente de Latinoamérica y habla hispana nativa.
- Experiencia manejando grupos y páginas en redes sociales (Instagram, Twitter y Facebook preferentemente).
- Buen humor y habilidad para satirizar contenido.
- Conocimientos y seguimientos de modas en redes sociales (memes, influencers, etcétera).
- Pasión y conocimiento de la industria de los cómics (no excluyente).
- Título de secundaria completo y experiencia en eventos de comics, anime, manga y videojuegos.
- Haber participado en foros, blogs y artículos relacionados a los cómics.
- Manejo de inglés.
- Habilidades para trabajar correctamente de forma remota.

### **2. Actividades:**

- Generar comunidad en las redes sociales de *Omic*s a través del humor con contenido relacionado a la plataforma.
- Informar claramente las necesidades que tiene el sector de marketing comunicar a los usuarios.
- Revisar y filtrar mensajes de las redes sociales para acercar información que puede afectar a la plataforma como lo perciben sus usuarios.
- Contestar inquietudes comunes sobre el uso de la plataforma *Omic*s.

- Brindar información clara del impacto de características y servicios ofrecidos en la plataforma (métrica de Facebook de posts realizados, alcances de historias en Instagram, etcétera).
- Ayudar al equipo de desarrollo estar pendiente de las necesidades principales de los usuarios (lectores o autores) más influyentes e importantes de la plataforma.

## **2. Diseñador UX/UI:**

### **1. Perfil:**

- Persona mayor de 25 años con experiencia comprobable (mediante portafolio de trabajos realizados y participación en eventos y/o concursos).
- Excelente observación sobre el uso de plataformas de contenido de entretenimiento visual.
- Conocimiento sobre el diseño y desarrollo base sobre realizar un cómic.
- Manejo intermedio de inglés.
- Habilidad en software como Balsamiq, Sketch o Axure (no excluyente).
- Tener buena comunicación para trabajar en equipo y captar necesidades para mejorar el uso de la plataforma.
- Preferentemente de nacionalidad argentina.
- Ser curioso, analítico y adaptable al cambio.
- Nociones de diseño gráfico y comunicación visual para poder realizar mockups y prototipos.
- Nociones básicas de programación web (excluyente).
- Habilidades para trabajar correctamente de forma remota de ser necesario.
- Conocimiento de metodologías ágiles.

### **2. Actividades:**

- Simplificar el uso de la plataforma *Omics* para los usuarios.
- Realizar un diseño claro y reconocible.
- Escuchar claramente las necesidades de los clientes para conocer y entender cómo utilizan la plataforma y así brindarles una mejor experiencia.
- Ejecutar pruebas del uso de la plataforma.
- Definición de modelos de interacción, flujos de tareas de usuario y especificación de UI.

- Trabajar en equipo los desarrolladores de *Omics* para incorporar la identidad de marca.

Es fundamental tener paciencia con la incorporación del personal y hacer sentir bien a los nuevos colaboradores que forman parte del equipo desde el primer momento. Son cruciales los primeros días en la organización para su adaptación e integración ya que esto se va a ver reflejado en su productividad y en la permanencia en el proyecto.

Una vez que estas dos personas formen parte del proyecto, es importante dar a conocer a los nuevos trabajadores con el resto del equipo del área de la que van a ser parte, y por supuesto profundizar sobre las actividades que van a realizar. También adecuar su nivel formativo al de los requisitos del proyecto, tanto en el momento de la incorporación como a lo largo de su estancia en *Omics*. Para ir evaluando su desempeño, es trascendente valorar las tareas en relación a su puesto y al proyecto, controlar la consecución de los objetivos que le han sido asignados y comunicados previamente, entre otros. Es positivo que todos los miembros se sientan involucrados y comprometidos con la planificación y la toma de decisiones del mismo. Contar con un equipo que se integre bien y que puedan hacerse consultas entre ellos en caso de que se presente alguna duda, permite un alto rendimiento que incrementa las posibilidades de éxito del proyecto. Sus capacidades, conocimientos, habilidades de innovación, motivación, entre otros, se conforman como el activo intangible estratégico más importante.

## 6. Liderazgo

Durante la ejecución del proyecto se debe utilizar el estilo de liderazgo **Democrático** porque es un tipo de liderazgo que fomenta la participación de todos los miembros consultando sus decisiones y acciones a tomar, lo cual es un tipo de relación muy saludable para el equipo porque todos los miembros del grupo aportan sus puntos de vista e intervienen en las decisiones a tomar (principalmente al ser un grupo reducido de personas).

Hay una colaboración activa por parte de todos los integrantes y el líder es un modelo de referencia, pero todos se sienten libres de participar en igualdad de condiciones. La premisa más importante es que hay un diálogo constante entre todos que favorece la consecución de los objetivos fijados.

El líder debe fomentar la comunicación y la participación conjunta en las decisiones, animar y agradecer las sugerencias de los integrantes. Cuando hay que tomar una decisión, se decide que el miembro del equipo con más experiencia en el área sea quien tenga la palabra final pero contando con la opinión del resto de los integrantes.

El perfil de un líder democrático debe tener paciencia y empatía. Además, los tipos de trabajadores deben ser variados y más especializados que sus superiores, así, su conocimiento único hace que su opinión deba ser tenida en cuenta.

Este tipo de liderazgo tiene ciertos riesgos tales como en los momentos que se requiere la toma de decisiones inmediata, ya que posponerlas a la espera de la consulta con el equipo puede hacer perder una oportunidad de negocio. Incluso puede ser peor y poner en riesgo la viabilidad de la empresa.

## 7. Resolución de conflictos

Siendo el Director de Proyecto un impulsor dentro del equipo, el cual dirige las acciones orientadas para lograr alcanzar los objetivos del proyecto, se necesita que sea una persona versátil. El Director de Proyecto debe aceptar que los conflictos son inevitables. Es decir, que frente a la situación de conflicto, debe optar por abordarlo aplicando el enfoque correcto, para así obtener resultados positivos.

Puede ser factible que resuelva el conflicto mediante una negociación, o bien apelando a normas y políticas institucionales para encontrar una solución.

### **Situaciones hipotéticas que se pueden presentar, se detallan a continuación:**

- ❖ Quedan dos semanas para comenzar con la planificación de las pruebas (pruebas de integración, pruebas unitarias, pruebas de carga, pruebas de seguridad) y un miembro del equipo piensa que hay que rediseñar toda la Etapa de Diseño para reducir el tiempo de planificación a futuro, y el resto de los miembros opinan que eso implicaría un riesgo de retraso inmediato considerando el tiempo que resta para entregar el próximo hito, optando por conservar entonces el diseño actual, y evaluar si es factible el cambio.

### Enfoque utilizado para resolver el conflicto:

- Enfoque de Colaboración.  
Bajo este enfoque el Director de Proyecto, puede asignar una tarea especial de rediseño al miembro del equipo que propuso dicho cambio en el Diseño del proyecto, y evaluar entre todos los miembros si realmente es factible realizar el cambio propuesto. Mientras tanto, el resto del equipo sigue con la planificación establecida con anterioridad, y el Director toma para sí las tareas correspondientes al miembro que ahora hará el rediseño del proyecto
- ❖ Un miembro del equipo no cumple con los tiempos de finalización de su User Story, retrasando todo el Sprint.

Enfoque utilizado para resolver el conflicto:

➤ Enfoque acomodaticio.

Bajo este enfoque el Director de Proyecto habla con el Scrum Master para saber cuáles fueron exactamente las causas de retrasos del miembro del equipo, y tras saber los motivos del retraso, realizar una reasignación de tareas al miembro en cuestión para que no se vean comprometidos los tiempos de entrega del equipo.

- ❖ La entrega del proyecto lleva 3 semanas de retraso y se considera que no se llega a cumplir con la entrega final debido a que se ha tomado un enfoque demasiado amplio del sistema.

Enfoque utilizado para resolver el conflicto:

➤ Enfoque Agresivo.

Bajo este enfoque el Director de proyecto al ver que quedan pocos días para realizar la presentación del proyecto a producción, decide imponer tiempo extra de trabajo a todos los miembros del equipo para sí poder cumplir con los tiempos de entrega establecidos en la planificación.

- ❖ Se descubre requisitos de gran importancia no contemplados en la etapa de diseño y, por lo tanto, de desarrollo.

Enfoque utilizado para resolver el conflicto:

➤ Enfoque Agresivo.

El líder de proyecto asigna las tareas que surgen de los requisitos correspondientes, recién descubiertos, en base a las capacidades y experiencia de cada integrante del equipo.

- ❖ Algún integrante, principalmente aquellos encargados del testing, descubre fallas leves (interfaz, sintaxis, funcionalidades de apoyo sencillas) cerca de una fecha de entrega o implementación. Puede generar conflicto entre los desarrolladores y testers.

Enfoque utilizado para resolver el conflicto:

➤ Enfoque de evasión.

Se evalúa el impacto de estas fallas para posponer su corrección y análisis a fondo antes de la siguiente implementación o despliegue.



## 8. Técnicas de motivación

Las técnicas de motivación elegidas para que el equipo pueda lograr ser óptimo, eficiente y productivo son las siguientes:

### 1. **Delegar autoridad a los miembros del equipo (Técnica de motivación positiva).**

Delegar autoridad, significa que se puede ceder, trasladar o traspasar a un subordinado, una parte de la autoridad que tiene el dirigente, para que se ocupe de algunas funciones o tareas de manera expresa.

La delegación de autoridad influye `positivamente en la motivación de los subordinados, a la vez que contribuye a desarrollar su sentido de responsabilidad.

Cuando se utiliza esto de manera correcta, se obtienen resultados muy favorables. Por otra parte, se incrementa la eficiencia del trabajo del dirigente, ya que su capacidad, tanto física como intelectual, resulta multiplicada y le permite entonces concentrar la atención en las tareas de mayor importancia.

En primer lugar, el dirigente debe definir con claridad, cuáles son las tareas que van a ser delegadas, el tiempo necesario para su ejecución, los objetivos que se quieren alcanzar, el marco de acción en el que podrá moverse y la persona que va a asumir cada una de las tareas. Si las tareas no son muy precisas, y no está definido lo que se puede y lo que no se puede hacer, es muy probable que se presenten errores. Es por eso que, en gran medida, prevenir y evitar esto está en manos del dirigente.

También es importante destacar, que la delegación de autoridad constituye una vía para la estimulación. El hombre, como ser que trabaja y piensa, tiene determinados intereses y aspiraciones. Para dirigir a los hombres con acierto, es necesario estimularlos de manera eficaz. El dirigente deber reconocer los buenos resultados alcanzados por los subordinados en el desempeño de sus tareas y criticar de forma constructiva los errores cometidos teniendo presente las causas de los errores y enseñarles las formas más correctas para la adecuación posterior. Sin embargo, si se le da instrucciones muy precisas y detalladas acerca de cómo se quiere que sean hechas todas y cada una de las tareas asignadas, puede eliminar la satisfacción del trabajador, pues hay que pensar que éste podría resolver o determinar el método de solución con su propia iniciativa.

En lugar de un clima de intolerancia a los errores, es preferible crear un ambiente de confianza y seguridad, en el cual los subordinados pueden probar sus capacidades e iniciativas, lo cual repercutirá favorablemente en el desarrollo del trabajo y en la propia formación de ellos.

Situaciones donde es necesario aplicar la técnica:

- ❖ Cuando las tareas desbordan a miembros del equipo, es recomendable delegar para que la realización de las mismas sea compartida o dividida en el caso que se pueda y para que se lleguen a las metas establecidas.
- ❖ Si alguien no está capacitado sobre la tarea que se asignó, es mejor delegar autoridad a otra persona.
- ❖ Si alguien no está capacitado sobre la tarea que se asignó, se puede delegar a una persona extra que pueda ayudarla para cumplir lo establecido.
- ❖ Por causas mayores, la persona que tiene autoridad asignada sobre una actividad puede no estar disponible (o ausente) y se delega a otro miembro del equipo.

## **2. Reconocer el trabajo bien hecho y estimular la mejora continua (Técnica de motivación positiva).**

El reconocer el trabajo de otras personas es una de las técnicas que mayormente funcionan en los equipos. Se basa en el respeto y celebrar los logros por los aportes individuales de cada miembro. Destacar que el aporte realizado es valorado y tenido en cuenta, hará que las personas se sientan más predispuestas a seguir generando propuestas, superar cada etapa impuesta con el entusiasmo necesario y estimular el trabajo en equipo.

Esta técnica sirve para dar incentivo a aquellos miembros que se sienten desmotivados, que cometen errores o equivocaciones porque se sienten desvalorizados frente a su trabajo o clima laboral y sienten miedo frente a situaciones que podrían ser humillantes. Es necesario reconocer este tipo de situaciones para guiarlos por el camino que se debe seguir y que el rumbo no se desvíe de la meta, para así lograr que mejoren progresivamente.

Una cosa es equivocarse y otra muy distinta es mantener una actitud tóxica o negativa. En estos casos, los miembros que no aportan nada al equipo y presentan una actitud perezosa o pesimista, necesitan lograr un cambio en ellos. Si no se logra y no se modifica la actitud negativa de las personas, es claro que hay que optar por algo que mitigue esa toxicidad para que el resto del equipo no se vea afectado y que no se genere un ambiente de malestar, ya que es muy contagioso.

Ciertamente no es un proceso ligero, pero paso a paso puede generar relaciones mucho más sólidas y constructivas, con proacción a los objetivos establecidos.

El reconocimiento, a su vez, puede ser: monetario, afectivo o por ejemplo un objeto material como un trofeo o un cuadro de reconocimiento. Es suficiente con demostrar el valor que se le da a la persona para que sus compañeros se sigan esforzando y crezca en cada uno el sentido de perseverancia.

En muchos casos, se hace alusión a la competencia para lograr resultados esperados en esta técnica, pero toda competencia no regulada es mala y puede retrotraer el esfuerzo aplicado para llevar la metodología a buen puerto.

Situaciones donde es necesario aplicar la técnica:

- ❖ Cuando la productividad de una persona empieza a crecer es recomendable reconocer ese trabajo frente a todos, para que los demás sigan su ejemplo.
- ❖ Tener cursos pagos de temas diversos (Platzi, Ed Team, Udemy, Código Facilito, etc.) de plataformas online para que el personal tenga incentivos de creatividad y nuevas tecnologías.
- ❖ El equipo logra alcanzar los objetivos antes de la fecha fijada por el líder, entonces se los podrá premiar con algo pactado con anterioridad para que el incentivo sea mayor y la competencia entre equipos sea productiva y no destructiva.

### **3. Estimular la comunicación (Técnica de motivación positiva).**

Esta técnica es una de las más importantes, ya que se debe emplear en cada ambiente laboral, organización, empresa o hasta en la vida cotidiana, al relacionarnos con personas. Sirve para motivar a equipos estableciendo una relación sana entre los pares, mejorar el rendimiento en conjunto, lograr facilitar los procesos que involucra el trabajo diario, prevenir posibles problemas frente a situaciones de mucho estrés, entre otros.

Además de lo dicho anteriormente, permite que en el equipo pueda aflorar el compañerismo, establecer parámetros de mejoras continuas tanto individuales como colectivas, disminuir la rivalidad y la competencia desleal, e incluso generar actividades post-office.

Situaciones donde es necesario aplicar la técnica:

- ❖ La poca comunicación entre áreas podría generar una baja en el avance de un proyecto, por ejemplo: en el área de desarrollo, se sigue avanzando según lo estipulado, pero el área de pruebas no ha logrado completar las pruebas del primer sprint. Esto ocasiona que cuando se llegue a producción, la aplicación

genere demasiados errores. Estimular la comunicación haría que el líder tome la decisión de acelerar el área de pruebas asignando recursos para que se logre el objetivo previsto.

- ❖ Si hay un conflicto personal, más allá del ámbito laboral, entre dos o más personas del equipo. Tener una forma de diálogo directo y un líder que pueda disminuir la tensión entre las personas, realiza una mayor productividad.
- ❖ En los momentos en que no hay un plazo muy urgente o próximo, realizar reuniones post-office con salidas previas del horario laboral. Esto ayudaría a disminuir tensiones y conocerse más en persona entre los que integran el equipo.

#### **4. Establecer las metas (Técnica de motivación positiva).**

Tener metas claras permite que el equipo avance hacia un objetivo claro y preciso estableciendo lo que se necesita para lograrlo. Al no tener un rumbo, es como dejar un barco sin timón, navegando por aguas turbulentas.

En el liderazgo de equipos, la información y la comunicación que se brindan son muy importantes. Al estar ausentes puede generar un desconcierto en el equipo. El jefe del grupo debe ser claro a la hora de explicar el trabajo que hay que realizar o las metas que hay que conseguir.

Si las metas no están claras nadie puede saber lo que tiene que hacer. Los objetivos siempre deben marcarse a corto, medio y largo plazo.

Es imprescindible destacar el dar instrucciones precisas, mostrar seguridad, pedir opiniones y colaboración, pero no cambiar de postura rápidamente. También recordar que al ser líder uno es referente frente a su equipo y hará que cuando se tome un camino, los demás puedan seguirte.

Situaciones donde es necesario aplicar la técnica:

- ❖ Es importante el establecer metas en momentos en que una parte del proyecto, en su desarrollo, está por terminarse (sea agregar nuevas funcionalidades, servicios o productos). De esa manera se mantiene una fluidez sobre lo que debe hacerse, sin tiempos muertos en la organización.
- ❖ Cuando por factores externos (un nuevo competidor o situación de emergencia sanitaria global) a planificación requiere un replanteo y nuevas metas para solucionar problemas que puedan surgir, esto debe llevarse de tal manera que no se pierda la continuidad en el proyecto.

- ❖ Una fusión o adquisición de otra empresa requiere tener metas claras y cambios en prioridades. De no ser así, puede perderse el tiempo y los recursos, además de perder la motivación.

#### **5. Utilizar incentivos o premios (Técnica de motivación negativa).**

El esfuerzo y el trabajo bien hecho han de ser reconocidos y premiados, pero a veces puede resultar contraproducente. Antes de ofrecer un incentivo o premio, es necesario evaluar el ambiente laboral, para no generar una competencia desleal en los miembros del equipo, es decir, si el clima de trabajo es constantemente negativo, utilizar esta técnica, hará que se torne en descontrol. Entonces se debe optar por otro método para calmar la situación. En cambio en un clima que es positivo, la gente es proactiva y presenta una buena comunicación, entonces beneficia la aplicación de este método. Hay empresas que bonifican la finalización de los objetivos impuestos con dinero y otras que lo hacen con días de vacaciones o de descanso.

Situaciones donde es necesario aplicar la técnica:

- ❖ Cuando se realiza una mejora sustancial en los ingresos de la empresa, se da un bono de 10% extra esperado a los involucrados en el proyecto realizado que produjo dicho ingreso.
- ❖ Al terminar antes de tiempo un proyecto o producto, se da la posibilidad de terminar antes su jornada laboral.
- ❖ Al acercarse un evento importante relacionado con las acciones de la organización, llevar representantes del equipo de desarrollo a los mismos.

#### **6. Estimular la superación personal y profesional (Técnica de motivación positiva).**

Si los miembros del equipo están bien preparados, el grupo puede conseguir un valor añadido. El líder debe preocuparse para que las personas se formen y especialicen, además de animarlos a que tomen la iniciativa de su propia superación personal y profesional. Las técnicas de motivación velan porque las personas asuman tareas desafiantes y retos propios.

Para aumentar el compromiso y la implicación, es importante que los líderes se interesen por las personas que forman parte del equipo, que les ayuden a conseguir los objetivos y a vivir con pasión lo que hacen.

Situaciones donde es necesario aplicar la técnica:

- ❖ Cuando se tiene previsto un cambio de tecnología a mediano plazo, es importante ir desarrollando al personal con cursos brindados por la organización en la misma. De esa manera se evita imprevistos.
- ❖ Si algún miembro de la organización tiene una situación personal muy fuerte que afecta su ritmo de trabajo, el líder que vela por su motivación debe asistir y recomendar profesionales para que se sienta más acompañado.
- ❖ Cuando hay falta de innovación, brindar acceso y material sobre cómics puede despertar pensamiento lateral para motivar y salir del estancamiento creativo.

## 9. Aspectos a tener en cuenta para negociar

### ESTRATEGIA DE NEGOCIACIÓN

Es importante tener en cuenta una serie de aspectos para realizar una negociación con un inversionista, ya que es la oportunidad que el equipo tiene de lograr convencerlo para que invierta en el proyecto.

Para la negociación es importante saber si el inversionista cita al encargado/s del equipo de *Omics* a un lugar específico, ya que si no lo hace se precisa buscar con antelación un lugar adecuado para poder hacer la reunión, el cual debe ser privado, sin gente ajena al proyecto. También hay que verificar que se puedan conectar todos los dispositivos electrónicos necesarios, buena iluminación y temperatura. Sería ideal que cuente con una sala para hacer un break, tomar un café y tener una charla más personalizada y distendida con el inversionista.

La reunión con el inversionista no debe ser muy extensa, ya que con 30 minutos es tiempo suficiente para explicar en qué consiste el proyecto, los objetivos, cuál es el público al que va dirigido, usar tablas para mostrar cómo se propone que se invierta el dinero en el proyecto, y cuáles son las ganancias obtenidas en un plazo de dos años.

Hay que contemplar que se presenta el plan de gastos en los que incurre el equipo para poder desarrollar el proyecto, detallando el destino de la financiación.

Otra consideración es buscar información acerca de quién es el potencial inversionista, si tiene redes sociales ver lo que comparte, conocer sus gustos, para lograr hablar su mismo idioma. Teniendo en cuenta todos los aspectos mencionados anteriormente, da una mayor probabilidad de éxito en las negociaciones al equipo de *Omics*.

### Se tiene en cuenta los siguientes aspectos para la negociación:

#### **1. Establecer una distinción entre la gente y el problema.**

Un aspecto fundamental de la negociación a tener en cuenta y que muchas veces se pasa por alto, es que estamos tratando con seres humanos. Los cuales tienen emociones, valores arraigados, experiencias vividas, y criterios en ocasiones diferentes a los nuestros, y además son impredecibles.

Una negociación encarada sin tener en claro que estamos tratando con seres humanos, que tienen reacciones humanas, puede tener consecuencias desastrosas.

El negociador tiene dos clases de intereses:

- Desea llegar a un acuerdo que satisfaga sus intereses de fondo.
- Desea mantener una buena relación de trabajo con la otra parte.

Cuando la negociación se asienta en las posiciones y no en los principios, tanto las cuestiones de fondo como la relación quedan comprometidas. Hay que separar la relación de los aspectos sustanciales.

Se debe pensar en términos de tres categorías básicas:

- Percepción: No la realidad, sino lo que está en la cabeza de la gente.
- Emoción: Cuando los sentimientos pueden ser más importantes que las palabras.
- Comunicación: Es el alma de la negociación.

## **2. Concentrarse en los intereses, no en las posiciones.**

El problema fundamental que se presenta en el proceso de negociación no es que haya posiciones enfrentadas, sino la existencia de un conflicto entre las necesidades, los deseos, las preocupaciones y los temores de cada parte; es decir los intereses.

La conciliación entre los intereses más que entre las posiciones da buenos resultados por dos razones:

- Respecto de todo interés hay varias posiciones posibles que pueden satisfacerlo.
- Detrás de las posiciones contrapuestas, se esconden muchos más intereses que los que salen a la luz en forma obvia.

## **3. Generar opciones para el beneficio mutuo.**

Ampliar la gama de opciones y no centrarse en una única respuesta, inventar formas para que la toma de decisiones sea fácil de tomar siempre procurando el beneficio mutuo

Existen obstáculos que pueden inhibir la producción de una multiplicidad de opciones:

- La formulación apresurada de un juicio.

- La búsqueda de una sola respuesta.
- El prejuicio de que el terreno es inmodificable.
- El pensar que la solución de los problemas de ellos es un problema de ellos.

Hay maneras de generar opciones creativas:

- Separar el acto de creación de opciones del acto de juzgarlas.
- Ampliar la gama de opciones en lugar de ir en busca de una única respuesta.
- Procurar el beneficio mutuo.
- Inventar formas de que las decisiones de ellos sean fáciles de tomar.

#### **4. Insistir en el uso de criterios objetivos.**

La negociación basada en principios desemboca en acuerdos inteligentes, aplicados de forma amigable y eficiente.

Cuanto más criterios de equidad, eficiencia o mérito científico, le acerquemos a la otra parte como sustento en la discusión de un determinado problema, más probabilidades tendremos de obtener una solución inteligente y justa.

Cualquiera sea el método de negociación que empleemos, siempre se obtienen mejores resultados si nos preparamos con anticipación. Elaborando criterios alternativos, y reflexionando sobre su aplicabilidad a nuestro caso.

- Criterios Justos: Los criterios objetivos deben ser independientes de la voluntad de cada una de las partes, además de legítimos y prácticos. Los criterios objetivos deben ser aplicables a ambas partes.
- Procedimientos Justos: Para generar un resultado que sea independiente de la voluntad, debemos utilizar criterios justos para así lograr la conciliación de los intereses en conflicto.

#### **5. La mejor alternativa para un acuerdo producto de la negociación.**

Tener en mente cuál es el resultado que se desea obtener negociando, y cuál es la mejor alternativa, para proteger a ambas partes del riesgo de aceptar condiciones desfavorables.

En vez de descartar una solución que no satisface nuestras metas, lo que debemos hacer es comparar la propuesta presentada con nuestra "Mejor Alternativa", y ver si satisface o no nuestros intereses.

Para elaborar nuestra "Mejor Alternativa", lo que debemos hacer es:

- Enumerar las medidas que se toman, en caso de que no se llegue a ningún acuerdo.



- Mejorar alguna de las ideas auspiciosas, y transformarlas en opciones prácticas.
- Seleccionar la opción que parece ser la mejor.

**Se busca llegar a un acuerdo con relación a los siguientes temas:**

- La aprobación para el inicio del proyecto.
- Cantidad de tiempo.
- Cómo se va a organizar el trabajo.
- Inversión.
- Gastos.
- Resultados.
- Confidencialidad.
- Garantía.

**• Aprobación para el inicio del proyecto:**

Informar las etapas y entregas del proyecto resumiendo las actividades principales a realizar en cada una de ellas.

Negociar acerca del momento indicado para realizar la firma del contrato para comenzar con el desarrollo del proyecto.

**• Cantidad de tiempo:**

Presentar de forma sencilla los aspectos técnicos de la infraestructura del proyecto.

Establecer como duración el tiempo mínimo planificado durante la etapa inicial del proyecto, que es de 9 meses desde que se comienza con la captura de requisitos hasta la entrega del producto final.

De acuerdo con ello, evaluar la respuesta del inversionista y, en el caso de tener que reducir el tiempo, reducirlo no más de 1 mes, para no comprometer la planificación y costos del proyecto.

**• Organización de trabajo:**

Presentar el equipo encargado del desarrollo del proyecto, y el perfil que desempeña cada uno.

Mostrar la metodología de trabajo empleada, aclarando que se realizan entregas parciales con incrementos del sistema una vez comenzado el desarrollo del mismo, debido a que se utiliza Metodología Scrum. Dichas entregas son de dos semanas, pudiendo así hacer un seguimiento del avance del desarrollo del sistema.

Es importante que el inversor sea considerado un integrante más del equipo, teniendo poder de decisión ya que él provee la fuente económica que hace posible la realización del proyecto.

● **Inversión:**

El inversor debe facilitar el monto necesario para poder comenzar la etapa inicial del proyecto. El proyecto tiene una inversión inicial del 40% del presupuesto, que tiene un retorno de inversión al año, a través de un estimativo del plan de suscripción. En el presupuesto se muestran los gastos de alquiler de coworking, pago de servidores, etc.

● **Gastos:**

Los gastos en los que incurre el desarrollo de la plataforma desagregados por recursos (personal y tecnología) los cuales son los pertinentes al pago mensual del personal, los servidores y equipos, y el espacio de coworking alquilado. Se muestra que los beneficios a obtener son superiores a los costos en que se incurre al desarrollar e implementar la plataforma.

Mostrar el plan de retorno de inversión del proyecto, en función de la cantidad de usuarios.

Mediante un estudio de costos realizado se estima un gasto total durante los 9 meses de desarrollo de la plataforma de \$1.455.706,99.

● **Resultados:**

Los resultados corresponden a la plataforma optimizada y sin fallas, mostrando las funcionalidades acordadas.

● **Confidencialidad:**

Mostrar la información manejada por las personas que componen el proyecto.

No es necesario mencionar nada sobre confidencialidad, ya que puede no ser bien visto por el inversor.

● **Garantía:**

Presentar la rentabilidad del proyecto en base al análisis de factibilidad económica realizado, la cual demuestra la viabilidad del proyecto. El retorno y el tiempo estimado de recuperación de la inversión.

**Estrategias de Negociación:**

- Realizar una presentación de proyecto, incluyendo integrantes del equipo, objetivos, plazos y fechas de entregas, inversión necesaria y las condiciones que se están buscando satisfacer. Está a cargo del *Product Manager* y el *Arquitecto* de la plataforma, respondiendo las preguntas técnicas del inversionista.

- Llegar una hora antes al lugar acordado para la reunión, para poder conectar los dispositivos, preparar la presentación y la demo a ser mostrada. Si los encargados de la reunión llegan justo a tiempo y se demoran en conectar los dispositivos, el inversor se puede molestar, ya que para ellos el tiempo es dinero.
- Asistir a la reunión ataviados con un look semi formal, mantener un tono amable y cordial, sin excederse en chistes o tocar temas de índole personales que pudieran incomodar al inversionista. La reunión debe durar un máximo de dos horas o menos, exponiendo el proyecto en la primera media hora. Luego, explicar sobre los distintos interesados en hacer uso de la plataforma, las garantías que promete la realización del proyecto y la cantidad de tiempo necesario para su desarrollo.
- Se deben mantener los plazos de desarrollo presentados, con una flexibilidad de hasta 15 días, garantizando la utilización de las herramientas elegidas en el proyecto y basándose en la metodología de trabajo establecida.
- Describir las posibles formas en que ambas partes se pueden beneficiar de la plataforma. Comentar sobre las necesidades que requiere el proyecto como inversión, lo que incluye, inversión inicial y gastos, así como también, los resultados esperados a corto y largo plazo.
- Cuando el inversionista realice una propuesta, negociar para que la inversión financie al menos un 80% de las necesidades planteadas. Cualquier monto inferior de inversión no es viable para la realización del proyecto.
- Si el inversionista propone la adición de alguna funcionalidad no contemplada con anterioridad, como parte del trato, se debe responder que se va a analizar con el equipo, y de ser aceptado, se agrega en el Backlog de desarrollo para llevarse a cabo, siempre y cuando no desvíe el objetivo principal del sistema y no incurra en una complejidad adicional que incida de manera significativa en los tiempos establecidos de desarrollo.
- Si el inversionista desea hacer preguntas con respecto a la confidencialidad, responder de forma amable y estar abiertos a considerar lo que se nos proponga.
- Al finalizar la reunión se le comunica al inversionista que su cuenta por lo consumido está pagada. En caso de llegar a un acuerdo, mencionar brevemente un resumen de todos los puntos acordados y enviarlos a su e-mail en un plazo no mayor a 24 horas de finalizada la reunión. Si no se llega a un acuerdo satisfactorio, despedirse con amabilidad dejando en claro que se agradece el tiempo que dedicado y dejando la puerta abierta, para trabajar en conjunto en el futuro.

**Aspectos a tener en cuenta durante la negociación:**

- Tener una charla informal al comienzo de la negociación.

- Tratar de forma despersonalizada los problemas que se mencionen.
- Controlar emociones y ser paciente si en algún momento surgen discusiones o desacuerdos.
- Escuchar atentamente a la otra parte, generar confianza y ser empático.
- Eliminar comportamientos negativos.
- Cambiar estrategia inicial si se identifican nuevas oportunidades.
- Tener en cuenta diferencias culturales e ideológicas, ya que el inversionista puede ser de otro país.

**Si la negociación se complica:**

- Simular tener una llamada de emergencia.
- Posponer la reunión para otro día con alguna excusa.
- Si no sabemos la respuesta a alguna pregunta, simular una llamada para pensar bien la respuesta.
- Estar dispuestos a ceder en algunos aspectos, para lograr alcanzar los objetivos más importantes.

**Argumentos en los que se basa la negociación:**

- El producto da una solución innovadora y de uso fácil para los usuarios.
- El producto presenta beneficios tanto para el lector como para el autor ya que no hay una plataforma en Argentina en la que puedan subir su material para darlo a conocer. Además, los lectores con una muy baja inversión mensual pueden disfrutar de contenido premium.
- El producto brinda un servicio a la comunidad para que puedan disfrutar de cómics, presentando la plataforma reportes con estadísticas que muestran la satisfacción e incremento de usuarios.

# TRABAJO PRÁCTICO

## INTEGRADOR N°2:

### *Gerenciamiento en T.I.*

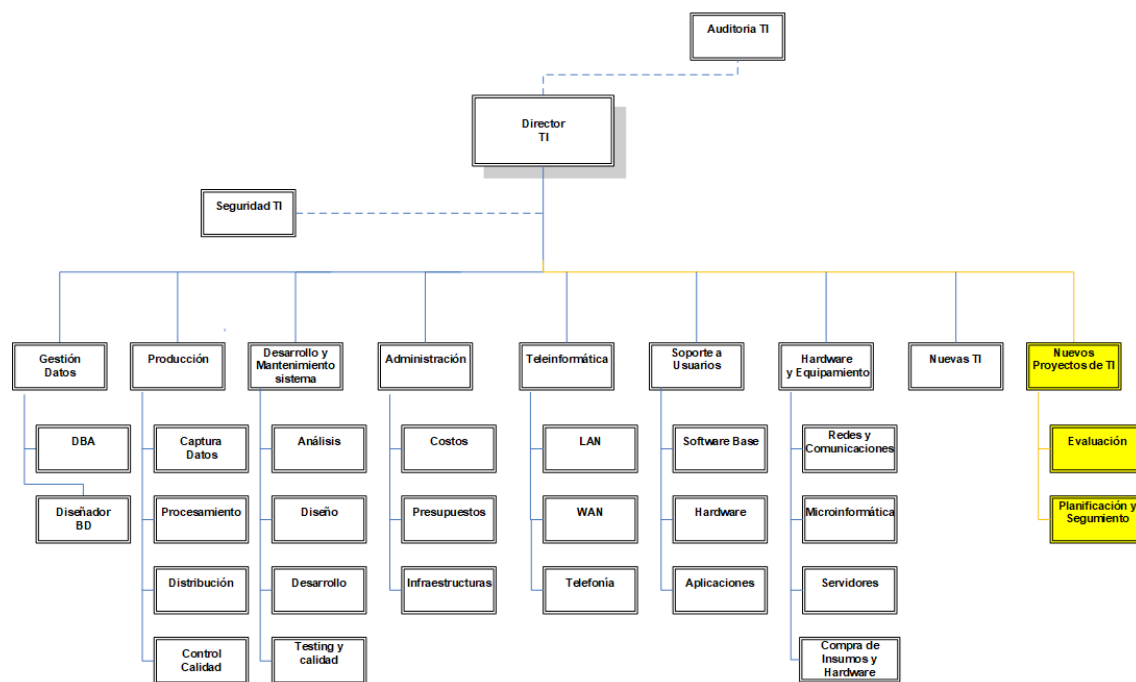


Figura 4. 1 Estructura organizativa de T.I., Data Center, áreas relacionadas y áreas de control.

**ÁREA SELECCIONADA:**

**Área Nuevos Proyectos de T.I.:** Encargada de evaluar nuevos proyectos necesarios, realizar la preevaluación, presupuestación, planificación y seguimiento.

## 1. Estructuras Organizativas

El tipo de estructura organizativa que se muestra en la **FIGURA 4. 1** es de: **Tipo Departamental**.

El diseño de Estructura Organizativa Departamental se refiere a la estructura organizacional de los departamentos o divisiones de la empresa, es decir, al esquema de diferenciación e integración existente en el nivel intermedio de la empresa. La diferenciación puede darse de dos maneras: Vertical y horizontal.

Además, esta estructura ayuda a los administradores a supervisar el rendimiento con mayor facilidad, dado que se puede realizar un seguimiento de los productos y actividades que la empresa desarrolla, permitiendo distinguir entre el servicio al cliente, la producción y la ubicación geográfica. Cada gerente puede concentrar los recursos específicos en su área y obtener mejores resultados.

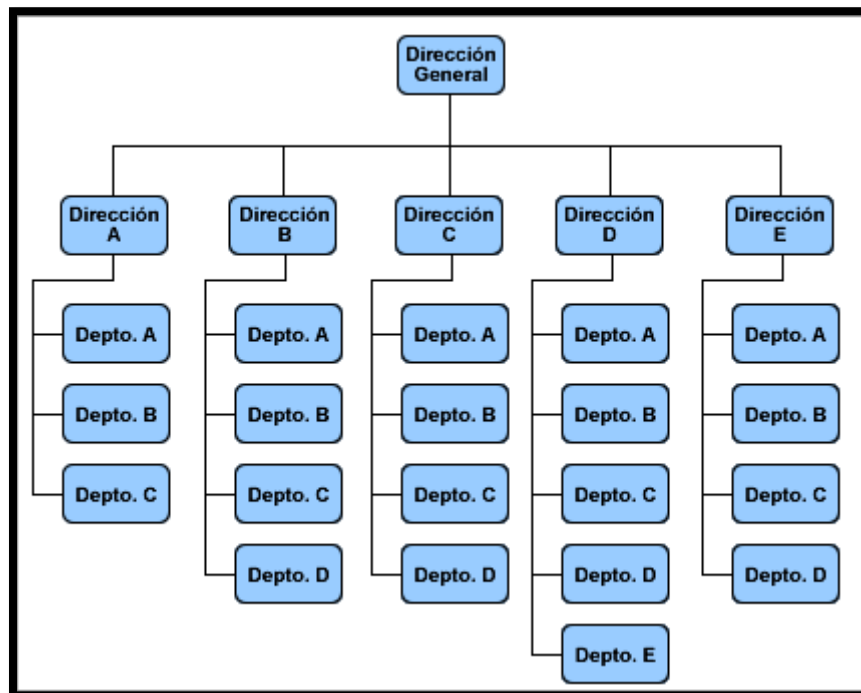


Figura 4. 2 Ejemplo de Organigrama de Tipo Departamental.

#### **Otros tipos de Estructuras Organizativas:**

- ❖ **Por Función:** La estructura organizativa funcional es la forma de departamentalización en la que las personas que se dedican a una actividad funcional, por ejemplo, finanzas o producción se agrupan en una unidad. Es una forma de organización empresarial tradicional, donde cada empleado tiene un superior, y los equipos son agrupados por especialidades.
- ❖ **Por Producto/Mercado:** Es la forma de organizar una empresa en divisiones que reúnen a las personas involucradas con un tipo dado de producto/mercado.
- ❖ **Matricial:** Estructura de organización en la que cada empleado depende tanto de un gerente funcional o de división, como de un gerente de proyecto o grupo.
- ❖ **Geográfica:** Organización a partir de las sedes geográficas, para empresas que operan a nivel internacional.
- ❖ **Por proyectos:** La organización cuenta con un equipo dedicado a tiempo completo y un director de proyecto que se sitúa al máximo nivel dentro de la organización. Habitualmente se estructuran también en unidades departamentales; sin embargo, todos ellos se reportan directamente al director de proyecto. Como se puede ver, se trata de una estructura organizativa muy sencilla y con ciertas limitaciones, como la dificultad de transferir conocimientos a través de proyectos.
- ❖ **Unidad de negocio:** Dentro de una empresa, es una unidad empresarial diseñada para fabricar y comercializar uno o más productos relacionados, dirigidos a mercados

muy específicos que requieren una oferta de productos muy delimitados, tal como si fuera un área especialista para esos mercados y productos.

- ❖ **Estructuras Organizativas Flexibles:** Una organización flexible es aquella que pretende, a corto plazo, utilizar los factores disponibles para adaptarse a los cambios inmediatos, mientras que, a largo plazo, se esfuerza por adaptarse al entorno competitivo, y lograr un desarrollo organizativo permanente.
  - **Organización virtual:** Es aquella que adoptan las empresas que mantienen únicamente un núcleo principal, subcontratando o cediendo mediante acuerdos de cooperación la mayoría o incluso la totalidad de sus actividades, ya sean productivas, comerciales, logísticas o de I+D.
  - **Estructura en trébol:** Modelo de organización que está constituido por un centro integrado por la alta dirección, a partir de la cual surgen las tres ramas u hojas del trébol. La primera hoja representa los trabajadores del núcleo, personas esenciales para la organización, que se ocupan de funciones críticas para las cuales la organización está especialmente dotada. La segunda hoja está compuesta por el conjunto de actividades productivas, comerciales y de servicios subcontratados a otras empresas. La tercera hoja del trébol es la fuerza de trabajo flexible, todos aquellos trabajadores de tiempo parcial y temporales.
  - **Organización por equipos o en racimos:** Se toma al equipo como unidad organizativa fundamental para la agrupación del trabajo. Tanto la toma de decisiones, como la planificación, organización y control del trabajo se descentralizan, pasando a ser llevadas a cabo por cada equipo de trabajo en todos los aspectos concernientes a la actividad que le es encomendada.
  - **Organización sin fronteras:** Tiene como objetivo intentar eliminar los límites verticales y horizontales de la estructura, rompiendo las barreras externas entre la empresa, sus clientes y sus proveedores.
  - **Organización federal:** Es una evolución de la forma multidivisional; las grandes empresas tienden a la desafección y externalización de subdivisiones, creando las llamadas "cuasi-empresas".

Las estructuras organizativas flexibles proponen un conjunto de características, clasificadas en dos grupos:

- **Estructurales:**
  - Estructuras planas, de base profesional y evolutivas.
  - Procesos técnicos de fabricación flexible y sistemas de gestión basados en tecnologías de la información innovadora.
  - Sistemas de información de apoyo a los procesos de decisión.
  - Función esencial de I + D en productos y procesos.



- Personas con la máxima movilidad profesional y geográfica.
- **De Comportamiento:**
  - Capacidad de reacción en las personas y en los sistemas.
  - Capacidad de previsión y de adaptación a los cambios.
  - Capacidad de asumir riesgos, lo que implica el conocimiento de los sistemas para su gestión.
  - Orientación al cliente y hacia la calidad total.
  - Capacidad para movilizar la empresa, para su rediseño y para la asignación de sus recursos.

## 2. Servicios del Área Nuevos Proyectos de T.I.

Dentro del área de Nuevos Proyectos de T.I. se pueden encontrar varios servicios externos o internos a la empresa que brinda a la misma. Entre ellos podemos destacar:

- ❖ Realizar planificación de cada Proyecto.
- ❖ Trabajar en la Ingeniería de Requerimientos.
- ❖ Dar el soporte técnico necesario para el desarrollo de nuevos proyectos, evaluando el impacto de los nuevos proyectos en el sistema instalado.
- ❖ Elaborar presupuestos.
- ❖ Supervisar los proyectos de desarrollo de software.
- ❖ Realizar instalaciones de hardware.
- ❖ Realizar las actualizaciones de red.
- ❖ Hacer el despliegue de la computación en nube y virtualización.
- ❖ Elaborar proyectos de gestión de datos y análisis de negocios.
- ❖ Realizar la implementación de servicios de TI.

## 3. Recomendaciones Técnicas y de Seguridad Física

Para el área de Nuevos Proyectos de T.I., las recomendaciones técnicas y de Seguridad física son las siguientes:

### Gestión del Data Center:

- ❖ **Recomendaciones Técnicas:** El Gerente del área fomenta la interactividad del equipo de Nuevos Proyectos T.I. con el área del Data Center, para así poder estar al tanto de las necesidades. Se crea un espacio en común para ambos equipos, donde pueden discutir temas relacionados con la tecnología, ayudando a crear una visión más amplia sobre ésta para así crear nuevos proyectos de T.I. que benefician al área del Data Center.

- ❖ **Seguridad Física del Data Center:** La seguridad física de los data centers implica proteger la infraestructura crítica de amenazas externas o intrusiones que atentan contra las actividades de una empresa. Se resguardan elementos de alto valor y de vital importancia para el desempeño de todas las áreas de la organización, tales como servidores, switches, y unidades de almacenamiento.

La seguridad está implementada por medio de la utilización de videovigilancia a través de cámaras, sistemas de control de acceso, y seguridad perimetral.

La seguridad física para los data centers se centra en 4 capas:

**Capa 1.** Seguridad Perimetral.

**Capa 2.** Seguridad de las Instalaciones.

**Capa 3.** Seguridad de la Sala de ordenadores.

**Capa 4.** Seguridad de los Racks.

#### **Capa 1. Seguridad Perimetral:**

- Acceso al Data Center controlado biométricamente.

El Data Center contiene el recurso más valioso de la compañía, por lo tanto, es de vital importancia que los datos que controla solo se pueden acceder por aquellos que pertenecen al área. Para evitar que intrusos ingresen físicamente al recinto, su entrada está resguardada por sensores biométricos.

#### **Capa 2. Seguridad de las Instalaciones:**

- Instalación de UPS, y estabilizadores de tensión.

El Data Center, asegura que los datos están a disposición de todas las áreas de la organización de manera constante. Por lo tanto, se previene cualquier falla de energía, mediante la instalación de UPS, y estabilizadores de tensión.

- Piso técnico.

Protege los cables, de las pisadas de los empleados.

- Sensores detectores de humo.

Colocados en el techo para prevenir posibles incendios.

- Extintores de fuego Clase C.

Están colocados dentro y fuera de la sala de ordenadores.

➤ Sistema de enfriamiento por hilera.

En el enfriamiento basado en hileras, las unidades CRAH (unidad de aire acondicionado para la sala de cómputo) se asocian con una hilera y, a los efectos del diseño, se consideran unidades dedicadas a una hilera. Las unidades CRAH se pueden ubicar entre los racks de equipos informáticos o se pueden instalar en altura. En comparación con la refrigeración tradicional basada en la sala sin contención, los caminos de flujo del aire son más cortos y están mejor definidos. Además, la circulación de aire es mucho más predecible, se puede utilizar toda la capacidad nominal de la unidad CRAH, y se logra una mayor densidad de potencia. Este tipo de refrigeración presenta varias ventajas adicionales, en las que, al reducir el trayecto del aire, también se reduce la potencia que tiene el ventilador de la unidad CRAH, con lo cual se aumenta la eficacia.

➤ Operaciones de seguridad.

Se incorporan operaciones de seguridad como: IDS/IPS, firewalls, honeypots, gestión antifraude, SIEM, DLPs.

➤ Cifrado.

Comprobar el cifrado de datos donde es susceptible interceptarlos, por ejemplo, en las copias de seguridad, o la posibilidad de interceptar el tráfico de explotación, que puede contener datos confidenciales como usuarios y contraseñas.

### **Capa 3. Seguridad de la Sala de Ordenadores:**

El objetivo de esta tercera capa de seguridad física es la de restringir el acceso a través de diferentes métodos de verificación de acceso, monitorear todos los accesos autorizados.

El acceso a la sala de ordenadores del Data Center está restringido a un reducido grupo de empleados. Existen diversos métodos para restringir el acceso a esta área, y estos pueden ser:

- Tarjetas magnéticas HID.
- Cerradura eléctrica.
- Cerradura biométrica.
- Cerradura electrónica digital.
- Cerradura inteligente Smart.

#### **Capa 4. Seguridad de los Racks:**

Esta última capa de seguridad es importante y efectiva para minimizar las amenazas internas. Se hace hincapié en el control y aseguramiento de los racks. Permite reducir una posible fuga de información causada por un empleado con malas intenciones.

Recomendaciones a tener en cuenta:

- Videovigilancia IP.
- Sistemas de bloqueo electrónico para racks de servidores.

## **4. Retroalimentación a 360° del Área Nuevos Proyectos de T.I.**

Con la retroalimentación de empleados estándar (llamada de 180 grados), el empleado es controlado por sus supervisores y pares. Su jefe y colegas son responsables de escribir un informe que contenga detalles sobre la ética de trabajo del empleado, su rendimiento profesional, sus habilidades de equipo, y otros factores importantes. Para tener una retroalimentación de 360°, se añaden 180 grados adicionales cuando el empleado obtiene respuestas de los clientes con los que trabaja.

La Retroalimentación de 360° sirve para saber cómo se desenvuelve la gerencia del área, mediante evaluaciones de los subordinados, colegas y superiores.

Luego de la implementación de un nuevo proyecto interno, el área de Nuevos Proyectos de T.I., decide realizar una evaluación 360° para conocer el desempeño del jefe de área. Para esto se le comunica el tipo de evaluación que se le realiza a él y a todos los que participen.

En primera instancia se define la estructura de la retroalimentación (cuestionarios, reuniones personales, entrevistas, etcétera) y luego las competencias o habilidades que se tienen en cuenta. Las fuentes de información que se aplican en el área son las siguientes:

- ❖ **Retroalimentación desde arriba (director T.I.):** Se realiza una reunión con el gerente para evaluar resultados a nivel organizacional. Luego de la implementación de este proyecto, el director de T.I. informa personalmente su propia opinión clara respecto al desempeño del jefe de área, en distintos aspectos (técnicos y humanos). Una vez concluida la reunión, se discuten los aspectos a mejorar.

- ❖ **Retroalimentación desde los costados (Colegas):** En el caso del área de “Desarrollo y Mantenimiento de Sistemas”, mediante un cuestionario ellos responden sus niveles de satisfacción respecto a la implementación del nuevo proyecto. Al saber las respuestas de esta área, el jefe de Nuevos Proyectos de T.I. concluye mejoras a realizar en un futuro pensando en sus colegas.
  
- ❖ **Retroalimentación desde abajo (Subordinados):** También, mediante un cuestionario, los subordinados expresan su opinión en lo referente a estos aspectos:
  - Si su opinión se tiene en cuenta a la hora de realizar el nuevo proyecto.
  - Si opinan que el resultado final del nuevo proyecto es el correcto.
  - Si se sienten cómodos respecto al liderazgo ejercido por el jefe de área.
  - Si piensan que hay mejoras a implementar en su área luego del nuevo proyecto.
  
- ❖ **Retroalimentación de Fuentes Informales:** Las fuentes informales tienen su origen en las conversaciones de los empleados, y clientes de la empresa y son de alto valor, porque tienen mucha relación con los productos, proyectos y nuevas estrategias.

Son Fuentes Informales las siguientes:

- Clientes.
  - Empleados de las diferentes áreas de la empresa.
  - Empresas subcontratadas.
  - Competidores.
  - Inversores.
  - Las ferias, exposiciones, exhibiciones.
  - Congresos, seminarios, jornadas.
  - Misiones empresariales.
  - Comités y asociaciones empresariales.
- 
- ❖ **Retroalimentación de Organismos de Control:** El control es una de las etapas que forman el proceso administrativo, en la cual se puede tener una información más precisa de lo que sucede. Es la función que consiste en medir y corregir el desempeño individual y organizacional para asegurar que los hechos se ajusten a los planes y objetivos de las empresas. Implica medir el desempeño contra las metas y los planes, muestra donde existen desviaciones con los estándares y ayuda a corregirlas.

- ❖ **Retroalimentación de Informes de Auditoría:** Este procedimiento se aplica a la realización de Auditorías Internas sobre todos los procesos y las actividades comprendidas dentro de la empresa para verificar el cumplimiento, la eficacia y el desempeño de la misma.

## 5. Coaching Eficaz en el Área de Nuevos Proyectos de T.I.

Debido a que del área de Nuevos Proyectos de T.I. surgen los nuevos proyectos a ser implementados en la organización, es vital prestar atención al resto de todas las áreas para poder detectar necesidades del gerente, de jefes y de colegas a cubrir. Es por esta razón que en el área es necesario que exista un espíritu de trabajo en equipo para trabajar de forma eficiente. Esto se logra mediante el coaching eficaz.

Cada uno de los integrantes del equipo tiene espíritu de coaching y presentan características que los hacen ser referentes a seguir. Algunos de los rasgos principales son los siguientes:

- ❖ Saber escuchar de distintas fuentes.
- ❖ Estar atento a lo que le ocurre o piensa cada persona de su equipo.
- ❖ Acompañar a cada miembro de su equipo en situaciones difíciles o que no se sabe cómo continuar.
- ❖ Proveer los recursos necesarios.
- ❖ Contener anímicamente.
- ❖ Ayudar en todo lo que es necesario para cada persona del equipo.

Además de las características anteriores, se siguen los siguientes principios para lograr una base estable en las relaciones personales:

- ❖ No criticar las nuevas ideas sugeridas por los colegas, no hacer comentarios de forma tendenciosa, sino explicar su causa y ayudar a encontrar la solución.
- ❖ Respaldar los esfuerzos de los subordinados, y colegas.
- ❖ Construir la confianza de los integrantes del equipo.
- ❖ Debe tener autoconciencia e inteligencia emocional.
- ❖ Debe mostrar empatía.
- ❖ Realizar preguntas poderosas y tener curiosidad.
- ❖ Debe tener escucha activa.
- ❖ Poseer comunicación efectiva y directa.
- ❖ Debe reconocer el esfuerzo de sus subordinados.
- ❖ Facilitar el aprendizaje de sus subordinados.
- ❖ Debe ser responsable de los resultados obtenidos.
- ❖ Asistir y facilitar las actividades del equipo.

En base a los principios, y rasgos nombrados anteriormente se analiza el Coaching Eficaz en cada subordinado del área:

### 1. Evaluación (2).

Su trabajo es realizar la evaluación de la posibilidad de llegar a implementar los nuevos proyectos propuestos por el área. Para lo cual debe tener las siguientes habilidades:

- Preparar un presupuesto.
- Tener conocimiento general de la tecnología empleada en el proyecto.
- Generar ideas y soluciones.
- Conocer el funcionamiento de todas las áreas de la organización.
- Controlar la calidad del proyecto.

### 2. Planificación y Seguimiento (2).

Su trabajo es realizar la planificación y seguimiento de los nuevos proyectos propuestos por el área. Para lo cual debe tener las siguientes habilidades:

- Conocer todas las áreas de la organización.
- Tener la capacidad de transmitir conocimientos.
- Estar abierto a otros puntos de vista.
- Estar dispuesto a que no todo salga según lo planificado, y saber actuar en consecuencia.
- Tomar decisiones.
- Desarrollar el trabajo en equipo, apoyado en la comunicación, la confianza, el respeto, la ética, las recompensas, y la resolución de conflictos.

## 6. Equipo de Trabajo Efectivo y Equilibrado

### EQUIPO EFECTIVO

#### Características del Equipo Efectivo:

- Libre expresión de todos los miembros.
- Principio del trabajo en conjunto, que se logra mediante una delegación eficaz del líder, generando sinergia entre los miembros del equipo de trabajo, cuando los resultados del trabajo en conjunto son mejores que los resultados del trabajo individual.
- Todos están dispuestos a asumir riesgos, ya que hay una adecuada planificación, y gestión de riesgos de parte del líder.
- Existe coaching eficaz.
- Hay objetivos comunes y metas claras bien arraigados en todos los miembros.

- Existen iniciativas, deseos y voluntad de participación, respeto por todos y siempre los miembros están dispuestos a colaborar.
- Aceptación de decisiones por consenso general, aun cuando existan divergencias individuales.
- Buena relación de los miembros con otros integrantes de otros proyectos y otras áreas, para aprovechar las experiencias ajenas, y poner en valor las propias.
- Retroalimentación de todos los integrantes del equipo de trabajo a los efectos de pensar, y poner en práctica permanente acciones de mejora continua.

**En Equipos Efectivos se siguen los principios detallados a continuación:**

- ❖ Libre expresión de todos los miembros, retroalimentación de todos los integrantes del equipo de trabajo, y aceptación de decisiones por consenso general: Cuando se desea proponer un nuevo proyecto que implica la mejora de algún área de la organización, se lleva a cabo mediante un debate entre los miembros. En dicha discusión, cada integrante expresa sus inquietudes con total libertad, tanto al hablar de los aspectos positivos de la realización del nuevo proyecto, como de aquellos inconvenientes que se encuentran para la implementación.
- ❖ Principio del trabajo en conjunto: Se delegan las tareas teniendo en cuenta quién es el individuo más calificado para llevarla adelante o, en su defecto, quién será el que resulte más beneficiado con el aprendizaje. Por ejemplo, se pueden asignar tareas como buscar y organizar la información necesaria para el relevamiento y análisis. Mientras que las finales, como realizar el diseño y desarrollo, se pueden asignar al diseñador y al desarrollador más experimentado para garantizar que las tareas se lleven adelante con eficacia y rapidez.

A todos los integrantes se les asignan tareas de acuerdo a su nivel de experiencia, de forma equitativa, donde la motivación principal reside en que aquellos con mayor experiencia guíen y eduquen a aquellos con menor experiencia. Ningún miembro de los distintos equipos de trabajo debe estar saturado.

El trabajo colaborativo eficiente se logra mediante la delegación adecuada por parte del líder de proyecto o del equipo de trabajo, y la sinergia generada entre los miembros que lo constituyen.

Además, cada integrante debe tener en claro, en todo momento, cuál es su trabajo, y responsabilidad respecto a las tareas que le competen.



- ❖ Hay objetivos comunes y metas claras en todos los miembros: Todos los integrantes del área tienen presente que el objetivo de su trabajo es crear, mantener y mejorar los proyectos de la organización. Y desempeña su rol de acuerdo a una porción de tareas del total que conlleven a la concreción del proyecto o equipo en el que se encuentra trabajando. Por ende, se debe asegurar de que cada tarea que realice un integrante del equipo no aleje al grupo de la concreción de dichos objetivos. Nuevamente, cabe resaltar que cada individuo debe tener en claro su responsabilidad y función a desempeñar para lograr los objetivos, tanto particulares como globales.
  
- ❖ Existen iniciativas, deseos y voluntad de participación: La motivación de los integrantes es un aspecto fundamental para repartir las tareas. Por ejemplo, si se delega la tarea de hablar con los empleados de otras áreas para recabar información sobre un proyecto, problema, o necesidad existente, se envía al integrante más informado y con mejor habilidad para comunicar. Se tienen en cuenta las propuestas de innovación de cada uno de los integrantes para aplicar a los proyectos actuales y futuros, generando así un ambiente de participación donde cualquier individuo puede realizar sus aportes y, aún más importante, sentirse íntegro y en pertenencia respecto al proyecto.

Cada relación e intercambio que surge de la relación entre todos los miembros en todos los equipos de trabajo debe estar fundamentada en el respeto, ante todo.

- ❖ Buena relación de los miembros con otros integrantes de otros proyectos y otras áreas: Todo miembro es consciente de la existencia de los demás proyectos, distintos al que se encuentra trabajando actualmente, por lo que la generación de vínculos con los demás integrantes es fundamental. Todo individuo, además de poder integrarse a otros equipos de trabajo y cambiar de proyecto, tiene conocimiento de la experiencia de los demás y sus capacidades para resolver determinados problemas. Así, cuando se enfrentan requerimientos o problemas complejos en un proyecto, como la integración de una nueva tecnología, patrón de diseño o, incluso, la utilización de buenas prácticas se puede consultar a miembros de otros equipos que tengan la experiencia suficiente para acatar el problema en cuestión.

Se generan reuniones constantemente para intercambiar experiencias entre diseñadores y desarrolladores, y a su vez mejorar las relaciones interpersonales.

## **EQUIPO EQUILIBRADO**

### **Características del Equipo Equilibrado:**

- Cantidad de integrantes, de acuerdo con recomendaciones de alcance de control del líder.
- Disponibilidad de tiempo.
- Necesidades personales y fines propios.
- Actitud (positiva, negativa, colaboración, egoísta, etc.)
- Roles (orientado a la tarea, orientado a la relación, etc.)
- Personalidad (introvertido, extrovertido, agresivo, sumiso, solitario, etc.)
- Ingenio, creatividad, generación de ideas, inquietudes, nuevos proyectos, etc.
- Competencias técnicas y nivel de capacitación.
- Adaptabilidad al stress.

### **En Equipos Equilibrados se siguen los principios detallados a continuación:**

- ❖ Cantidad de integrantes: Se cuenta con cuatro integrantes además del Gerente: 1 analista, 1 diseñador, y 2 desarrolladores. Este número está dentro del alcance de control sugerido.
- ❖ Disponibilidad de tiempo: Todos los integrantes, independientemente del equipo de trabajo en el que se desempeña, trabaja 8 horas. Se tienen en cuenta excepciones en casos particulares para permitirle a los miembros compensar trabajo en el futuro, en caso de que el proyecto y el tiempo lo permitan.
- ❖ Necesidades personales y fines propios: Cabe señalar que, aunque algunos integrantes del área perciben su empleo como una forma de subsistencia para ellos y sus familias, por lo que hacen todo lo posible para llevarlo a cabo de la mejor manera y conservar esa estabilidad económica. Por el contrario, otros empleados lo ven como una gran oportunidad para adquirir conocimiento y experiencia profesional por lo que son más proactivos a encarar tareas desconocidas y a investigar tecnologías novedosas que pueden aplicar a la elaboración de nuevos proyectos de T.I. para mejorar el rendimiento de todas las áreas de la organización.
- ❖ Roles: El rol del analista es realizar el relevamiento de los nuevos proyectos de T.I., el diseñador se encarga de diseñar el sistema y definir la arquitectura a utilizar, los desarrolladores son los encargados de la codificación, el testing e implementación del nuevo sistema, y puesta en producción. Como son tareas en las que hay que tomar decisiones y se tiene tiempo para decidir, lo ideal es aplicar un liderazgo orientado a las personas.
- ❖ Personalidad: El equipo del área tiene personalidades variadas. Algunos, como los desarrolladores, gustan de tareas a más alto nivel y son propensos a trabajar en grupo.

Sin embargo, el diseñador es más introvertido y prefiere trabajar en algunas ocasiones de manera individual. Un aspecto por mejorar es que la mayoría de los integrantes tienen una personalidad dominante, y apasionada lo que complica llegar a acuerdos cuando surgen discusiones.

- ❖ Competencias técnicas y nivel de capacitación: El gerente cuenta con 12 años de experiencia en el rubro T.I., el diseñador tiene también experiencia profesional, al igual que los desarrolladores, en cambio el analista está recién comenzando su carrera. Los primeros ofrecen ventajas como la rapidez para resolver problemas rutinarios, y un conocimiento técnico amplio debido al largo tiempo que lleva ocupando ese rol. Mientras que el empleado más novato ofrece al equipo un punto de vista más innovador para resolver problemas recientes, y una motivación extra para su trabajo que es la adquisición de conocimientos.
- ❖ Crear un equipo de trabajo agradable y equilibrado: Mantener el equilibrio en un equipo de trabajo no es una tarea titánica, pero sí requiere esfuerzo para generar un equipo equilibrado y unido, y así obtener beneficios no solo productivos, sino que personales. Suele suceder que en las empresas sólo conocemos el lado profesional de los compañeros, lo que cierra puertas a generar relaciones más ricas y profundas, que son tan necesarias a la hora de realizar modificaciones al equipo de trabajo. Si el Equipo no está equilibrado y consolidado, ante cambios de integrantes puede verse seriamente perjudicado en su sinergia y productividad. En un grupo de trabajo hay que dar espacios para conocerse mejor, y así saber qué es lo que el otro considera importante o lo que puede afectarle negativamente. El crear actividades grupales lúdicas fuera del ambiente formal de la empresa es una buena idea, pero no basta con una sola jornada, sino de convertirlo en una costumbre que no tiene por qué ser cara o elaborada. Celebrar cumpleaños u ocasiones importantes, almuerzos de camaradería o incluso encuentros familiares, logran que el grupo de trabajadores pase a ser un equipo en que las personas se aprecian más allá de lo meramente profesional.
- ❖ Para lograr que el equipo se mantenga equilibrado: Se tiene en cuenta los roles a desempeñar por cada miembro en el grupo. La efectividad de los equipos de trabajo está directamente relacionada con la existencia equilibrada de los roles en los equipos de trabajo. Un equipo compuesto por miembros brillantes e ingeniosos tiene un desempeño deficiente si dichos miembros no cuentan con habilidades alternativas a su ingenio creativo. Este planteamiento genera la hipótesis del "equilibrio de roles", la cual afirma que aquellos equipos que aúnen una representación más equilibrada de los roles propuestos a nivel de "roles naturales", desempeñan mejor las tareas. Si se presenta un equilibrio de roles, se reduce la probabilidad de una sobresaturación de un rol específico, evitándose de este modo un solapamiento de roles, lo cual puede

facilitar la aparición de conflictos en el equipo, disminuyendo los niveles de eficiencia grupal.

➤ **Los Roles en un equipo son:**

- Mentales:
  - Cerebro: Aporta ideas novedosas al equipo. Fortalezas: Es creativo, imaginativo, y heterodoxo. Debilidades: Pasa por alto los detalles y le cuesta comunicarse con eficacia.
  - Especialista: Desde su área contribuye con un dominio técnico. Fortalezas: Es cumplidor, dedicado, y ordenado. Debilidades: Aporta ideas sólo en áreas muy concretas, y se enrolla en tecnicismos.
  - Monitor-Evaluador: Analiza y evalúa todas las opciones posibles que son pertinentes para el equipo. Fortalezas: Es serio, perspicaz, y estratega. Debilidades: Carece de iniciativa, y de habilidad para motivar.
- Sociales:
  - Coordinador: Coordina los esfuerzos del equipo, y distribuye las tareas y responsabilidades. Fortalezas: Es maduro, confiado, y fiable. Debilidades: Se le ve como manipulador, se descarga de su trabajo personal.
  - Investigador de Recursos: Establece contactos con el exterior, para gestionar las contribuciones necesarias para el equipo. Fortalezas: Es extrovertido, entusiasta, y comunicativo. Debilidades: Demasiado optimista, pierde el interés al desaparecer el entusiasmo inicial.
  - Cohesionador: Se preocupa por las necesidades de los miembros, y promueve la unidad en el grupo. Fortalezas: Es cooperativo, apacible, y diplomático. Debilidades: Indeciso en situaciones críticas e importantes.
- Acción:

- Impulsor: Impulsa al equipo a la acción. Fortalezas: Es retador, dinámico, y adicto al trabajo. Debilidades: Propenso a provocar, hierde los sentimientos.
- Implementador: Transforma las ideas en acciones concretas. Fortalezas: Es disciplinado, conservador, y eficiente. Debilidades: Inflexible a veces, y lento al responder a nuevas posibilidades.
- Finalizador: Corrige los errores y se preocupa por el cumplimiento de los plazos. Fortalezas: Es esmerado, concienzudo, y ansioso. Debilidades: Tiende a preocuparse indebidamente, y es reacio a delegar.

## 7. Funciones del Tablero de Comandos para el Área de Nuevos Proyectos de T.I.

El tablero de control o tablero de comando, también llamado CMI (Cuadro de Mando Integral), es una herramienta de control que permite agrupar y monitorear un conjunto de indicadores que ayudan de manera rápida a conocer los principales números que se han decidido usar. El objetivo es diagnosticar adecuadamente las situaciones que se presenten en una empresa. Ese diagnóstico y monitoreo permanente de los indicadores y de la información, sirve para mantener el control en distintas disciplinas. A través de luces de colores y alarmas se puede ver el estado actual de las actividades y poder tomar una decisión.

El tablero de control es una metodología clara para enseñar a los directivos a organizar y configurar la información. Es necesario brindar metodologías gerenciales para que las empresas no sólo se basen en su intuición y el conocimiento de sus directivos.

### **A. Tablero de Control Operativo:**

Es una herramienta de dirección que permite realizar el seguimiento y la monitorización de procesos operativos para detectar situaciones críticas y diagnosticar estados de desempeño. Permite también, hacer un seguimiento del estado de situación de un sector o proceso de la empresa, para poder tomar a tiempo las medidas correctivas necesarias.

### **B. Tablero de Control Directivo:**

Es el panel que abarca a toda la empresa en su conjunto segmentándola por áreas clave y permitiendo monitorear la evolución de cada área a través de indicadores específicos. Da la capacidad de diagnosticar de forma rápida la situación global de la

empresa hacia adentro a través de cifras, coeficientes, porcentajes y todo tipo de indicadores.

### **C. Tablero de Control Estratégico:**

Es importante medir la situación actual, cambios y avances para tomar decisiones y realizar correcciones a los planes y las acciones. Un tablero de control debe estar diseñado con el fin de ayudar a alcanzar el objetivo fijado, que es obtener una mayor rentabilidad para generar el mayor valor percibido posible al menor costo. Brinda información interna y externa necesaria para conocer la situación y evitar llevarse sorpresas desagradables con respecto al posicionamiento estratégico y a largo plazo de la empresa.

### **D. Tablero de Control Integral:**

Es la consolidación en una "unidad integrada" de los tres tipos de Tableros: operativo, directivo y estratégicos, para ser usada por los primeros niveles directivos de una empresa en orden a avanzar en una dirección estratégica sin papeles. Esto implica que la información más relevante para dirigir se encuentre organizada y sistematizada en una sola herramienta tecnológica e incorporada en el proceso directivo.

De acuerdo al área elegida de Nuevos Proyectos de T.I., el tablero de comando más apropiado es el **Tablero de Control Operativo**.

### **Funciones del Tablero de Control del área Nuevos Proyectos de T.I.:**

- ❖ Validar y filtrar los datos a incorporar de cada fuente (planes, tareas, personal, jefes, colegas, equipos, procesos, sistemas, entre otros) que se requieren para nuevos aspectos del proyecto.
- ❖ Automatizar los procesos en la incorporación de datos que limitan el crecimiento de los proyectos dentro de la organización. Esto permite más velocidad en los procesos, escalabilidad sin errores, menores costos, implementación inmediata y un diagnóstico personalizado.
- ❖ Automatizar la gestión, organización y proyección de los procesos que se realizan en el área de TI.
- ❖ Parametrizar las reglas de control, valores objetivos, información por excepción, alertas, detalles y relación automática de otros sistemas con el área de nuevos proyectos.
- ❖ Simular decisiones para corregir posibles fallas en la organización.
- ❖ Planificar y establecer los objetivos de los nuevos proyectos a implementar.

- ❖ Realizar el seguimiento, control y retroalimentación de los proyectos.
- ❖ Evaluar el comportamiento del nuevo proyecto de TI.
- ❖ Medir el nivel de satisfacción de los usuarios con las nuevas características.

**Ejemplos de Tableros de Control Operativo para el área T.I.:**

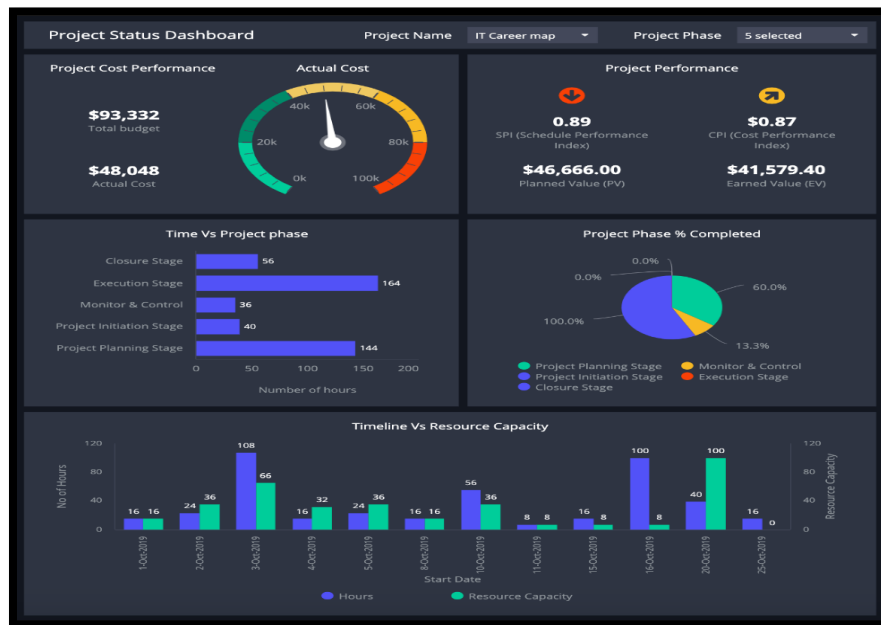


Figura 4. 3 Ejemplo de Tablero de Control Operativo (a)



Figura 4. 4 Ejemplo de Tablero de Control Operativo (b)

**Ejemplos de Tableros de Control Operativo para el área de Nuevos Proyectos de T.I.:**

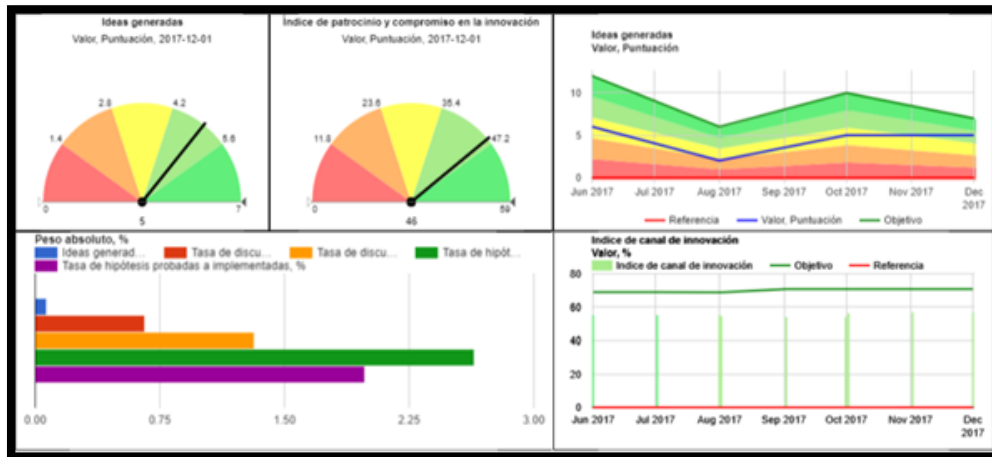


Figura 4. 5 Ejemplo de Tablero de Control del área de Nuevos Proyectos de T.I.

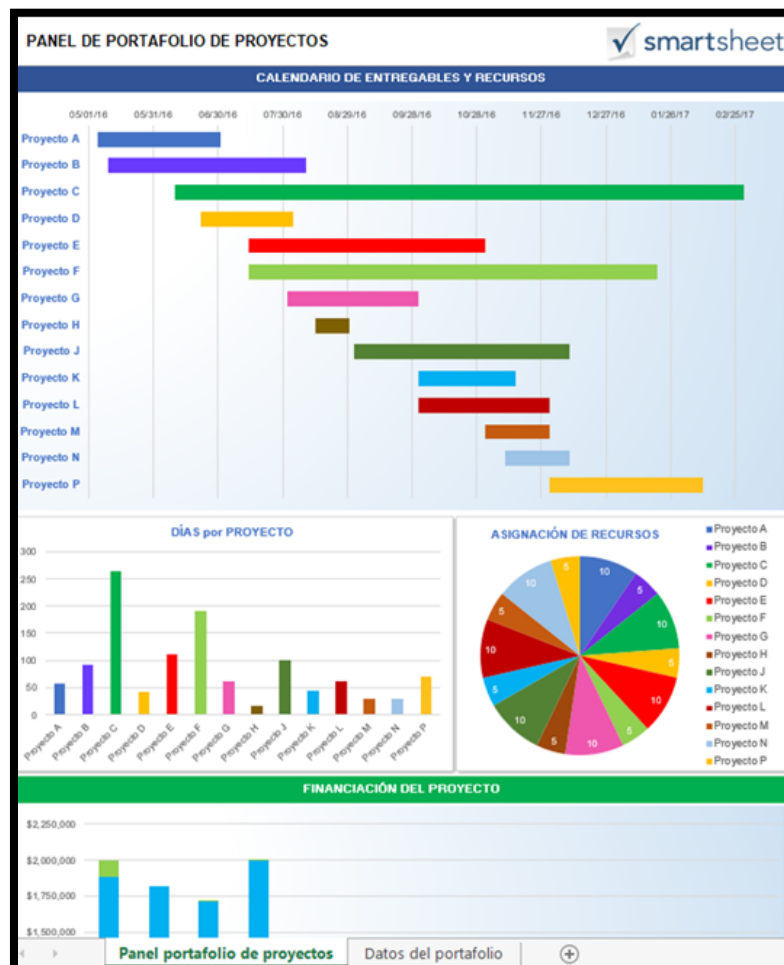


Figura 4. 6 Ejemplo de tablero de control del área Nuevos Proyectos de T.I.





## 8. Estrategia de Mejora: Área Nuevos Proyectos de T.I.

Se ha elegido separar la estrategia de mejora en dos años. Cada año incluirá aspectos relacionados al trabajo en equipo y la capacitación del personal para las tareas que debe realizar cotidianamente.

### PRIMER AÑO

<i>Meses</i>	<i>Actividades</i>	<i>Cantidad de Actividades</i>
Enero	<ul style="list-style-type: none"> <li>❖ Informar a los empleados del área sobre las intenciones de mejora que se quieren realizar.</li> <li>❖ Comunicar de manera clara y precisa los objetivos que se desean alcanzar.</li> <li>❖ Análisis de la situación actual del proyecto.</li> </ul>	3
Febrero	<ul style="list-style-type: none"> <li>❖ Realizar encuestas de motivación personal.</li> <li>❖ Realizar encuestas de trabajo en equipo.</li> <li>❖ Realizar encuestas sobre la situación actual del área seleccionada.</li> </ul>	3
Marzo	<ul style="list-style-type: none"> <li>❖ Evaluar y analizar los resultados obtenidos de las encuestas.</li> <li>❖ Reforzar los puntos que han dado inferiores a los que se esperaba o incluso aquellos en los que el personal tiene interés e incentivo de cambiar</li> <li>❖ Realizar actividades para incentivar a las personas y que lo puedan aplicar a su área de trabajo.</li> <li>❖ Definir reuniones con los encargados de las áreas.</li> <li>❖ Proveer un espacio de comunicación para todos los integrantes del área, y comentar los puntos importantes que se consideraron en la reunión anterior.</li> </ul>	5
Abril	<ul style="list-style-type: none"> <li>❖ Incorporar el coaching eficaz en aquellos procesos donde se detectaron la mayor cantidad de fallas para mejorar los resultados obtenidos en las evaluaciones.</li> </ul>	2

	<ul style="list-style-type: none"> <li>❖ Analizar los cambios observados luego de realizar el coaching eficaz y documentarlos.</li> </ul>	
Mayo	<ul style="list-style-type: none"> <li>❖ Se capacita a todo el personal para afrontar los problemas, y distintas alternativas que se pueden tomar ante el surgimiento de inconvenientes presentados en el área.</li> <li>❖ Controlar el tiempo actual para ver si fue correcta la planeación, y acciones del equipo que se determinaron en un principio.</li> </ul>	2
Junio	<ul style="list-style-type: none"> <li>❖ Capacitación relacionada a la originalidad de nuevos proyectos, teniendo en cuenta que ya existen diversos tipos de proyectos, pero que algunos representan un gran reto porque no se han realizado antes. Se trabaja la importancia de la innovación y la creatividad.</li> <li>❖ Se realizan nuevas encuestas que permiten conocer si han cambiado los puntos que se buscaban mejorar.</li> </ul>	2
Julio	<ul style="list-style-type: none"> <li>❖ Estudio de benchmarking en el que se pueda averiguar las tecnologías que están siendo usadas actualmente en el área de Nuevos Proyectos de T.I.</li> </ul>	1
Agosto	<ul style="list-style-type: none"> <li>❖ Se modifican las actividades y procesos que han tenido más inconvenientes, y se documenta en los manuales de usuario todos los cambios y mejoras propuestas con el fin de que sean accesibles por toda persona que lo requiera.</li> <li>❖ Se informan todos los cambios implementados al personal, capacitando al mismo en las nuevas actividades que deben realizar, remarcando las diferencias que posee y comunicando las mejoras que se obtienen debido a las nuevas actividades desempeñadas.</li> </ul>	2
Septiembre	<ul style="list-style-type: none"> <li>❖ Con el fin de lograr la inclusión de todos los participantes se realizarán actividades recreativas</li> </ul>	2

	<p>tales como salidas after office, torneos de ping-pong, juntadas cinéfilas, y truco.</p> <ul style="list-style-type: none"> <li>❖ Realizar una dinámica para desarrollar ideas creativas. Por ejemplo, armar tres grupos en el que hay que escribir cinco ideas en un tiempo de cinco minutos.</li> </ul>	
Octubre	<ul style="list-style-type: none"> <li>❖ Participar de charlas sobre técnicas y metodologías que permitan construir grupos humanos más cohesionados motivados para cumplir sus tareas. Bajo los principios de complementariedad y solidaridad, se sienten más seguros para afrontar cualquier proyecto.</li> <li>❖ Realizar reuniones con los jefes de áreas relacionadas a la nuestra, para conocer la situación de las mismas e informar los cambios y modalidades nuevas que han sido implementadas en nuestra área.</li> </ul>	2
Noviembre	<ul style="list-style-type: none"> <li>❖ Realizar la evaluación de desempeño de los integrantes del área.</li> </ul>	1
Diciembre	<ul style="list-style-type: none"> <li>❖ Realizar el dictado de talleres, y cursos dentro de las instalaciones.</li> <li>❖ Generar un debate sobre tópicos que surgen en un equipo para poder profundizar y salir de la zona de confort, y luego llegar a una conclusión en la que se pueda hacer hincapié en las mejoras.</li> </ul>	2

Tabla 4. 1 Actividades del primer año.

## SEGUNDO AÑO

<b>Meses</b>	<b>Actividades</b>	<b>Cantidad de Actividades</b>
Enero	<ul style="list-style-type: none"> <li>❖ Capacitar a los integrantes del área sobre dinámicas de grupo que fomenten los hábitos de trabajo y faciliten la coordinación de un mismo equipo.</li> </ul>	2

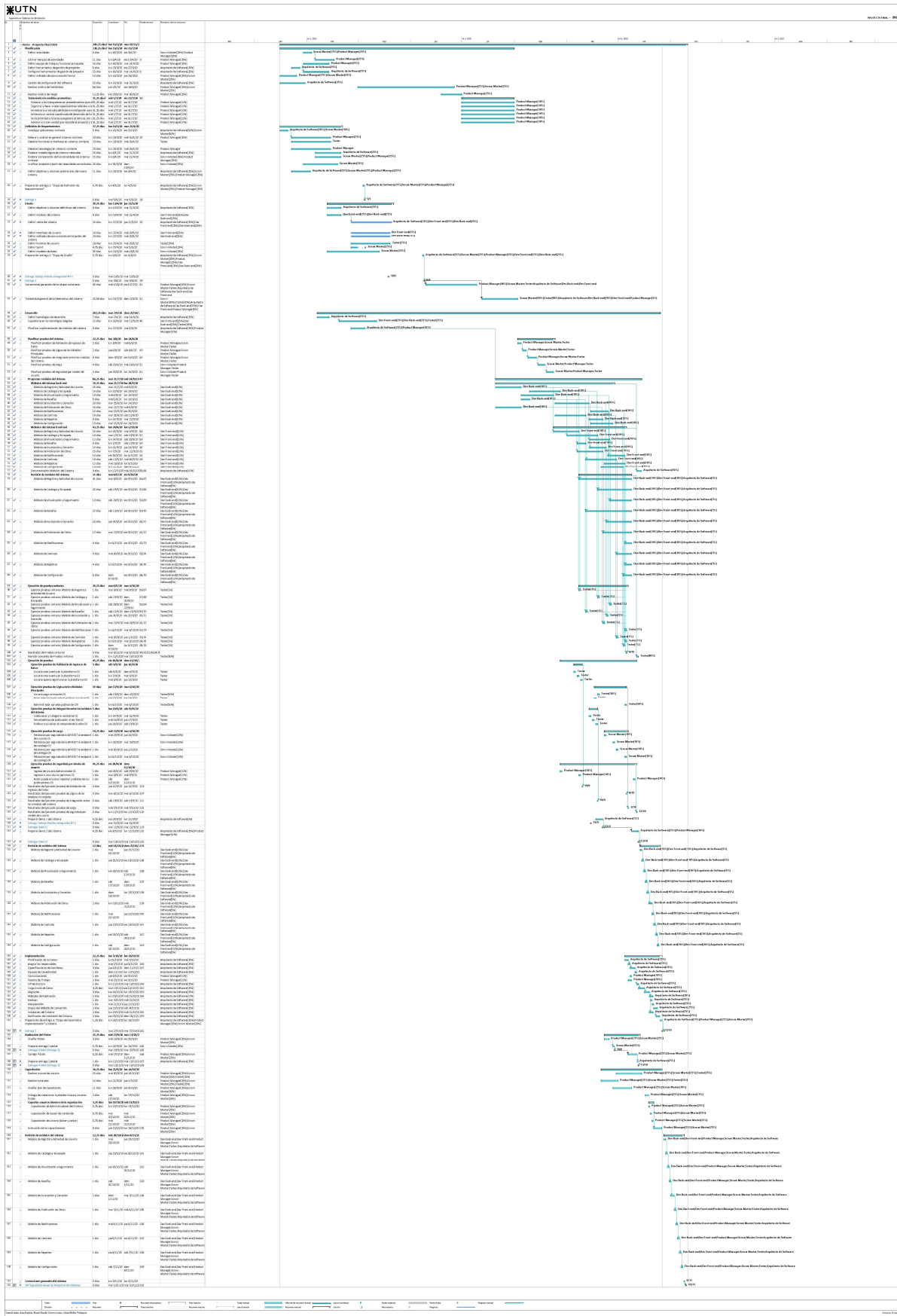
	<ul style="list-style-type: none"> <li>❖ Plantear situaciones reales que surjan en el área para contemplar acciones preventivas.</li> </ul>	
Febrero	<ul style="list-style-type: none"> <li>❖ Seleccionar las diferentes tecnologías que más se adecuen a los requerimientos, y metas esperadas.</li> <li>❖ Búsqueda, selección y contratación del personal para incorporar al área de Nuevos Proyectos T.I.</li> <li>❖ Realizar talleres y cursos de capacitación al nuevo personal en las tecnologías seleccionadas para integrarlos al nuevo proyecto.</li> </ul>	3
Marzo	<ul style="list-style-type: none"> <li>❖ Delimitar normas, trazar una estrategia y establecer valores y hábitos que guíen al equipo de trabajo hasta su objetivo.</li> <li>❖ Establecer canales de comunicación dentro del área para que no haya malos entendidos que conduzcan a errores indeseados.</li> </ul>	2
Abril	<ul style="list-style-type: none"> <li>❖ Revisar informes sobre los talleres, y cursos de capacitación al nuevo personal.</li> <li>❖ Desarrollar un plan de gestión de calidad. Plan de mejora y criterio de medición del proyecto.</li> </ul>	2
Mayo	<ul style="list-style-type: none"> <li>❖ Elaborar el plan de gestión del tiempo. Estructurar un cronograma.</li> <li>❖ Realizar dinámicas en grupo que permita potenciar diferentes cualidades: como el liderazgo, el carácter resolutivo, la empatía, la prudencia, flexibilidad, la mentalidad analítica, la disciplina o la responsabilidad corporativa. Cada persona cuenta aptitudes específicas que deben ser desarrolladas por sus compañeros gracias al trabajo en equipo.</li> </ul>	2
Junio	<ul style="list-style-type: none"> <li>❖ Analizar y hacer hincapié en los contratiempos que surjan en el proyecto para fortalecer el trabajo en equipo y la colaboración entre ellos.</li> </ul>	1

Julio	<ul style="list-style-type: none"> <li>❖ Realizar reunión con el equipo de trabajo sobre la solución de problemas, rápidos y efectivos. Los imprevistos ocurren, y es mucho más probable que un grupo de personas logre afrontarlo de manera más exitosa de lo que podría serlo un individuo por su propia cuenta.</li> <li>❖ Actividad recreativa para potenciar al equipo, su productividad y su grado de iniciativa y participación en la toma de decisiones.</li> </ul>	2
Agosto	<ul style="list-style-type: none"> <li>❖ Realizar matriz FODA.</li> <li>❖ Realizar matriz de las 4 acciones (eliminar, incrementar, reducir y crear).</li> </ul>	2
Septiembre	<ul style="list-style-type: none"> <li>❖ Analizar y priorizar nuevas estrategias que surjan de las matrices anteriormente mencionadas.</li> <li>❖ Realizar manuales con respecto a las funciones de los integrantes del área.</li> </ul>	2
Octubre	<ul style="list-style-type: none"> <li>❖ Evaluar la evolución y desempeño del equipo dentro del área, teniendo en cuenta la coordinación de sus esfuerzos.</li> <li>❖ Mejorar los procesos en base a recomendaciones de las buenas prácticas en el área de Nuevos Proyectos T.I.</li> </ul>	2
Noviembre	<ul style="list-style-type: none"> <li>❖ Pruebas de rendimiento sobre lo realizado en el nuevo proyecto.</li> <li>❖ Análisis de las pruebas de rendimiento.</li> </ul>	2
Diciembre	<ul style="list-style-type: none"> <li>❖ Recopilar las lecciones aprendidas durante todo el ciclo de vida del nuevo proyecto.</li> <li>❖ Cierre anual con cena, y brindis para todo el personal del área.</li> </ul>	2

*Tabla 4. 2 Actividades del segundo año.*

# Anexo N°1

## Diagrama de Gantt

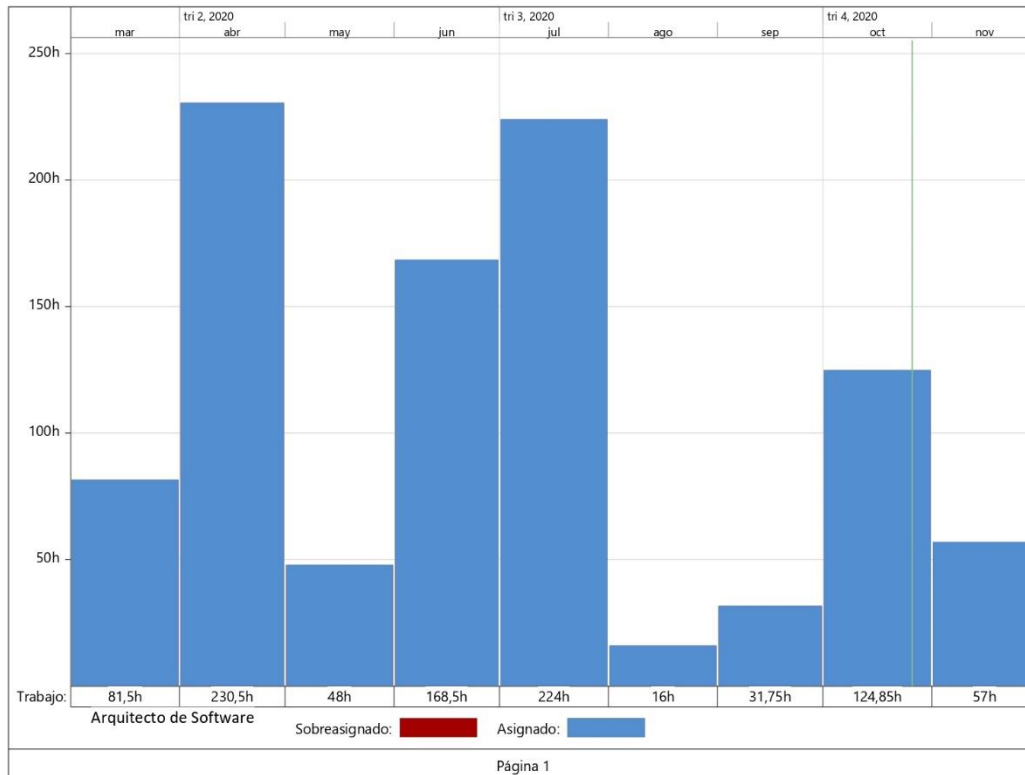




# Anexo N°2

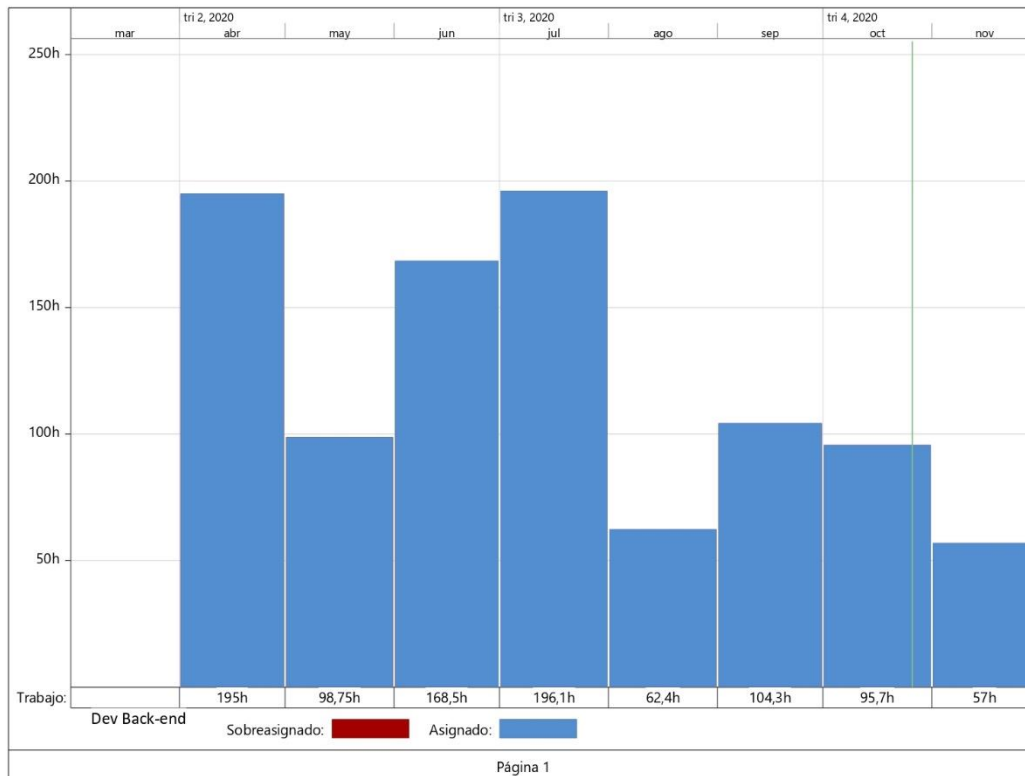
## Diagrama de Recursos

# 1. Diagrama de Recursos de Arquitecto de Software



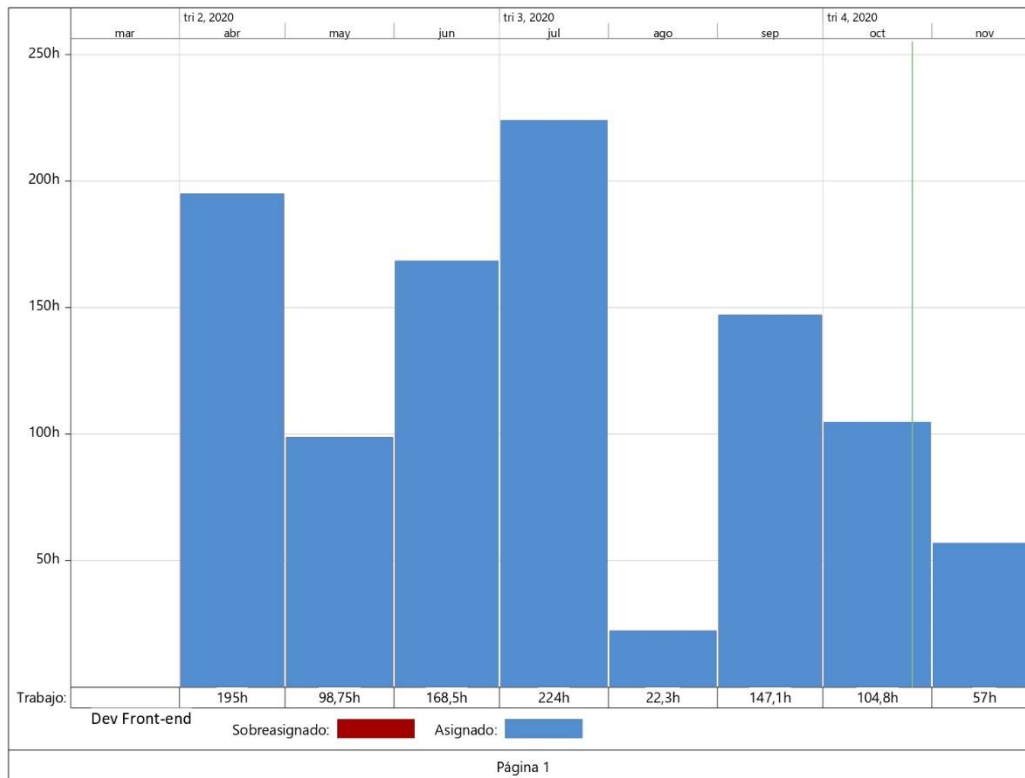
Anexo N° 2 - Figura 1. 1 Diagrama de Recursos de Arquitecto de Software.

## 2. Diagrama de Recursos de Dev Back-end



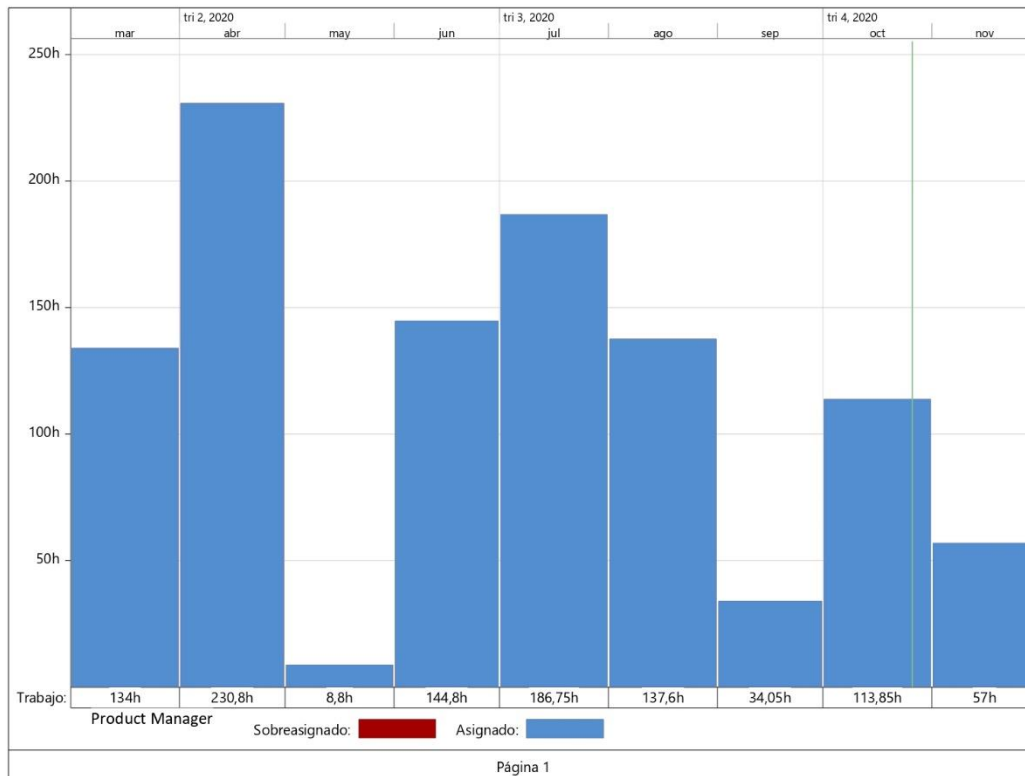
Anexo N° 2 - Figura 1. 2 Diagrama de Recursos de Dev Back-end

### 3. Diagrama de Recursos de Dev Front-end



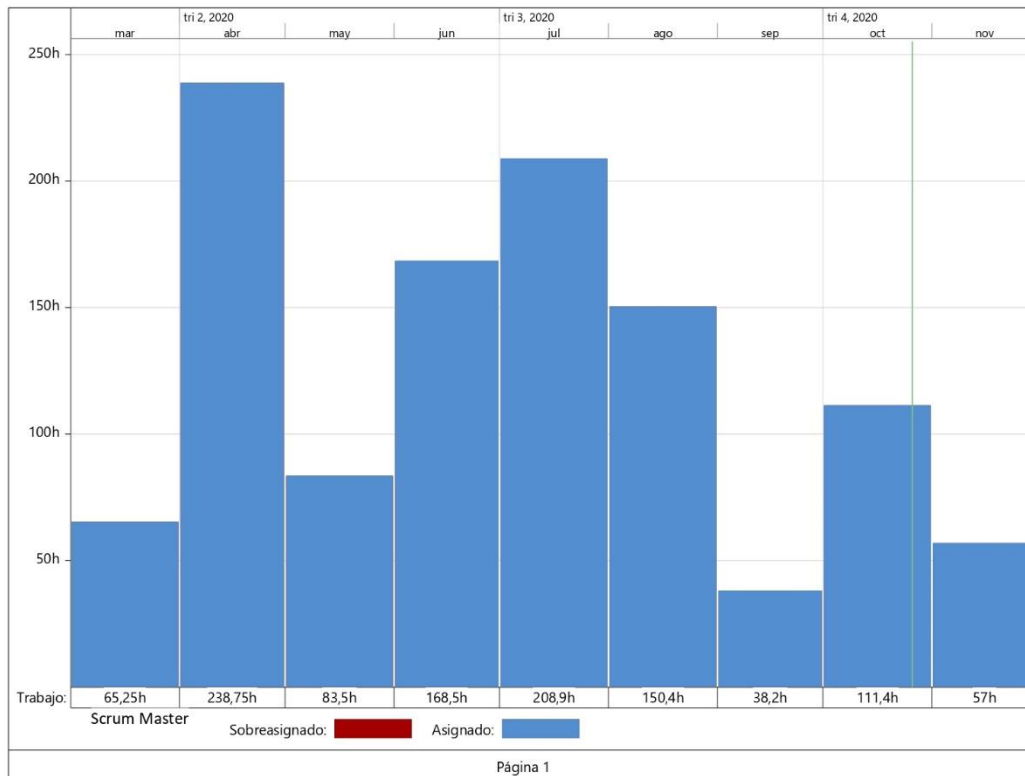
Anexo N° 2 - Figura 1. 3 Diagrama de Recursos de Dev Front-end.

## 4. Diagrama de Recursos de Product Manager



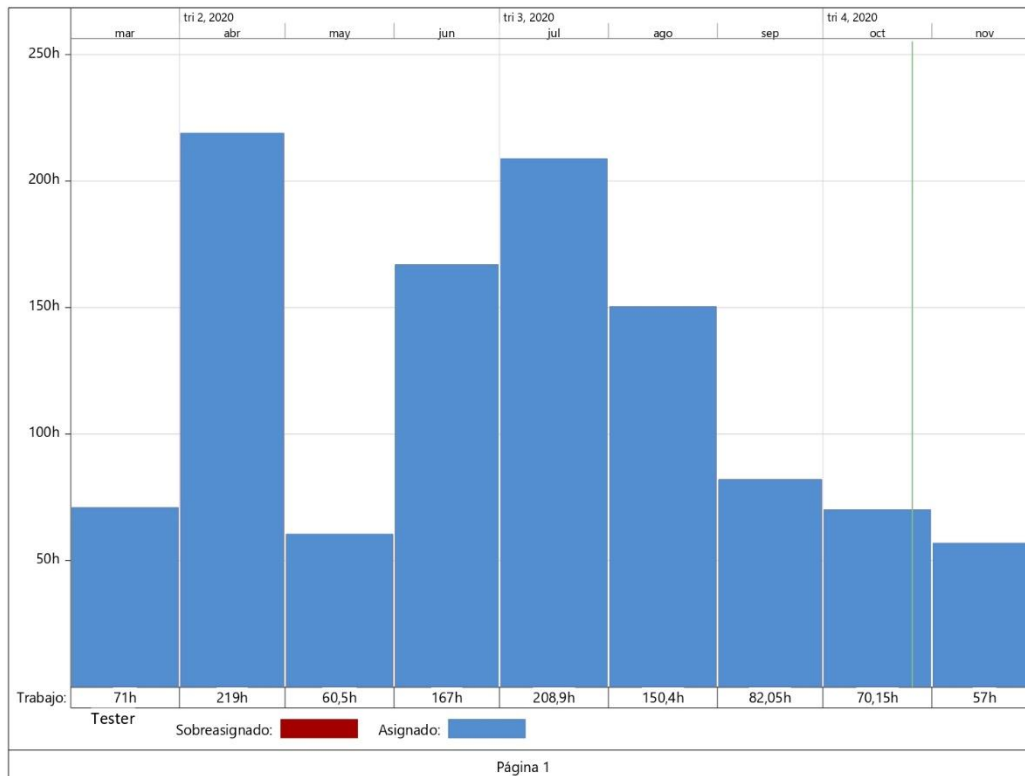
Anexo N° 2 - Figura 1. 4 Diagrama de Recursos de Product Manager.

## 5. Diagrama de Recursos de Scrum Master



Anexo N° 2 - Figura 1. 5 Diagrama de Recursos de Scrum Master.

## 6. Diagrama de Recursos de Tester

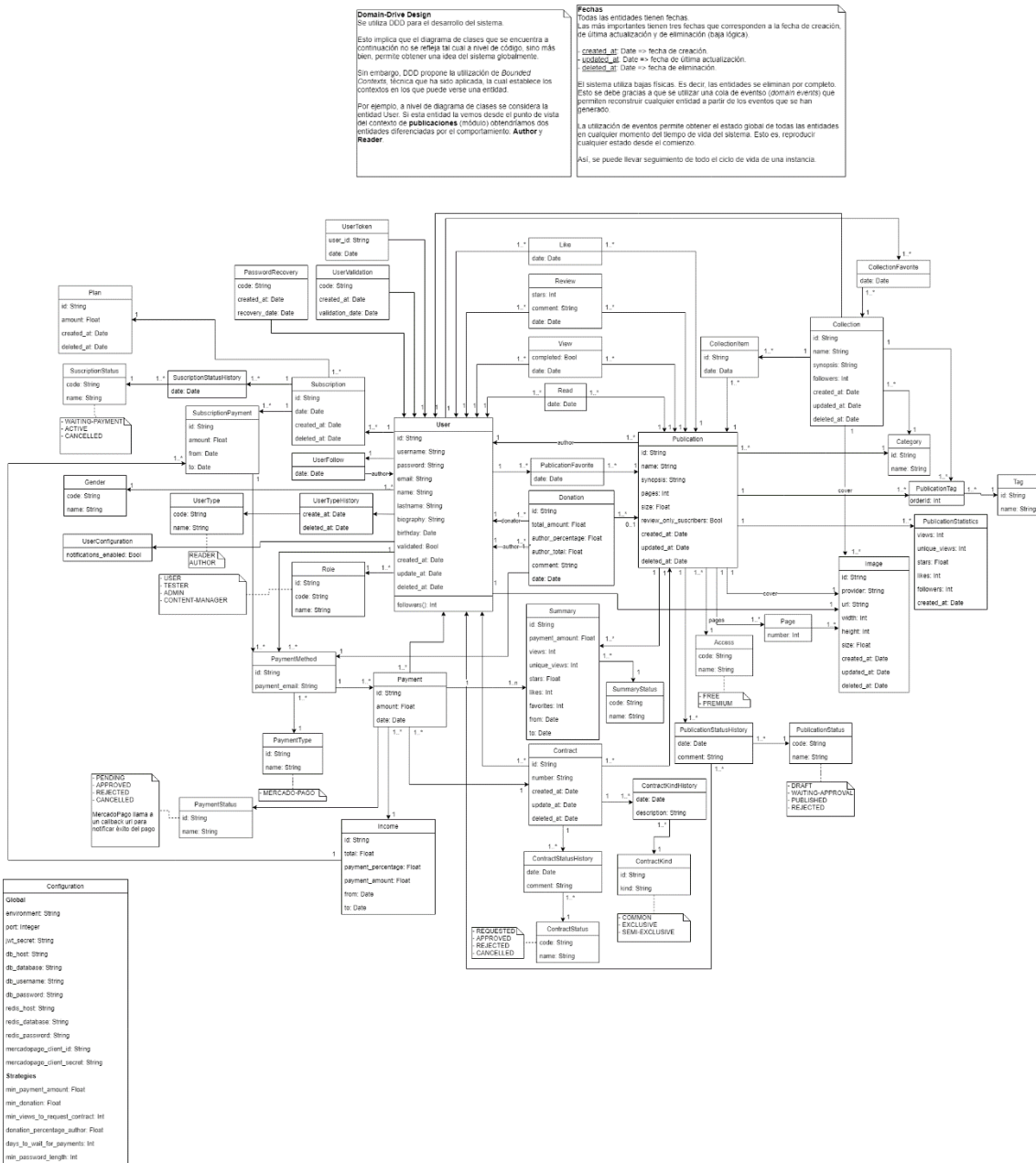


Anexo N° 2 - Figura 1. 6 Diagrama de Recursos de Tester.

# Anexo N°3

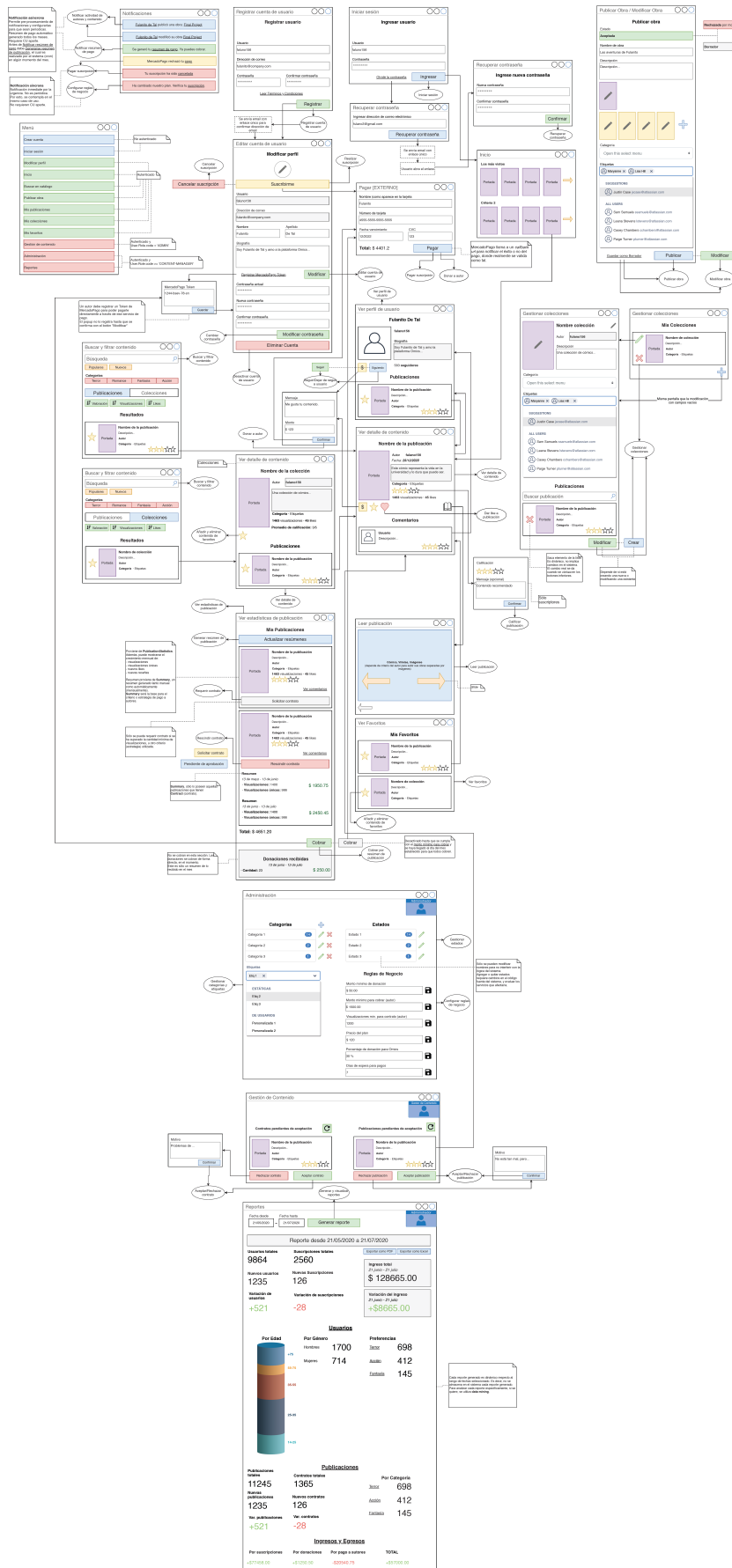
## Diagrama de Clases





# Anexo N°4

## Mockups



# Anexo N°5

## Manual de Usuarios

## 1. Introducción

Esta guía de usuario contiene documentación escrita e imágenes que ayudan a la comprensión y orden de las actividades que se pueden realizar en el sistema. Se detalla todo el manejo necesario para el uso apropiado del sistema, en el que se tienen en cuenta los conceptos básicos y una pauta informativa para aprender a usar el mismo de manera fácil y rápida. A medida que uno lo va leyendo, se encuentra bien detallado el uso del sistema Omics con sus objetivos y funcionalidades. Lo importante es poder brindarles todas las herramientas necesarias para utilizar la plataforma, aprovechando al máximo las funcionalidades que se ofrecen.

### 1.1. ¿Qué es Omics?

*Omics* es una plataforma para distribuir contenido de alta calidad, acompañando a los autores que recién comienzan con sus primeros cómics y otros que ya poseen trayectoria, pero que no encuentran un lugar seguro para publicar sus obras.

Es el "Spotify" de los cómics donde pueden darse a conocer artistas que no tienen el alcance de grandes editoriales como DC, Marvel o Dark Horse. Enfatiza su cercanía al usuario, el cual no tiene que invertir en papel, y puede seguir sus gustos según autor, categoría, estilo, etcétera.

El sistema se compone de 3 tipos de usuarios: lectores, autores y administradores. Un lector es el usuario final de la aplicación, el autor es un usuario que ha publicado alguna obra en la plataforma, y el administrador es designado por los administradores de *Omics*, para realizar la selección de obras cargadas por los autores.

### 1.2. Objetivos

Los objetivos de la plataforma son:

- ❖ Distribuir contenido (obras, cómics, historietas) ofreciéndose como catálogo a usuarios lectores. Permitirles filtrar el contenido bajo múltiples criterios.
- ❖ Brindar la posibilidad de reseñar, suscribirse y agregar a favoritos los cómics.
- ❖ Por otro lado, los autores deben poder publicar sus obras, con una previa aprobación por parte del administrador.
- ❖ Los lectores pueden donar a los autores creadores de contenido.

## 2. Instalación del Sistema

A continuación, se explica la instalación del Back-end y Front-end:

### 2.1. Instalación del Back-end

El back-end de *Omics* se ejecuta sobre un VPS, servidor privado virtual.

Los servicios de los que depende el backend se ejecutan sobre contenedores de Docker, por lo tanto, los servicios de base de datos, caché, archivos, etc., se ejecutan mediante Docker y están orquestados mediante docker-compose.

Por esto, para levantar los servicios sólo se requiere clonar el repositorio del proyecto, acceder a la rama master y ejecutar sobre línea de comandos (terminal):

```
docker-compose up -d
```

Con lo anterior, todos los servicios estarán corriendo en contenedores, con la configuración que se encuentran en "*docker-compose.yml*".

El backend, desarrollado en Rust, utiliza **cargo** como manejador de paquetes y gestor de configuración. Levantar el backend sólo requiere compilarlo y ejecutarlo como un binario más, gracias a las bondades de Rust:

```
cargo build --release --target/release/omics
```

El proyecto se compila a un binario y luego se ejecuta. Esta compilación puede realizarse en un servicio de Integración Continua y realizar deploy únicamente de este binario al servidor, por lo que sólo el último comando sería requerido.

A su vez, todo lo anterior ha sido resumido en un archivo encargado de ejecutar las distintas tareas requeridas para levantar, operar, depurar y realizar testing del proyecto. Gracias a make (binario de UNIX encargado de automatizar comandos).

Por lo tanto, levantar los servicios en Docker, limpiar la base de datos (en caso de ser necesario), ejecutar las migraciones, correr los tests unitarios y de integración, y, por último, ejecutar el backend, puede resumirse en una serie de comandos:

```
make run-services
```

```
make clean-db
```

```
make migrate
```

```
make build
```

```
make run
```

O todo junto:

```
make run-services clean-db migrate build run
```

En caso de que no se quiera limpiar la DB, sino únicamente ejecutar las migraciones para actualizar los esquemas de la base de datos, se omitiría "clean-db".

Cabe aclarar que Rust, gracias a cargo, su manejador de paquetes oficial, instala automáticamente todas las dependencias declaradas en el archivo especial llamado "Cargo.toml".

## 2.2. Instalación del Front-end

- 1) Instalar node (versión 10 o superior) en [WWW.NODEJS.ORG](http://www.nodejs.org).
- 2) Instalar angular CLI mediante un terminal: "`npm install -g @angular/cli`".
- 3) Dentro de la carpeta del proyecto, ir a la carpeta "web" y abrir una terminal ahí para ejecutar: "`npm install`".
- 4) Desde la carpeta web, ejecutar: "`ng serve`".
- 5) Ir a "[HTTP://LOCALHOST:4200/](http://localhost:4200/)" para visualizar la plataforma.

## 3. Funciones del Administrador

El manual está dirigido a los usuarios con el rol de Administrador del sistema que son los encargados de realizar las diferentes funciones necesarias para el correcto funcionamiento de la plataforma.

**En el Sistema existen 3 tipos de Administrador y se encargan de las siguientes funciones:**

- ❖ **Administrador General.**
  - Configuraciones iniciales.
  - Gestionar las Reglas de Negocio.
  - Gestión de Categorías de Publicaciones.
  - Generar Reportes.
- ❖ **Administrador Content Manager (Gestor de Contenidos).**
  - Aceptar Contrato.
  - Rechazar Contrato.
  - Aceptar Publicación.
  - Rechazar Publicación.
- ❖ **Administrador de la Base de Datos.**
  - Asignación de roles y permisos de usuario.
  - Realizar el Backup del sistema.

### 3.1. Generales

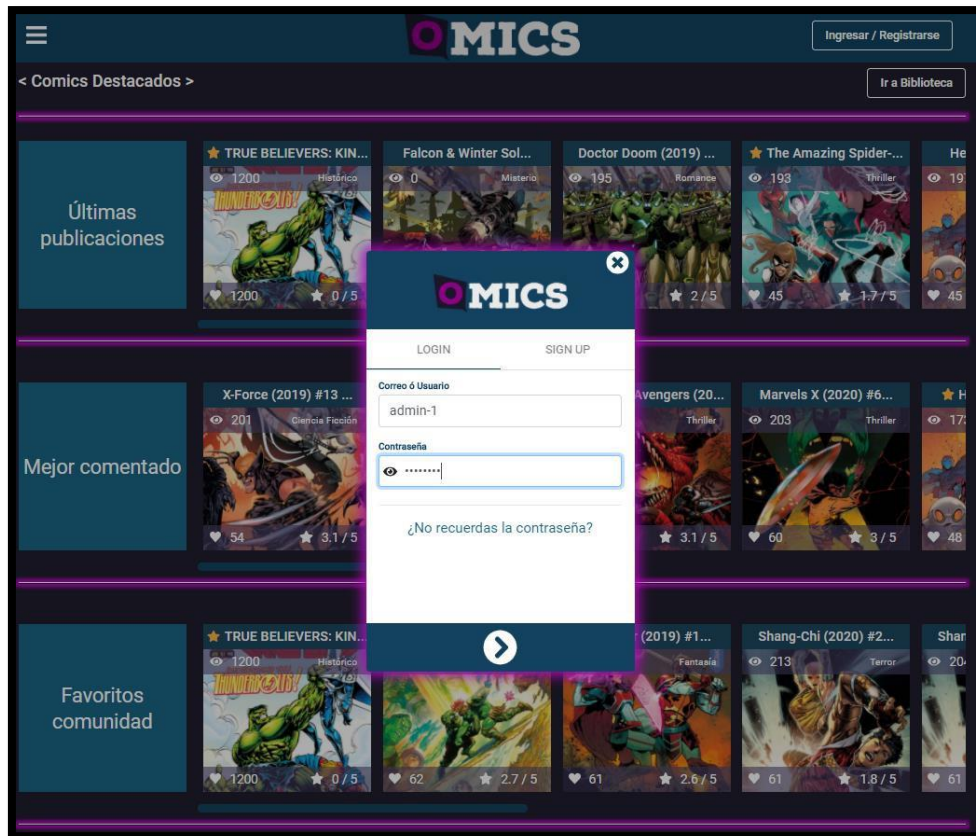
#### 3.1.1. Configuraciones iniciales

##### ❖ Pantalla Iniciar Sesión

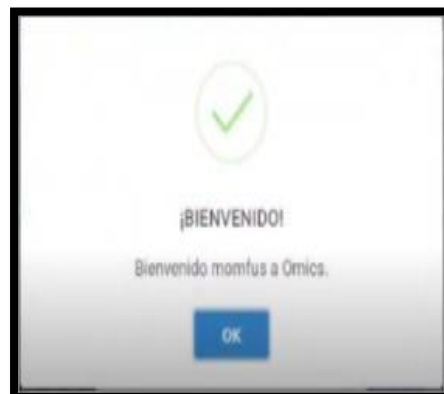
Para loguearse el Administrador debe ir a la esquina superior derecha y seleccionar el botón Ingresar/Registrarse. Al hacerlo se despliega la pantalla de LOGIN donde debe ingresar los siguientes datos:

- Correo de usuario.
- Contraseña.





Anexo N°5 - Figura 1. 1 Pantalla de Login del Administrador.



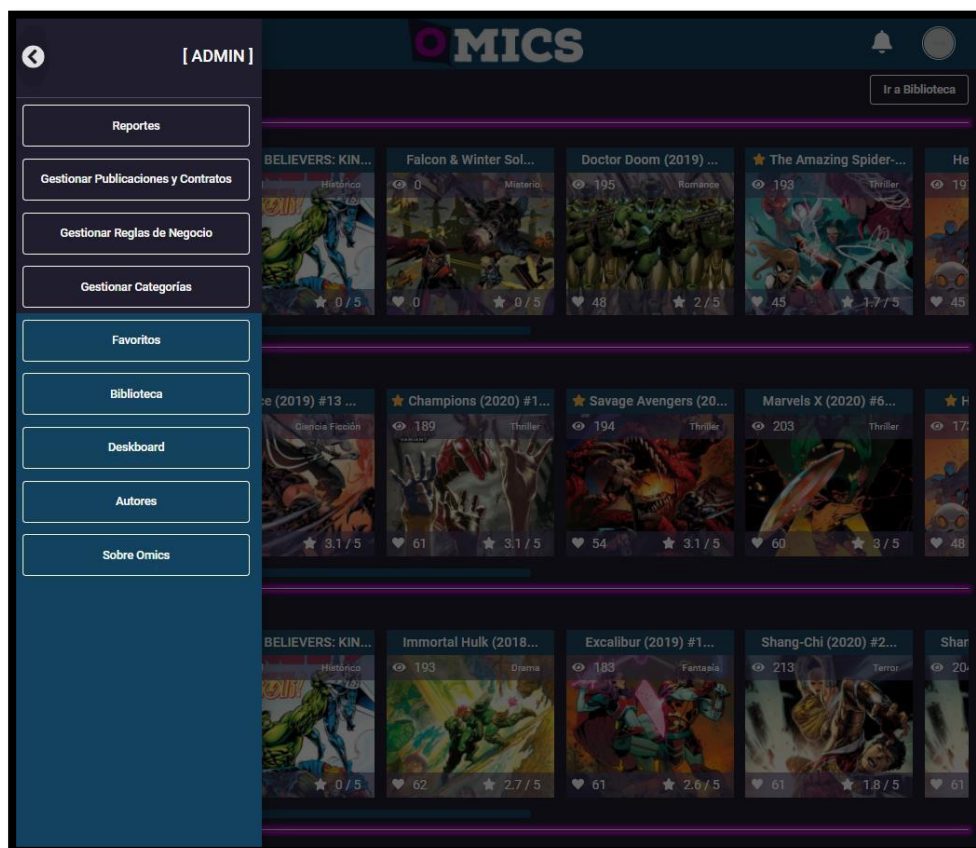
Anexo N°5 - Figura 1. 2 Pantalla de logueo exitoso.

❖ **Pantalla Deskboard del Administrador General**

Tras loguearse el Administrador General, debe ir a la esquina superior izquierda para acceder al **Menú de Despliegue**, el cual al desplegarse muestra las siguientes opciones:

- Reportes.
- Gestionar Publicaciones y Contratos.

- Gestionar Reglas de Negocio.
- Gestionar Categorías.
- Favoritos.
- Biblioteca.
- Deskboard.
- Autores.
- Sobre Omics.

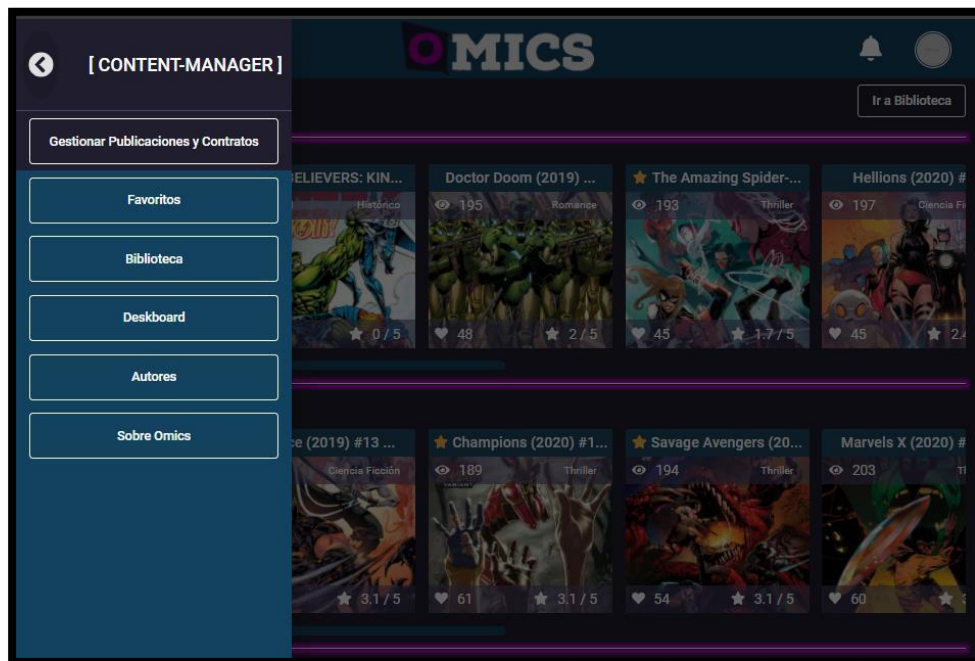


Anexo N°5 - Figura 1.3 Pantalla de opciones del Administrador general.

### ❖ Pantalla Deskboard del Content Manager

Una vez logueado el Content Manager tiene acceso a su Deskboard con las siguientes opciones:

- Gestionar Publicaciones y Contratos.
- Favoritos.
- Biblioteca.
- Deskboard.
- Autores.
- Sobre Omics.



Anexo N°5 - Figura 1. 4 Pantalla de opciones del Content Manager.

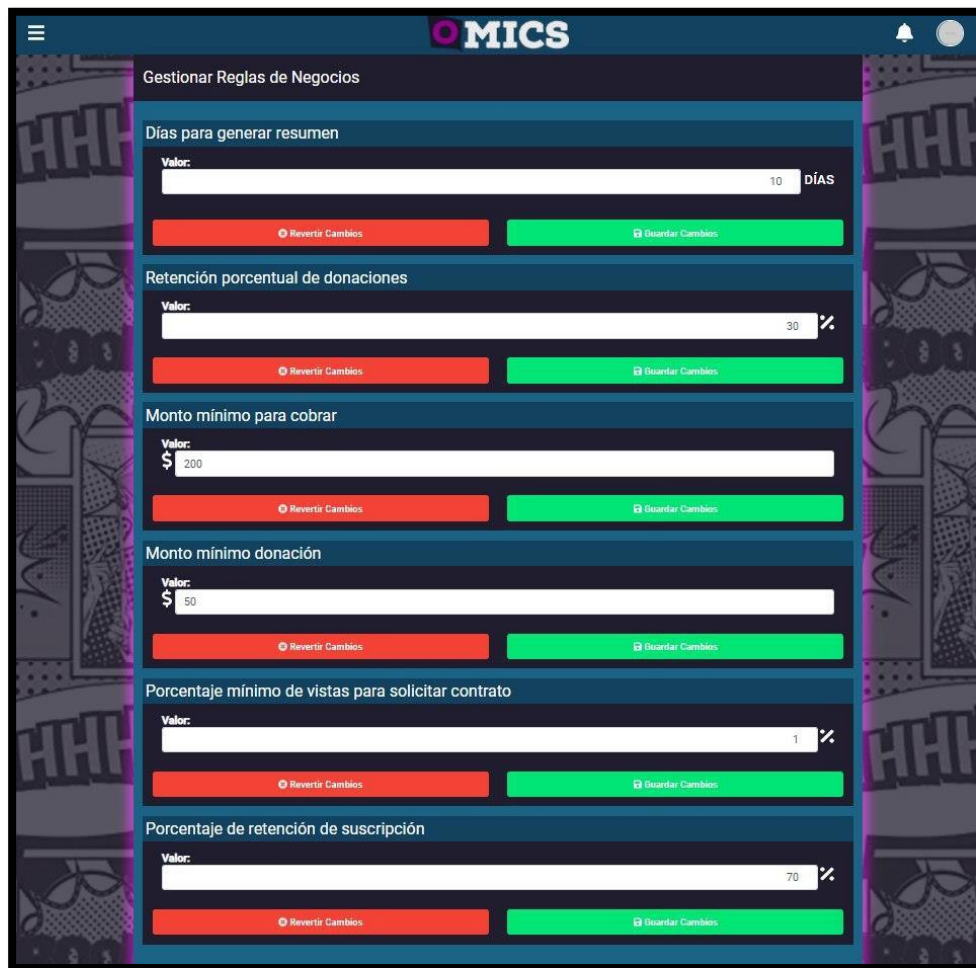
## 3.2. Administrador General

### 3.2.1. Gestionar las Reglas de Negocio

#### ❖ Pantalla Gestionar Reglas del Negocio

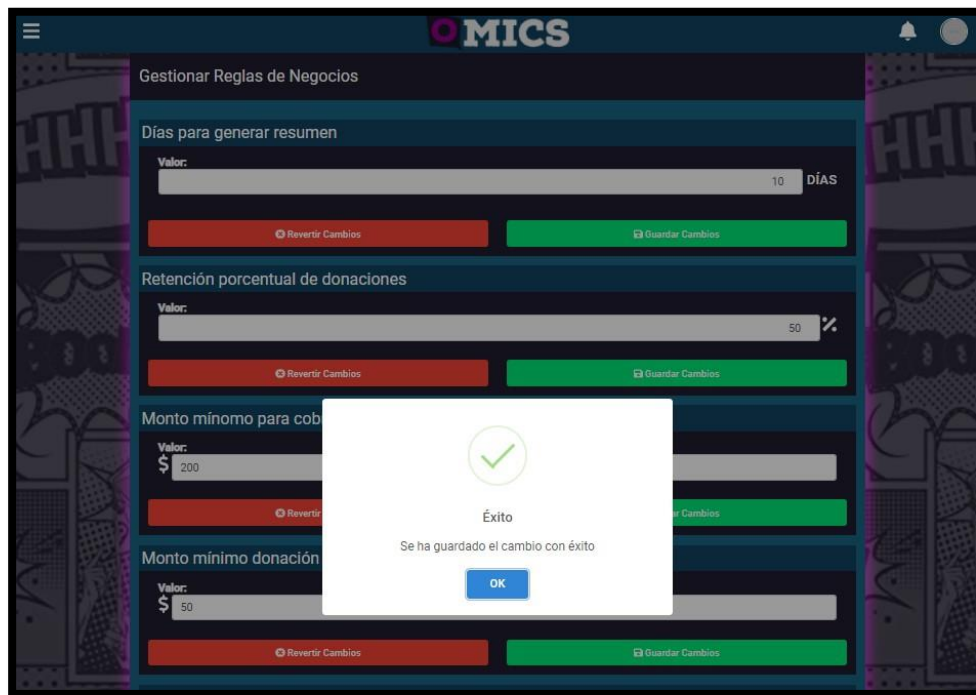
Para Gestionar las Reglas de Negocio, el Administrador debe estar logueado, y seleccionar del **Menú de despliegue** la opción **Gestionar Reglas de Negocio**. Al hacerlo se le muestra la pantalla de Gestionar Reglas de Negocios, en la cual puede modificar la siguiente información:

- Días para generar resumen.
- Retención porcentual de donaciones.
- Monto mínimo para cobrar.
- Monto mínimo donación.
- Porcentaje mínimo de vistas para solicitar contrato.
- Porcentaje de retención de suscripción.



Anexo N°5 - Figura 1. 5 Pantalla Gestionar Reglas de Negocio.

Al modificar una regla de negocio de forma correcta, se muestra un mensaje de éxito.

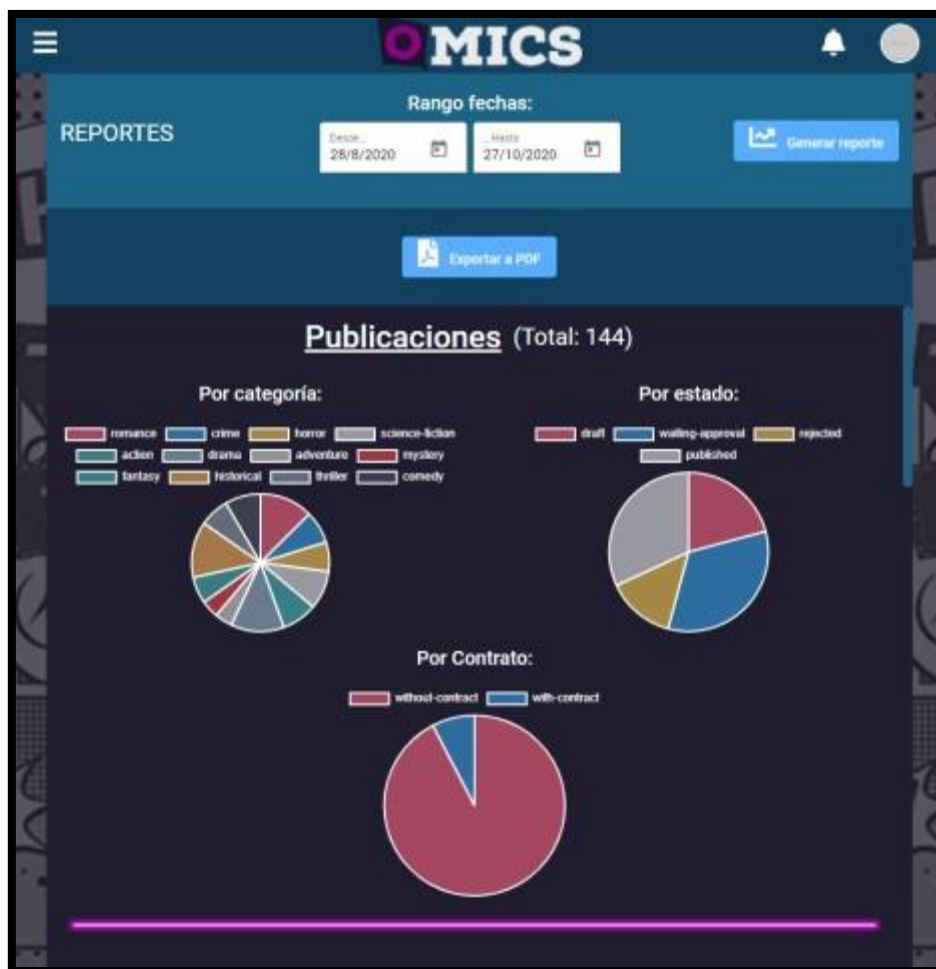


Anexo N°5 - Figura 1. 6 Pantalla regla de negocio modificada, mensaje de éxito.

### 3.2.2. Generar Reportes

#### ❖ Pantalla Generar Reportes

Para Generar los Reportes, el Administrador debe estar logueado, seleccionar del **Menú de despliegue** la opción **Reportes**, escoger el **Rango de fechas** (Desde – Hasta) que desea generar el reporte. Luego seleccionar el botón **Generar Reporte** para visualizar los datos en el rango de fechas seleccionad (por defecto, las fechas visualizadas del reporte desde dos meses atrás hasta la fecha actual).



Anexo N°5 - Figura 1. 7 Pantalla de Reportes.

❖ **Generar PDF de Reportes**

Para exportar los datos en formato PDF, seleccionar el botón **Exportar a PDF**. Luego se abre una pestaña en el navegador con el PDF para descargar y los datos entre los rangos de fechas es seleccionado.

### 3.3. Administrador Content Manager

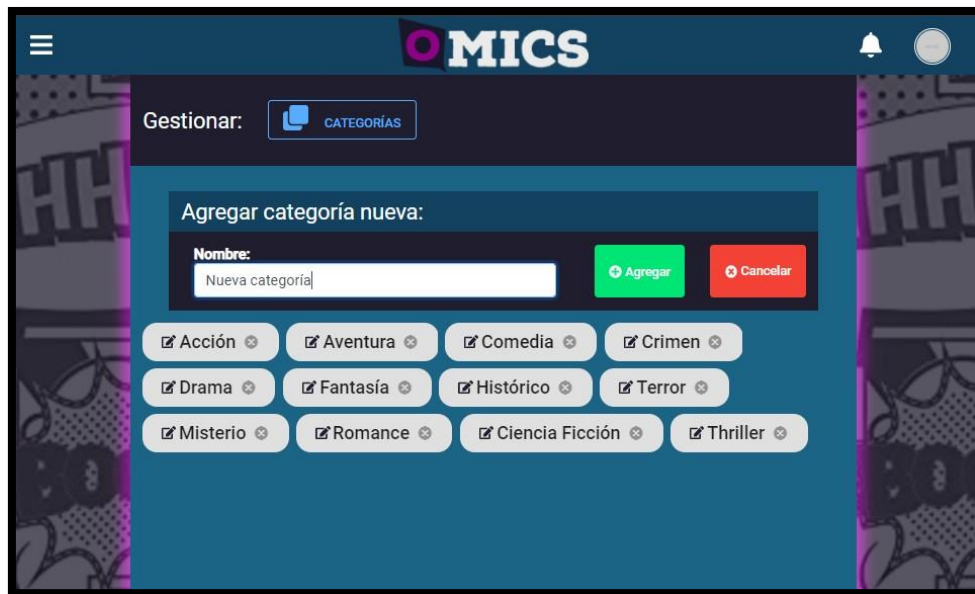
#### 3.3.1. Gestión de Categorías de Publicaciones

❖ **Pantalla Gestionar Categorías de Publicaciones**

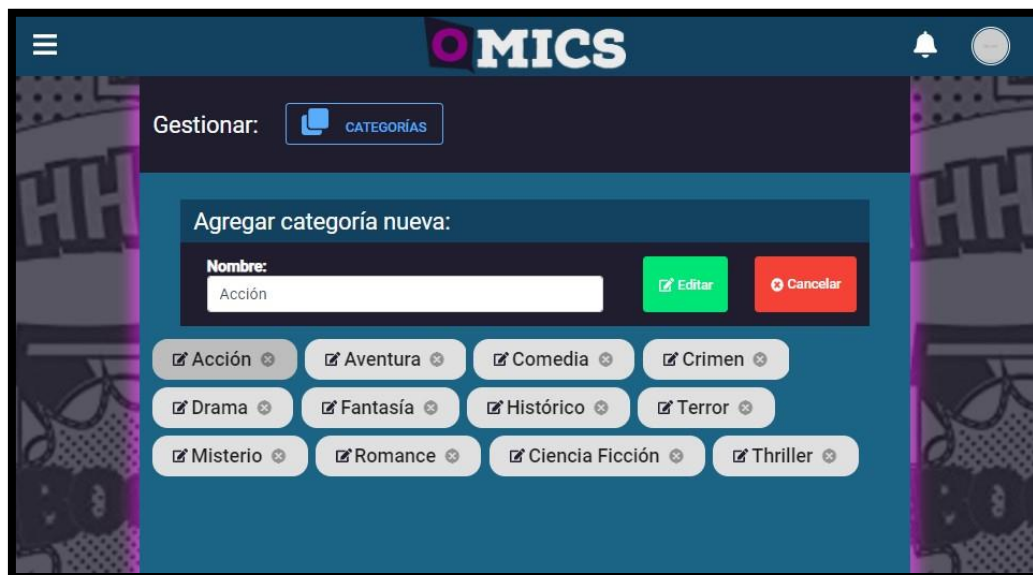
Para Gestionar las Categorías de las Publicaciones, el Administrador debe estar logueado, y seleccionar del **Menú de despliegue** la opción **Gestionar Categorías**. Al hacerlo se le muestra la pantalla de Gestionar Categorías de Publicaciones, en la cual se visualiza la siguiente información:

- Las categorías existentes.

- Botones: Agregar, Eliminar, Cancelar, Editar.

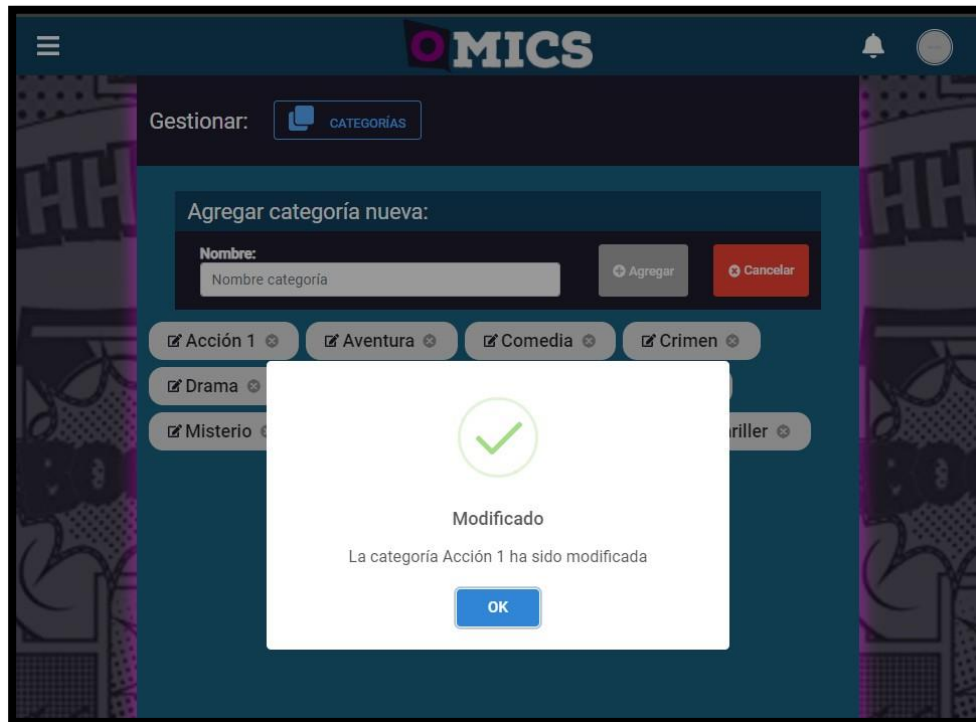


Anexo N°5 - Figura 1. 8 Pantalla Agregar una Categoría.



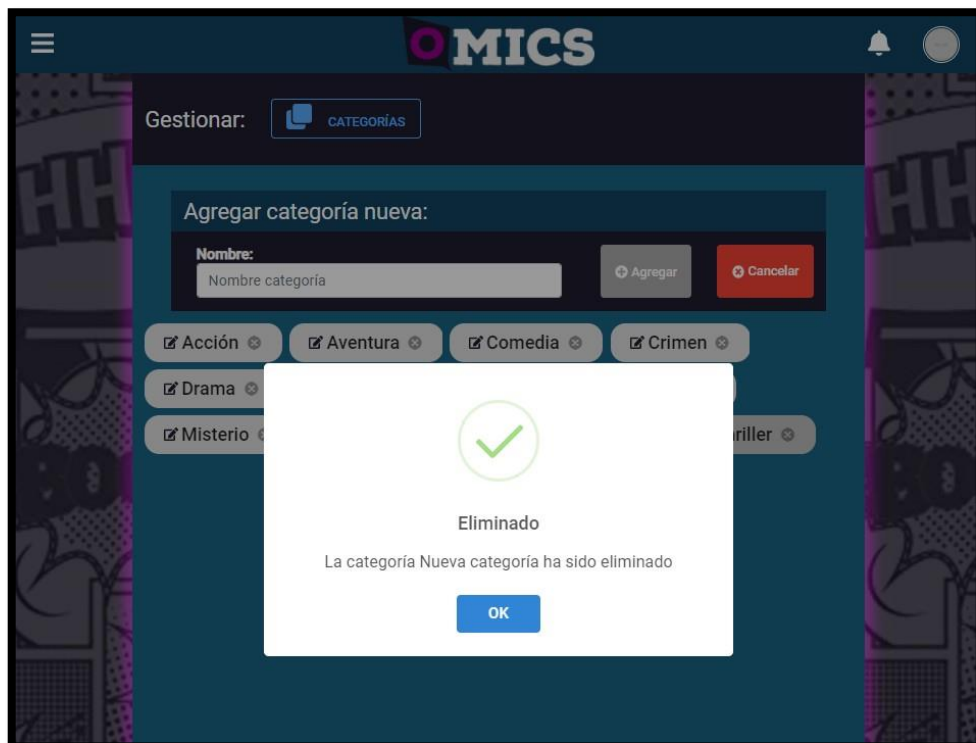
Anexo N°5 - Figura 1. 9 Pantalla para Editar una Categoría.

Al Editar una Categoría se muestra mensaje de éxito, indicando que se modificó la Categoría.



Anexo N°5 - Figura 1. 10 Pantalla de éxito, Categoría modificada.

Sólo se pueden eliminar las Categorías que no tengan Publicaciones asociadas. Al eliminar una Categoría se muestra mensaje de éxito, indicando que se eliminó la Categoría.





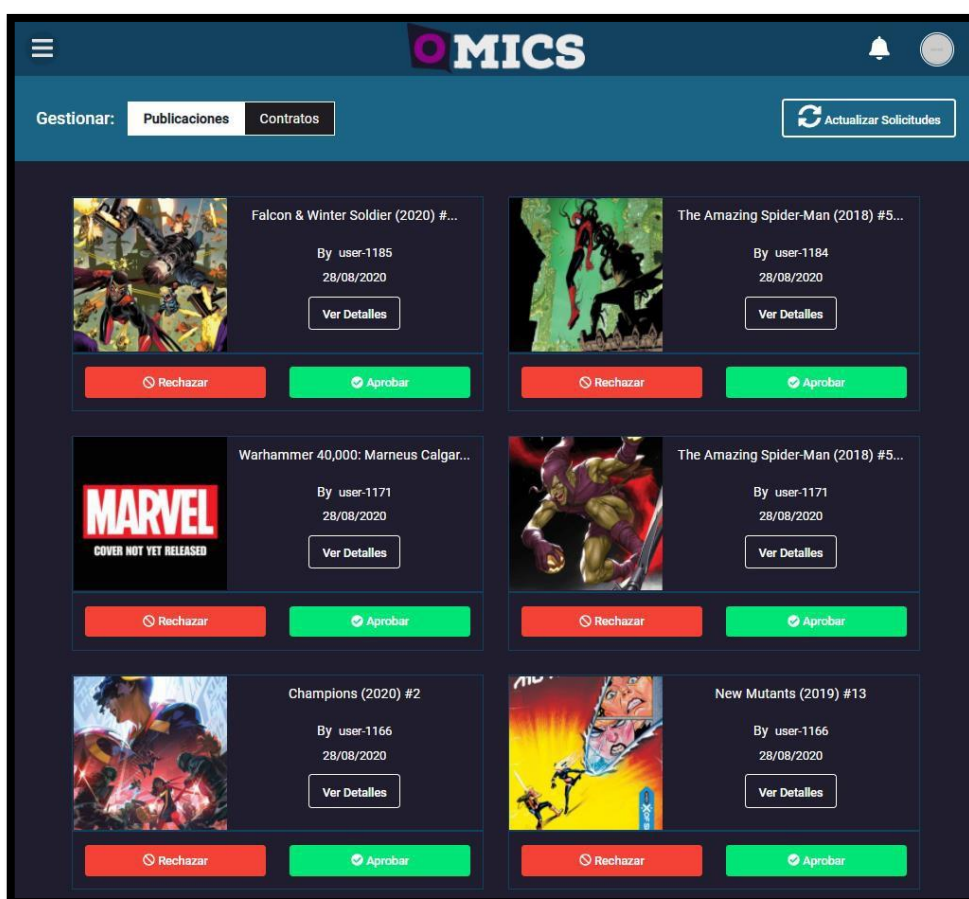
Anexo N°5 - Figura 1. 11 Pantalla de éxito, Categoría eliminada.

### 3.3.2. Gestión de Publicaciones

#### ❖ Pantalla Gestionar Contenido

Para Gestionar el Contenido que se publica en la plataforma, el Content Manager debe ir a la esquina superior izquierda y seleccionar el **Menú de Despliegue**. Al desplegarse elige la opción **Gestionar Publicaciones y Contratos**. Al hacerlo se muestra una pantalla con los botones de:

- Publicaciones.
- Contratos.
- Actualizar Solicitudes.



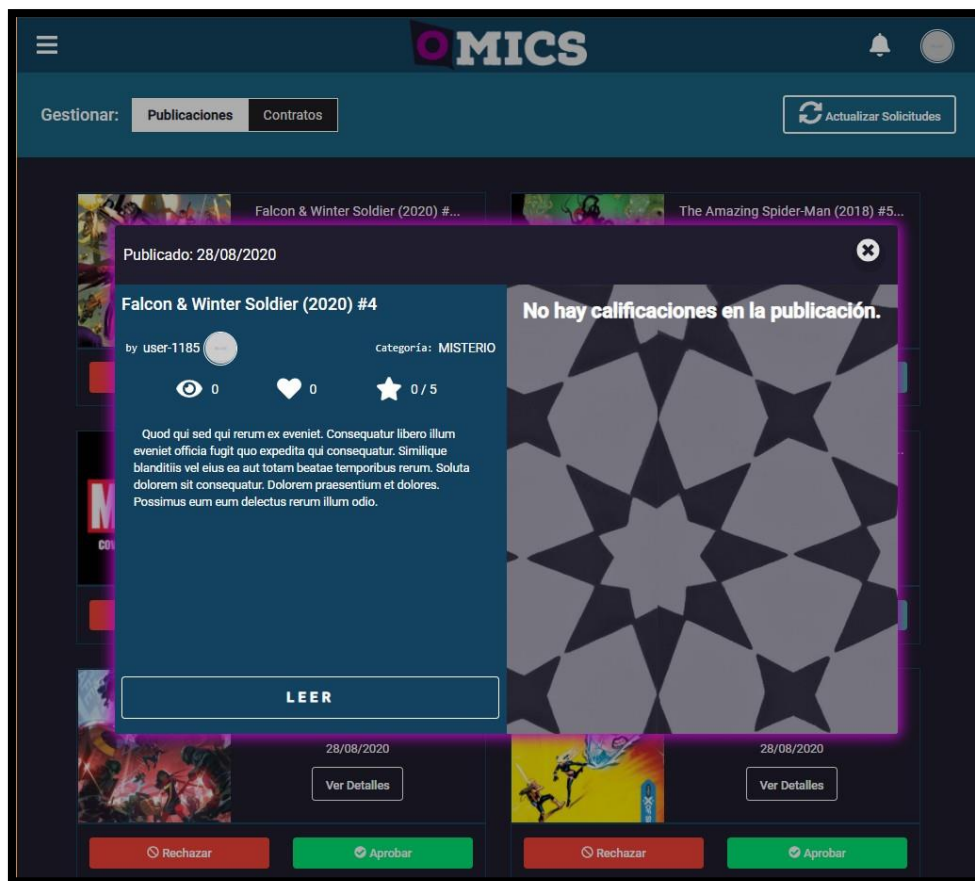
Anexo N°5 - Figura 1. 12 Pantalla Gestionar Publicaciones del Content Manager.

Si desea gestionar las Publicaciones de los Autores, debe seleccionar el botón **Publicaciones**. Al realizar dicha acción se despliega una pantalla con el listado de todas las Publicaciones que tienen estado "Pendiente de aprobación". Cada Publicación se muestra con los siguientes botones:

- Rechazar.
- Aceptar.
- Ver Detalles.

❖ **Pantalla Gestionar Publicaciones**

Si el Content Manager desea ver los detalles de la Publicación en estado "Pendiente de aprobación", debe seleccionar el botón **Ver Detalles** y se despliega una pantalla con los detalles de la Publicación.



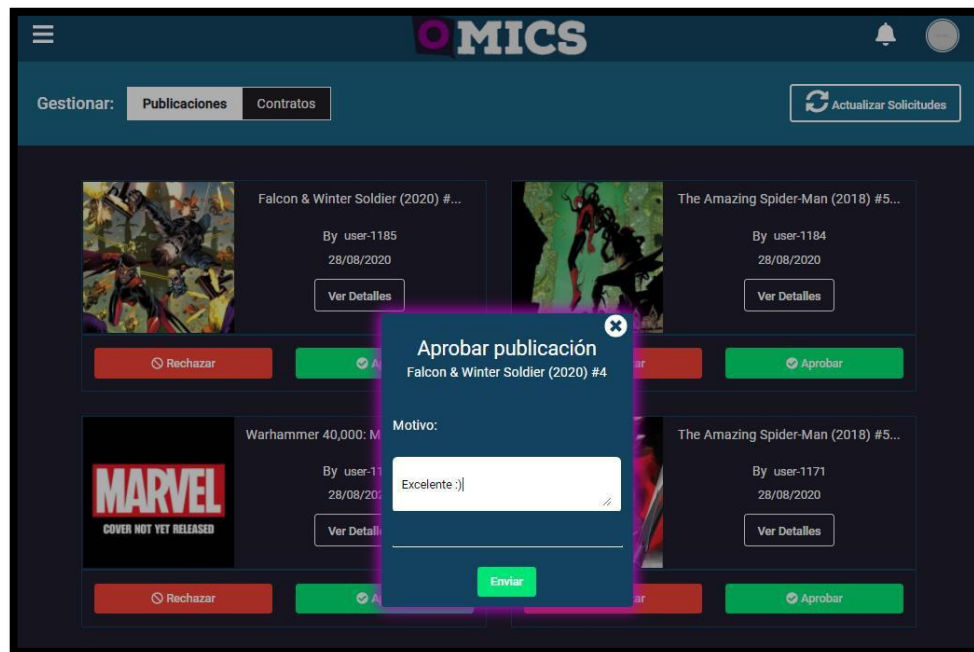
Anexo N°5 - Figura 1. 13 Pantalla Detalles de la Publicación.

### 3.3.2.1. Aceptar Publicación

❖ **Si el Content Manager decide Aprobar la Publicación:**

Una vez visto el contenido de la publicación decide Aprobar la misma para que sea publicada en la plataforma. Para esto debe seleccionar el botón **Aprobar** (botón verde). Luego, aparece una ventana donde se ingresa el "Motivo" de la aprobación.

Le llega al Autor una notificación y un E-mail para avisarle que su Publicación fue "Aprobada" con el "Motivo" adjunto.



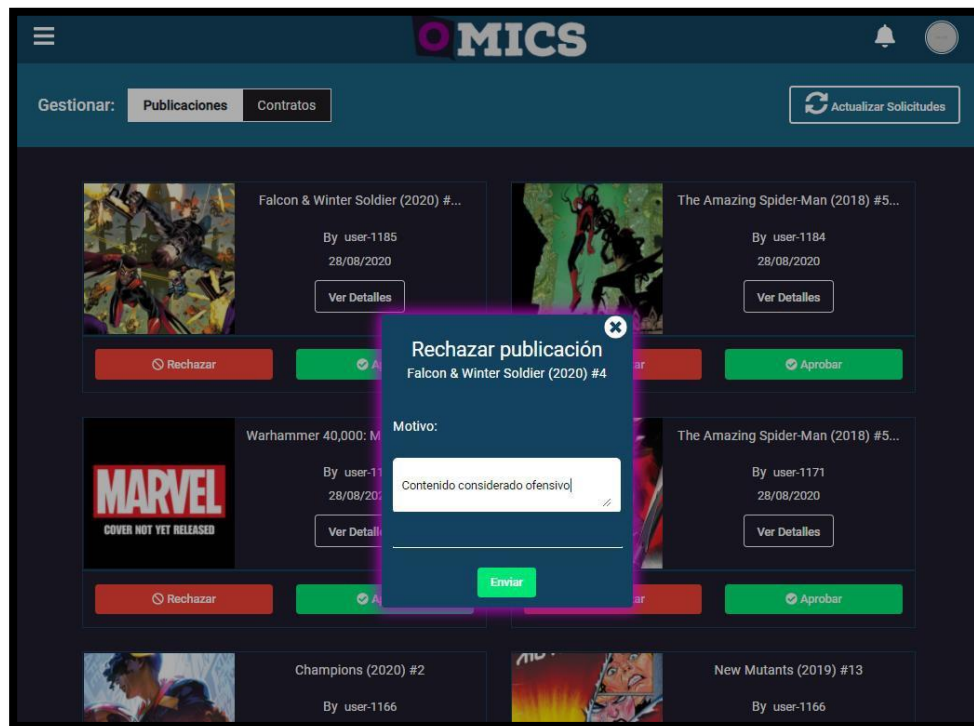
Anexo N°5 - Figura 1. 14 Pantalla Publicación Aprobada.

## Rechazar Publicación

### ❖ Si el Content Manager decide Rechazar la Publicación:

Una vez visto el contenido de la Publicación decide Rechazar la misma para que no sea publicada en la plataforma, por lo cual debe seleccionar el botón **Rechazar** (botón rojo). Luego, aparece una pantalla donde ingresa el "Motivo" del rechazo de la Publicación.

Le llega al Autor una notificación y un E-mail, para avisarle que su Publicación fue "Rechazada" con el "Motivo" adjunto.



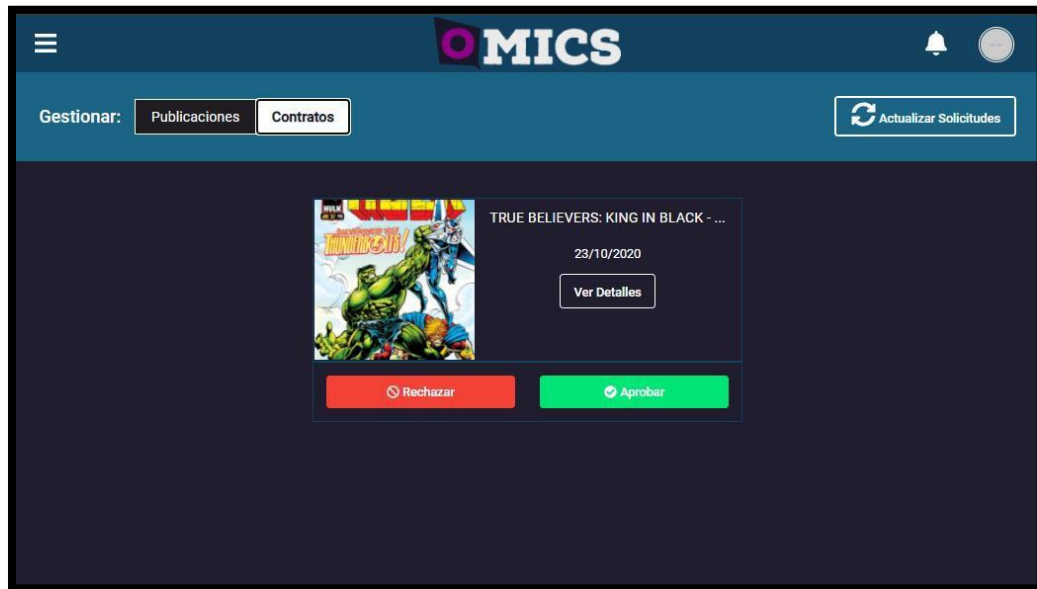
Anexo N°5 - Figura 1. 15 Pantalla Rechazar la Publicación.

### 3.3.3. Gestión de Contratos

#### ❖ Pantalla Gestionar Contratos

Si desea gestionar las Publicaciones de los Autores, debe dirigirse al menú desplegable marcando "Gestionar Publicaciones y Contratos". Luego, seleccionar el botón **Contratos**. Al realizar dicha acción se despliega una pantalla con el listado de todas las Publicaciones que tienen Contrato con estado "Pendiente de aprobación". Cada Publicación se muestra con dos botones, para aceptar o rechazar el contrato:

- Rechazar.
- Aceptar.



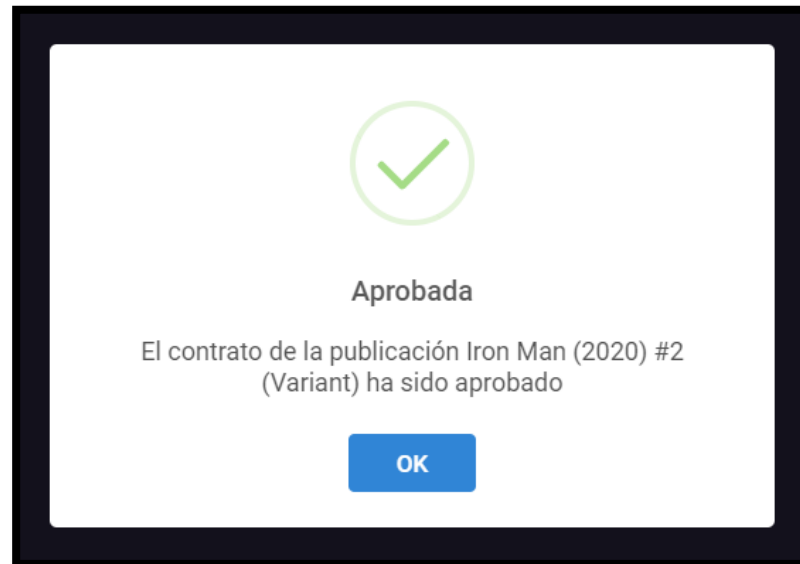
Anexo N°5 - Figura 1. 16 Pantalla Gestionar Contratos.

Cuando el Content Manager desea ver los detalles de la Publicación que tienen Contrato en estado "Pendiente de aprobación", debe seleccionar el botón **Ver Detalles** y se despliega una pantalla con los detalles de la Publicación.

### 3.3.3.1. Aceptar Contrato

#### ❖ Si el Content Manager decide Aprobar el Contrato:

Una vez visto el Detalle de la Publicación y la cantidad de visualizaciones que tiene, decide Aprobar el Contrato para que el Autor comience a cobrar por dicha Publicación. Para esto debe seleccionar el botón **Aprobar** (botón verde). Le llega al Autor una notificación y un E-mail, para avisarle que el Contrato fue Aprobado con el Motivo adjunto.

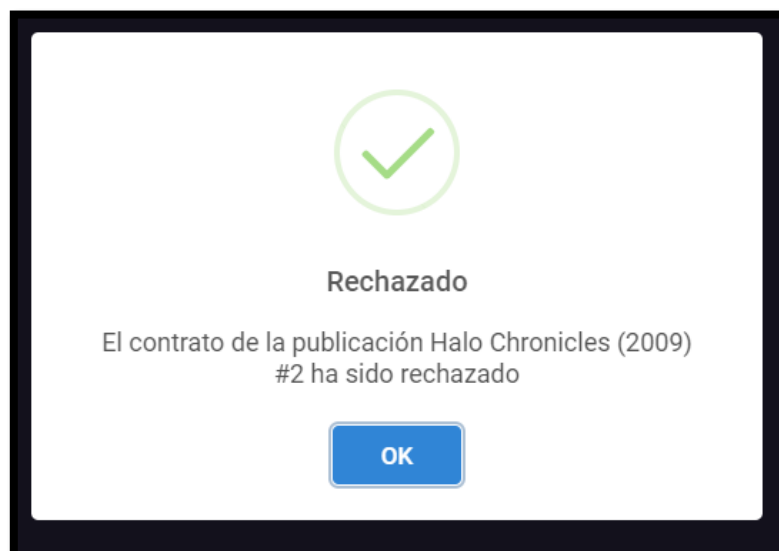


Anexo N°5 - Figura 1. 17 Pantalla mensaje Contrato Aprobado.

### 3.3.3.2. Rechazar Contrato

#### ❖ Si el Content Manager decide Rechazar el Contrato:

Una vez visto el Detalle de la Publicación y la cantidad de visualizaciones que tiene, decide Rechazar el Contrato, para lo cual debe seleccionar el botón **Rechazar** (botón rojo). Le llega al Autor una notificación y un E-mail, para avisarle que su Publicación fue Rechazada con el Motivo adjunto.



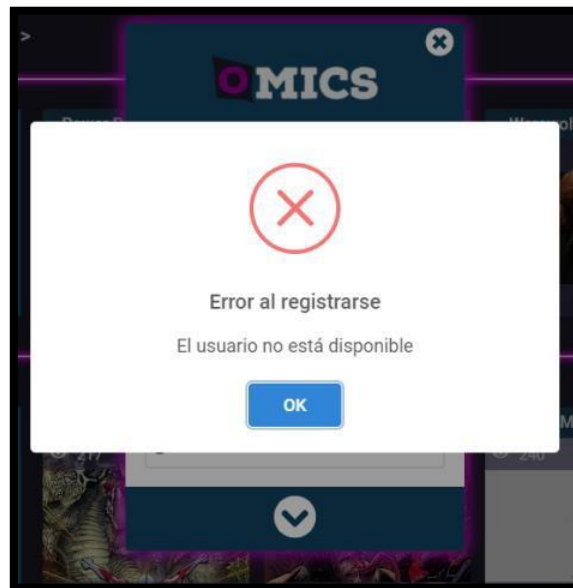
Anexo N°5 - Figura 1. 18 Pantalla mensaje Contrato Rechazado.

## 4. Mensajes de Error

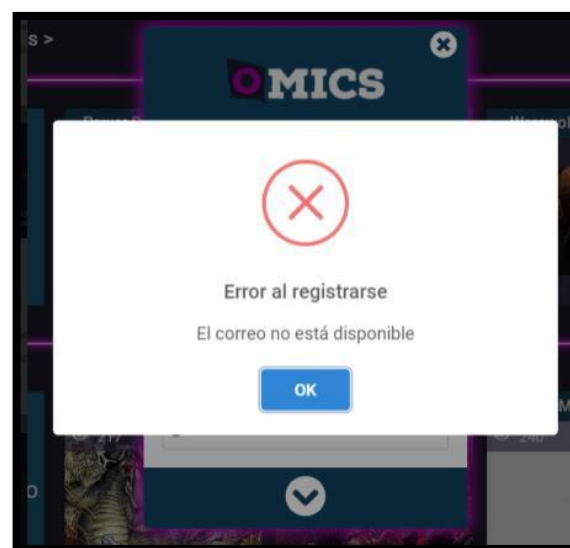
### 4.1. Errores de Pantalla de Registro

#### Errores:

- ❖ El usuario no está disponible porque ya existe.
- ❖ El correo no está disponible porque ya existe una cuenta con el mismo.



Anexo N°5 - Figura 1. 19 El usuario no está disponible

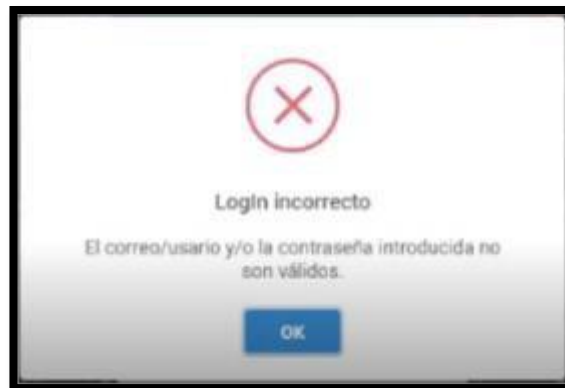


Anexo N°5 - Figura 1. 20 El correo no está disponible

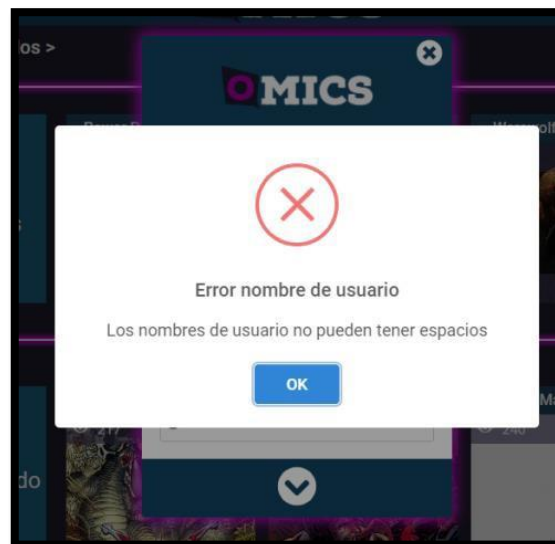
## 4.2. Errores de Pantalla de Login

### Errores:

- ❖ E-mail ingresado no coincide con una cuenta existente.
- ❖ El nombre de usuario tiene un formato incorrecto.
- ❖ El nombre de usuario no está disponible porque ya existe.
- ❖ Contraseña ingresada no coincide con una cuenta existente.



Anexo N°5 - Figura 1. 21 Pantalla de logueo erróneo.



Anexo N°5 - Figura 1. 22 Nombre de usuario erróneo.

## 4.3. Errores de Pantalla Recuperar Contraseña

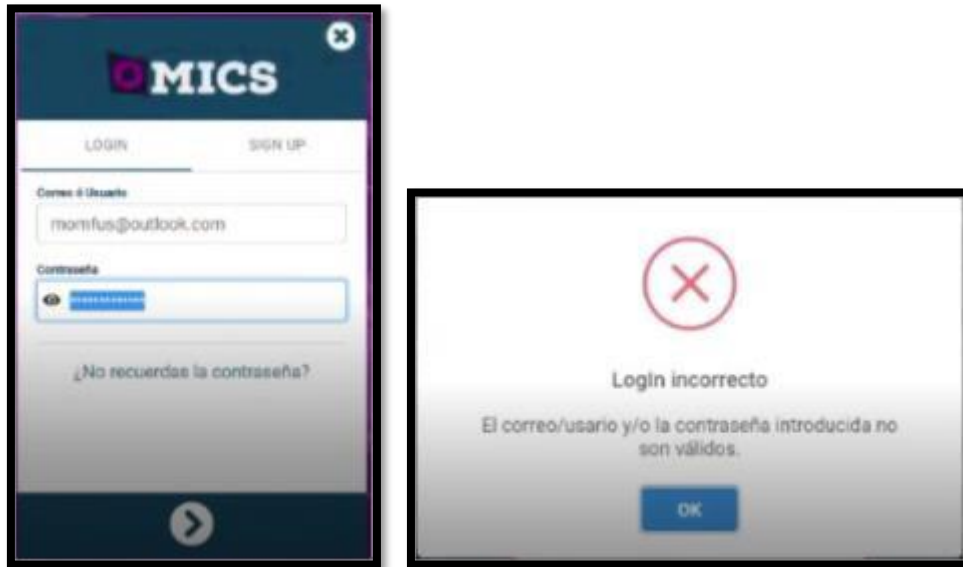
### Errores:

- ❖ La contraseña introducida es incorrecta.
- ❖ El correo ingresado no coincide con el que utilizó para registrarse en la plataforma por primera vez, al crear su cuenta de usuario.



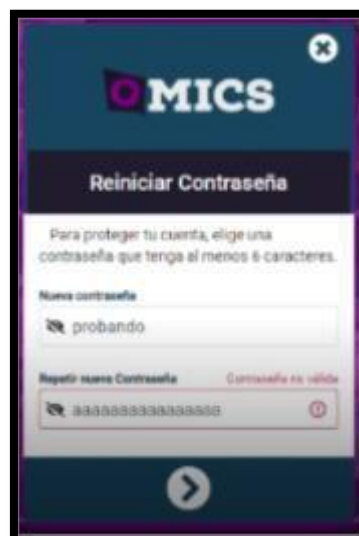
- ❖ La contraseña ingresada en el campo **Nueva contraseña**, y la ingresada en el campo **Repetir nueva contraseña** no coinciden.

El usuario selecciona la opción **¿No recuerdas la contraseña?**, se despliega pantalla para ingresar correo del usuario y te llega un email al mismo para recuperar la contraseña.



Anexo N°5 - Figura 1. 23 Pantalla ingreso de contraseña incorrecta.

El usuario ingresa la **Nueva Contraseña** y repite la contraseña ingresada en el campo **Repetir Nueva Contraseña**.



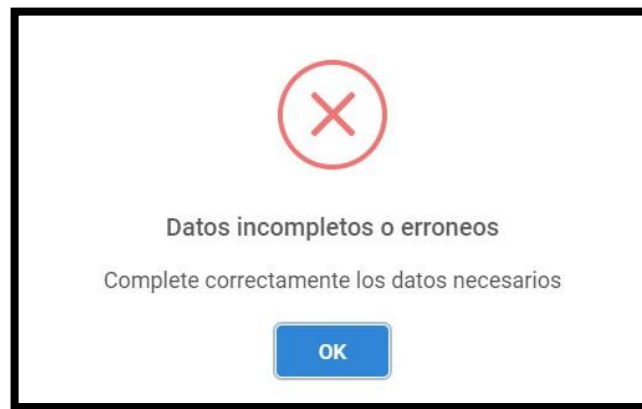
Anexo N°5 - Figura 1. 24 Las contraseñas ingresadas no coinciden.

#### 4.4. Errores de Pantalla Editar Perfil

##### Errores:

- ❖ E-mail ingresado no tiene un formato correcto.
- ❖ Formato inválido si el nombre de email tiene caracteres especiales.
- ❖ Formato inválido si la contraseña reescrita no es igual a la contraseña ingresada.
- ❖ No seleccionó botón "Guardar Cambios", al haber modificado algunos o todos los campos del Perfil de usuario.
- ❖ Ingresó contraseña incorrecta al querer "Cambiar Contraseña".

Si el usuario al modificar algunos de los campos no ingresó de forma correcta algunos de ellos, o no completó todos los campos y selecciona **Guardar cambios** (botón verde), se muestra un mensaje de error: "**Datos incompletos o incorrectos**". Para que el usuario vuelva a completar los campos debe seleccionar **Ok** (botón celeste) y volver a llenar los mismos.

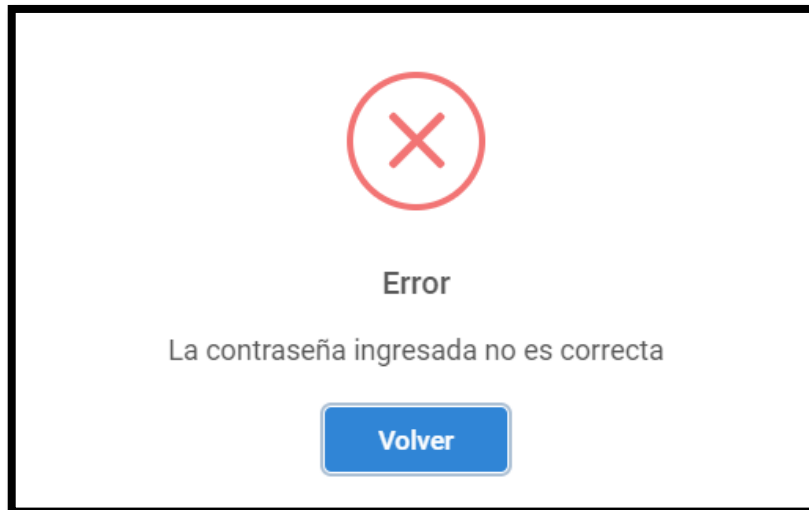


Anexo N°5 - Figura 1. 25 Pantalla mensaje de error al editar perfil.

## 4.5. Errores de Pantalla Eliminar Cuenta

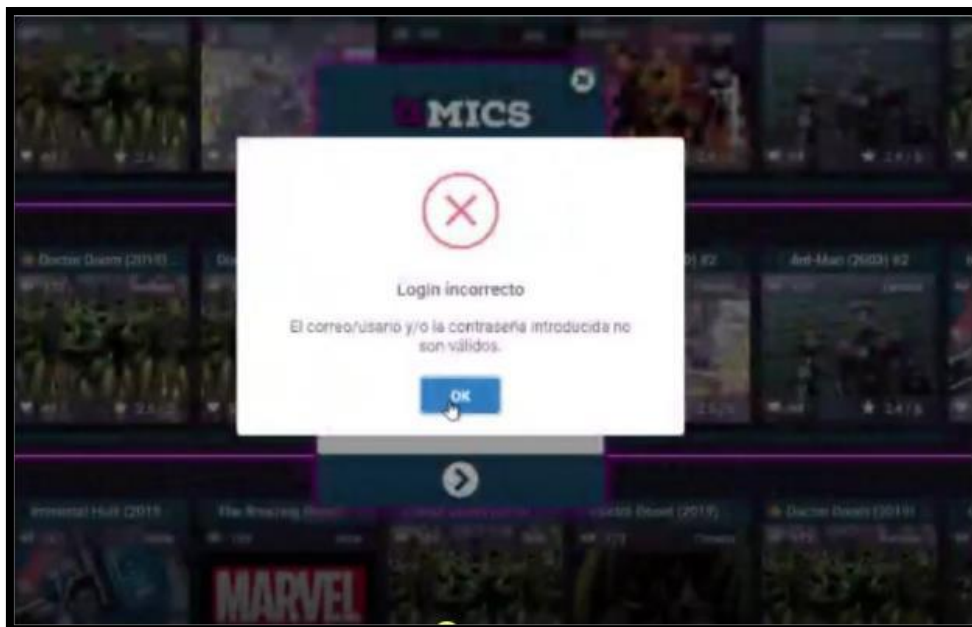
### Errores:

- ❖ Usuario ingresa contraseña incorrecta, no puede eliminar la cuenta.
- ❖ Usuario tiene eliminada su cuenta e intenta iniciar sesión en la plataforma hasta que vuelva a registrarse en la plataforma.



Anexo N°5 - Figura 1. 26 Usuario ingresa contraseña incorrecta, no puede eliminar la cuenta

Al eliminar la cuenta, el usuario ya no podrá ingresar si intenta loguearse con su antigua cuenta. En ese caso le aparece un mensaje de error.



Anexo N°5 - Figura 1. 27 Pantalla cuenta eliminada, el usuario ya no puede acceder.

## 4.6. Errores de Pantalla Introducir Pago

### Errores:

- ❖ Nombre ingresado no es igual al que figura en la tarjeta.
- ❖ Número de tarjeta ingresado incorrecto.
- ❖ Código de verificación (CVC) tiene fecha de vencimiento actual.

- ❖ Saldo de tarjeta insuficiente para realizar el pago de la suscripción.

The image shows a payment form with the following fields:

- Monto a pagar: 740.00
- Tipo de tarjeta: Master Card
- Banco: BANORTE
- Nombre del titular: Roberto González

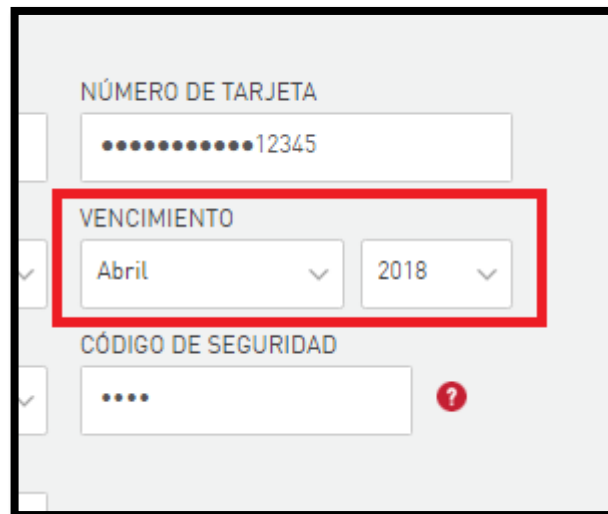
The 'Nombre del titular' field is highlighted with a red border.

Anexo N°5 - Figura 1. 28 Nombre ingresado no es igual al que figura en la tarjeta.

The image shows a payment form with the following fields:

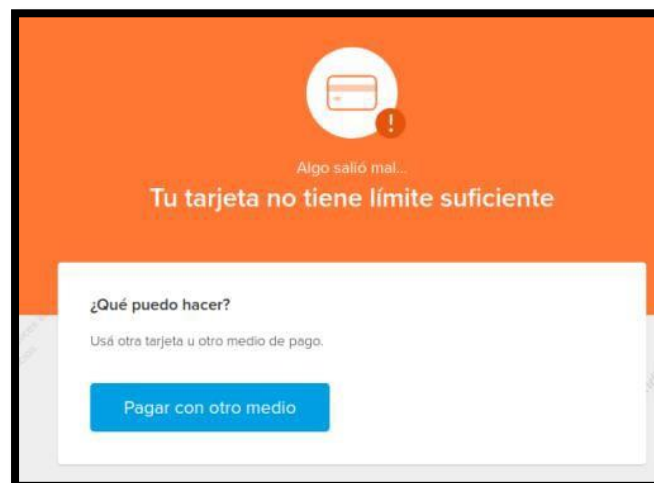
- Usar datos de comprador
- Tipo de tarjeta: Visa Debito
- Titular de la tarjeta: Indica tu nombre y apellidos
- Número de tarjeta: 4356-6870-2301-3333 (marked as incorrect with a red X and the word 'Incorrecto')
- Caducidad: Mes, Año
- Código CVV

Anexo N°5 - Figura 1. 29 Número de tarjeta ingresado incorrecto.



The image shows a payment form with three main sections: 'NÚMERO DE TARJETA' (Card Number) with a field containing '.....12345'; 'VENCIMIENTO' (Expiration Date) with two dropdown menus showing 'Abril' and '2018', which are highlighted by a red rectangular box; and 'CÓDIGO DE SEGURIDAD' (Security Code) with a field containing '....' and a red question mark icon to its right.

Anexo N°5 - Figura 1. 30 Código de verificación tiene fecha actual.

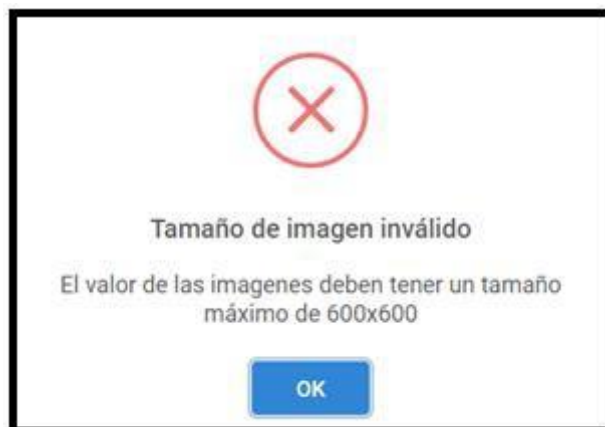


Anexo N°5 - Figura 1. 31 Saldo de la tarjeta insuficiente

## 4.7. Errores de Pantalla Publicar Obra

### Errores:

- ❖ Que el Autor suba una publicación con una resolución inadecuada a la permitida.
- ❖ No seleccionó página alguna publicación a publicar.
- ❖ No colocó Título de la Publicación.
- ❖ No colocó una Descripción de la Publicación.



Anexo N°5 - Figura 1. 32 Resolución inadecuada a la permitida para imágenes.

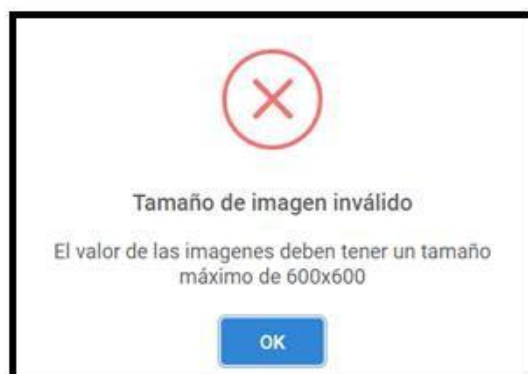


Anexo N°5 - Figura 1. 33 No seleccionó página alguna publicación a subir

## 4.8. Errores de Pantalla Crear Colecciones

### Errores:

- ❖ Autor subió un tamaño no permitido de Portada (Cover).

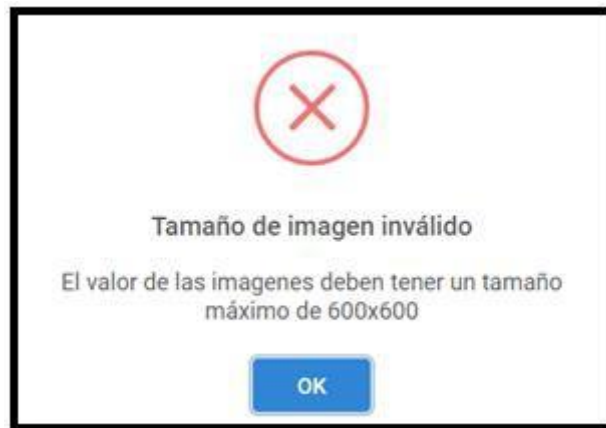


Anexo N°5 - Figura 1. 34 Resolución inadecuada a la permitida para portada.

## 4.9. Errores de Pantalla Editar Colección

### Errores:

- ❖ Autor subió un tamaño no permitido de Portada (Cover).



Anexo N°5 - Figura 1. 35 Resolución inadecuada a la permitida para portada.

## 4.10. Errores de Gestionar Colecciones

### Errores:

- ❖ Un usuario intenta ingresar a una ruta de la plataforma que no tiene permisos para poder gestionar colecciones.

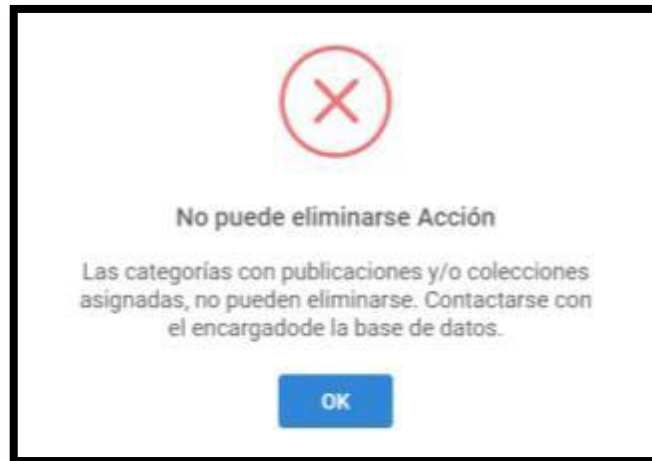


Anexo N°5 - Figura 1. 36 Ingreso a una ruta que no tiene permisos.

## 4.11. Errores de Gestionar Categorías

### Errores:

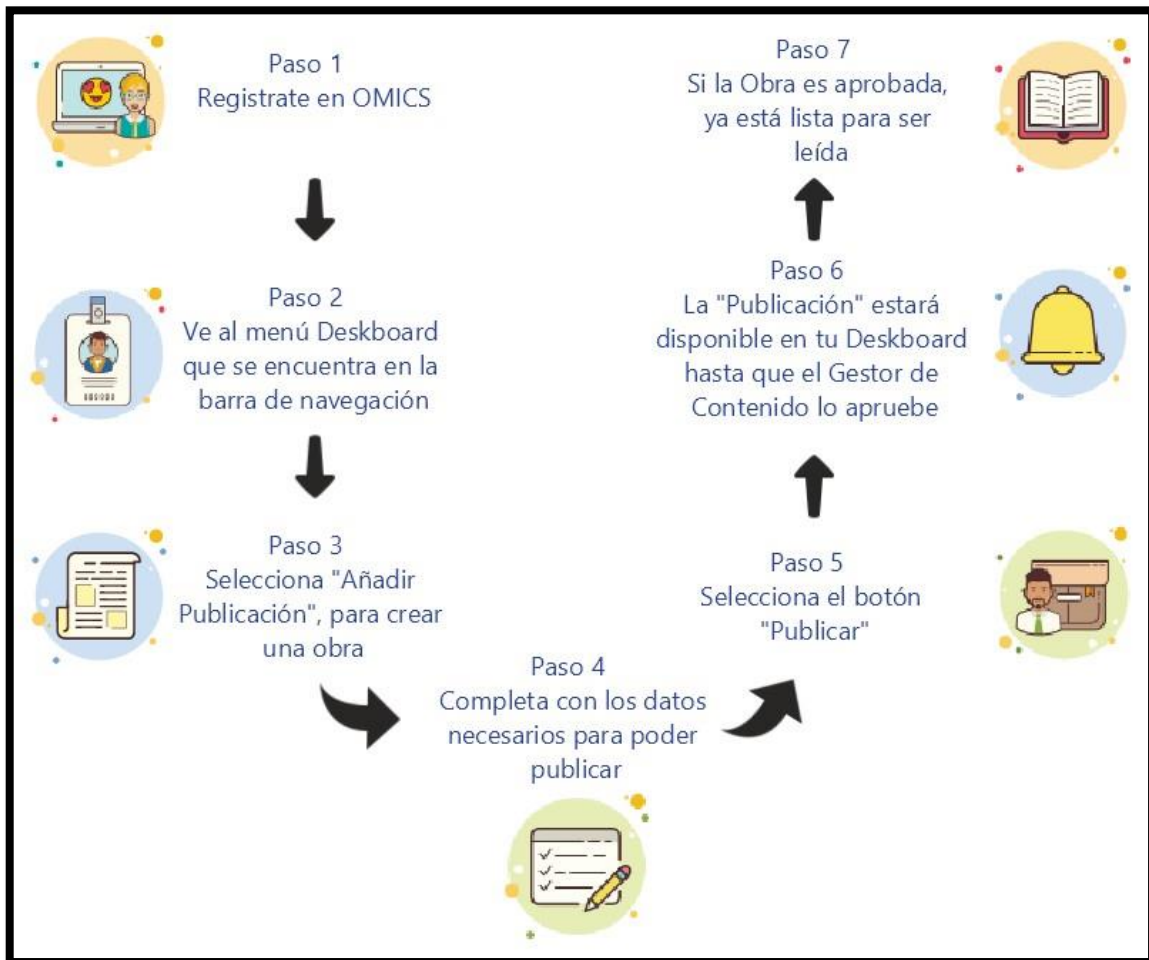
- ❖ Que se quiera eliminar una categoría que tiene asignada publicaciones y /o colecciones.



*Anexo N°5 - Figura 1. 37 No se puede eliminar Categoría.*



## 5. Guía de creadores de contenido



Anexo N°5 - Diagrama 1. 1 Flujo de Publicación de Obra.

Para mayor información sobre publicación de obra, se deben consultar los vídeos tutoriales **VER EN SECCIÓN 10**.

## 6. Políticas de Privacidad

### Omics- Elementos Relevantes de la Política de Privacidad

El Sistema Líder de Gestión Inteligente Digital de Cómic, Omics, registrado para el Proyecto Final de la Carrera de Ingeniería en Sistemas de Información, República Argentina, en la provincia de Mendoza, entiende sus preocupaciones en materia de privacidad. La Política de Privacidad se aplica a la información que se obtiene a través de la plataforma y/o el sitio de Omics (conjuntamente, los "Servicios").

### Visión general de la política de privacidad

Para introducir la Política de Privacidad de Omics, describimos los diferentes tipos de información personal que se recolectan a través de los Servicios, la forma en que se usa dicha información, las personas con las que se pueden compartir y las opciones que usted tiene en relación con el uso de dicha información y el ejercicio de sus derechos. De igual forma, se describen las medidas que se toman para proteger la información. Además, como usuario y miembro de la plataforma, se les informa cómo contactarse en relación con las prácticas de privacidad que se usan.

### A) Información que se recolecta

A través de los Servicios, se recolecta la siguiente información personal:

- ❖ Información que usted proporciona directamente.  
Algunos de los Servicios permiten que Usted nos proporcione información directamente. Por ejemplo:
  - Al momento en que Usted cree su cuenta para ingresar a la plataforma y cuando acceda a dicha cuenta una vez creada, se puede necesitar su nombre de usuario y dirección de correo electrónico.
  - Cuando usted accede al perfil de la cuenta que tenga con Omics, se le puede solicitar información tal como su nombre de usuario, nombre y apellido, fotografía de perfil, género, fecha de nacimiento y dirección de correo electrónico. Foto/Imagen.
  - Para que se pueda responder cuando Usted contacte a Atención a Usuarios, se recolecta la información que Usted decida compartir, incluyendo sus solicitudes y preguntas.
  - Para realizar un pago (donación y/o suscripción) se solicitan datos personales para vincular al servicio externo de Mercado Pago incluyendo nombre de usuario, DNI y CVU (clave virtual uniforme) para que se genere el token único de cobro y registrarlo en su Perfil dentro de la plataforma para efectos fiscales.

- ❖ Información respecto del uso que usted hace de los servicios.

Se recopila información en relación al uso que Usted haga de los Servicios a través de la plataforma en sus dispositivos y por otros medios. Al efecto, se recogen:

- Información sobre el Uso del Servicio: la interacción con los Servicios (tales como términos de búsqueda, y las aplicaciones y características que se utilizan), fechas y horario de uso de los Servicios, información sobre el uso de los Servicios, la cual es recolectada a través de cookies. Para mayor información respecto de cómo se usan las cookies para el Servicio de Omics, consulte la Política de Cookies.
  - Información relacionada con la organización de concursos, sorteos u otras operaciones promocionales: Fecha de participación, respuestas a los concursos y naturaleza de los premios ofrecidos.
  - Información de Dispositivos y Accesos: Dispositivos e información de acceso incluyendo, modelo de dispositivo, versión del Sistema Operativo, dirección IP, configuración de dispositivos, país de ubicación basado en la dirección IP, URL y tipo de red.
  - Datos de Operación: Únicamente se procesa información técnica que no permite identificarlo (p.e., número de operación) para validar las operaciones.
- ❖ Información recolectada a través de terceros analíticos.
    - Mediante ciertos Servicios, se puede recaudar información personal relativa a las actividades del usuario en línea en páginas de Internet y dispositivos conectados en distintos momentos, y a través de sitios de Internet, dispositivos, aplicaciones y otros servicios y funcionalidades de terceros. Se pueden usar servicios analíticos de terceros en los Servicios, tales como Google Analytics. La información que se obtiene puede ser divulgada o recopilada directamente por estos proveedores y otros terceros que utilicen la información, p.e., para evaluar el uso de los Servicios o que ayuden en la administración de los Servicios. Para mayor información

[HTTPS://WWW.GOOGLE.COM/POLICIES/PRIVACY/PARTNERS/](https://www.google.com/policies/privacy/partners/).

## **B) Cómo usamos la información que recolectamos.**

- ❖ Prestación de servicios basados en su relación contractual con Omics.
  - Prestar un Servicio o característica que usted solicite, incluyendo responder a su solicitud, preguntas e instrucciones a través de los Servicios.
  - Conducir encuestas y análisis necesarios para mejorar el Servicio.

- Enviarle información y premios en relación con campañas y eventos para incentivar el uso de la plataforma.
  - Procesamiento de pagos cuando un usuario solicite la suscripción, cuando se realice una donación y cuando se les pague a los autores que tengan contrato o les hayan hecho una donación.
  - Proporcionarle funcionalidades personalizadas tales como recomendaciones basadas en las páginas de Internet que más visite y contenido adicional personalizado a sus intereses a través de la pestaña de "Intereses".
  - Identificarle y autenticarle a fin de que pueda usar ciertos Servicios.
  - Responder a sus preguntas y envío a usuarios de notificaciones importantes cuando sea necesario.
  - Verificar su identidad a fin de cumplir con la legislación federal aplicable.
  - Enviar correos electrónicos en relación con el uso que haga de los Servicios, según sus preferencias de comunicación vía correo electrónico.
  - Dar respuesta a sus preguntas.
  - Cumplir y aplicar los requerimientos de la legislación aplicable, nuestras obligaciones legales (p.e., obligaciones fiscales), los estándares de la industria y nuestras políticas, incluyendo esta Política de Privacidad.
- ❖ Uso de información para otros propósitos.
- Protección contra, identificación y prevención de fraudes y otras conductas criminales, denuncias y otras contingencias.
  - Generación de información estadística relativa al uso de los servicios.
  - Asegurar y monitorear la efectividad y disponibilidad de los servicios a fin de mejorar el funcionamiento de la página de Internet y la aplicación.
  - Cumplir con lo detallado y aplicar los requerimientos aplicables conforme a la ley, los estándares de la industria y nuestras políticas, incluyendo esta Política de Privacidad.

### **C) Con quién compartimos su información.**

Se divulga información personal relacionada con Usted a través de los Servicios, según lo que se describe a continuación:

- ❖ Proveedores de Servicios: Se puede divulgar su información a terceros que presten Servicios por cuenta de Omics, tales como empresas que ayudan a desarrollar y operar sistemas para el Servicio con centros de contacto con usuarios, así como las actividades de atención al cliente, mercadotecnia (incluyendo anuncios personalizados en las páginas de Internet de Omics, páginas de Internet de terceros, o plataformas en línea), llevar a cabo encuestas

de satisfacción del cliente, facturación o que envíen correos electrónicos a nombre de la plataforma. A través de disposiciones contractuales estas entidades están limitadas respecto del uso que puedan dar a su información para fines distintos a la prestación de los Servicios en favor al trabajo que realiza el equipo de Omics.

- ❖ Otros Terceros cuando sea Requerido por Ley o Necesario para Proteger los Servicios de Omics: Pueden darse casos en los que se divulguen su información a terceros:
  - Para cumplir con la ley o dar respuesta a cualquier proceso legal (tales como órdenes de búsqueda u otras órdenes judiciales).
  - Para verificar y asegurar el cumplimiento con las políticas que rigen los Servicios de Omics.
  - Para proteger los derechos de Omics, propiedades o seguridad o en cumplimiento de las leyes de protección de datos, de las respectivas afiliadas, socios comerciales o usuarios.
- ❖ Otros Terceros en Relación con Operaciones Societarias: Se puede divulgar su información a terceros como parte de fusiones, transferencias, adquisiciones o ventas, actuales o potenciales, o en caso de quiebra.
- ❖ Otros Terceros con su Consentimiento o a su Solicitud: En adición a las opciones descritas en esta Política de Privacidad, se puede compartir su información con terceros cuando Usted lo convenga o solicite por separado.
- ❖ Hipervínculos a Otras Páginas de Internet, Dispositivos, Aplicaciones y Funcionalidades: El Servicio puede permitirle conectar con otras páginas de Internet, dispositivos, aplicaciones y otras funcionalidades que operan independientemente de Omics y tener sus propios avisos o políticas de privacidad, las que enfáticamente, se le sugiere revisar. Omics no se hace responsable del contenido, uso o prácticas de privacidad de cualquier página de Internet, dispositivo, aplicación u otra funcionalidad que esté vinculada pero que no sea propiedad del equipo o sea controlada por la misma.

## **D) Retención de información personal.**

Se toman medidas adecuadas para asegurar que se procesa y retiene información del usuario, conforme a lo siguiente:

- ❖ Al menos durante el término por el cual la información sea utilizada para prestarle un servicio.
- ❖ En los términos que requiera la ley, un contrato, o de conformidad con las obligaciones legales (p.e., leyes fiscales).

- ❖ Únicamente durante el tiempo necesario para el objeto con el que fue recabada o procesada, o durante un mayor plazo de ser requerido conforme a cualquier contrato, por la legislación aplicable, o de forma anónima, para fines estadísticos, sujeto a medidas de protección apropiadas.

## **E) Sus opciones.**

Se ofrecen ciertas opciones en relación con la información personal que se obtiene del usuario. A fin de limitar las comunicaciones que reciba de Omics, o para presentar una solicitud, sírvase contactarse conforme a lo señalado en la sección Contáctenos (sección J de esta Política de Privacidad). Para mayor información respecto al uso de las cookies que se recolectan, por favor consulte "política de cookies". Usted puede también restringir o deshabilitar ciertas cookies visitando a:

[HTTP://WWW.ALLABOUTCOOKIES.ORG](http://www.allaboutcookies.org)

[HTTPS://OPTOUT.NETWORKADVERTISING.ORG/?C=1](https://optout.networkadvertising.org/?c=1)

## **F) Sus derechos.**

La información personal del usuario es propiedad del mismo. Se puede pedir que el equipo de Omics le proporcione detalles de la información que se recolecta y puede solicitar corregir cualquier inexactitud. Desde el equipo, se puede no dar trámite a solicitudes que sean irrazonablemente repetitivas, que requieran esfuerzos técnicos desproporcionados, poner en riesgo la privacidad de terceros, sean extremadamente imprácticas o que la legislación aplicable local no requiera dicho acceso. Para presentar una solicitud en relación con sus derechos o para hacer una consulta, se pueden contactar. Por favor, refiérase a la sección Contáctenos.

## **G) Cómo protegemos la información personal.**

Se toma la protección de la información del usuario con absoluta seriedad y se mantienen protecciones diseñadas para resguardar la información que se adquiere en relación con los Servicios. No obstante, considere que, si bien se toman pasos razonables para proteger su información, ningún sitio de Internet, transmisión por Internet, sistema informático o conexión inalámbrica es completamente seguro.

## **H) Transferencia Internacional de información.**

El uso que el usuario haga del Servicio disponible en Omics implica la transmisión, almacenamiento y procesamiento de su información personal dentro y fuera de su país

de residencia en términos de esta política. Por favor, considere que las leyes de protección de datos y otras leyes de los países a los que su información sea transferida, pueden no ser tan amplias como las de su país.

## **I) Actualizaciones a Nuestra Política de Privacidad.**

La Política de Privacidad puede ser actualizada para reflejar cambios en las prácticas en materia de información personal en relación con el Servicio, o cambios en la legislación aplicable. Se publica un aviso en el Servicio de Omics para informarle con anticipación a los usuarios de los cambios sustanciales de la Política de Privacidad y se indican, al principio del aviso, la fecha más reciente de actualización.

## **J) Contáctenos.**

Para más información respecto de las prácticas de privacidad, o para ejercer cualquier derecho que pudiera tener conforme a la legislación aplicable, sírvase contactar al departamento de Omics de manejo de información personal en el Formulario de consultas en la sección de ayuda o vía correo a dirigido a [omics.contacto@gmail.com](mailto:omics.contacto@gmail.com).

## **K) Información personal de menores.**

Los Servicios han sido diseñados para una audiencia general y no están dirigidos a menores. El equipo de Omics no solicita o recauda, a sabiendas, información personal de menores de 13 años, con motivo de los Servicios. Si se llega a tener conocimiento de se ha recabado información de un menor de 13 años, de inmediato eliminaremos dicha información. Si algún usuario considera que un menor de 13 años pudo haber proporcionado información personal, sírvase contactarse conforme a lo que se indica en la sección Contáctenos de esta Política de Privacidad.

## 7. Términos de Uso de la plataforma

### A) Aceptación de las Condiciones de Uso

OMICS Digital Reader (Sistema Líder de Gestión Inteligente Digital de Cómic) le presta los servicios en, para, o a través del Sitio Web OMICS (en conjunto, el "Servicio") y con matriz argentina, a todos los usuarios inscriptos para el uso de las funcionalidades que se brindan a través de la misma.

Al usar el Sitio Web OMICS y/o usar el Servicio, y/o al hacer clic en un botón indicando su consentimiento cuando esta opción le sea proporcionada, usted, y en caso de ser Menor, su Representante, acepta (1) las Condiciones de Uso de OMICS, (2) la Política de Privacidad, (3) la Política de la Comunidad y las Directrices de Carga, todos los cuales son incorporados en este documento por referencia. El contrato constituye el acuerdo completo entre la Empresa y Usted. Si no es su voluntad convenir a los términos del Contrato, Usted no deberá acceder a o utilizar los Servicios. Este Contrato aplica para todos los usuarios del Servicio, incluyendo usuarios que puedan hacer contribuciones al Contenido Digital y/o a las Publicaciones de Usuarios. Por favor, sírvase leer la totalidad del Contrato, según éste ha sido publicado en el Sitio Web OMICS antes de usar el Servicio.

Adicionalmente, al utilizar el Servicio, usted estará sujeto a los lineamientos o reglas aplicables a dicho servicio, sus características o a las ofertas que se publiquen de tiempo en tiempo. Dichas guías o reglas se incorporan al presente Contrato por referencia.

### B) Cambios a las condiciones de Uso

La Empresa podrá modificar en cualquier tiempo este Contrato a través de la publicación de las versiones modificadas en el Sitio Web OMICS. Usted acepta que, con dicha publicación, la Empresa le ha notificado debidamente la modificación correspondiente. Si bien, haremos todo lo posible por notificarle cualquier cambio sustancial a este Contrato, Usted deberá revisar periódicamente la versión actualizada del Contrato en el sitio web de OMICS.

### C) Definiciones

Para efectos de estas Condiciones de Uso de OMICS:

- ❖ "Contenido Digital" significa los webcómic, y cómic digitalizados, incluyendo las ediciones únicas y publicaciones realizadas, y cualquier otro



contenido digital incluido que se ofrezcan en el Servicio de la plataforma de Omics.

- ❖ "Publicaciones de Usuarios" significa ciertas funciones de nuestro Servicio que permite a Usted o a otros usuarios finales publicar en nuestro Servicio, o enviar a la Empresa contenido que incluya texto, datos, fotografías, imágenes, información, y cualquier combinación de dichos elementos.
- ❖ "Donación de usuarios" (ya sean lectores u otros autores): Son aquellos que pueden realizar una colaboración monetaria para generar un beneficio económico a autores e incentivar para que crezca su contenido.
- ❖ "Suscripción" significa que todos los usuarios con un pago mínimo mensual pueden acceder a la plataforma con el agregado de contenido Premium para consumir, con algunas funcionalidades extras para que puedan utilizar.
- ❖ "Contrato de publicaciones" significa que una vez que una publicación alcanza cierta cantidad de visualizaciones puede solicitar un contrato para la misma y de esta forma tener un beneficio económico extra para tener más ganancias.
- ❖ Para más noticias de la plataforma de cómics, se puede acceder a la red social de Omics. Significa <HTTPS://WWW.FACEBOOK.COM/OMICSREADER/>

## **D) Contenido Digital y Plazo de Licencia Limitada**

- ❖ Contenido Digital. Usted entiende y acepta que el Sitio Web OMICS son propiedad de OMICS Digital Reader. El Contenido Digital contenido en el Servicio es propiedad de OMICS Digital Reader o es materia de licencia otorgada por el propietario de los derechos de autor OMICS Digital Reader. El Sitio Web OMICS y el Contenido Digital están protegidos por las leyes del derecho de autor en la República de Argentina, así como otras leyes y tratados en materia de propiedad intelectual. OMICS Digital Reader no le transfiere ningún título, derecho, interés respecto del Sitio Web OMICS y/o el Contenido Digital.
- ❖ Uso del Contenido Digital. Al momento usar el Contenido Digital, OMICS Digital Reader le otorga una licencia limitada, no exclusiva, no transferible, no sub-licenciable, y revocable, para acceder, ver y utilizar dicho Contenido Digital, de manera personal y no para fines comerciales, conforme a los términos de este Contrato. La licencia no le otorga ningún derecho de propiedad respecto de dicho Contenido Digital. En el Sitio Web OMICS, usted podrá acceder, ver o utilizar el Contenido Digital. OMICS Digital Reader concede a Usted una licencia respecto del

Contenido Digital, en el entendido que el mismo no le es vendido, transferido o cedido.

- ❖ Limitaciones. Salvo que OMICS Digital Reader disponga lo contrario por escrito, Usted se abstendrá de copiar, redistribuir, vender, transmitir, retransmitir o sublicenciar el Contenido Digital. Usted se deberá abstener de intentar eludir, evitar, omitir, eliminar, desactivar, impedir o de otro modo anular cualquier tecnología de cifrado, señalización de derechos o tecnología de protección contra copia incluido en la plataforma o asociado con éste el Contenido Digital, y se abstendrá de editar, modificar, traducir o crear trabajos derivados a partir de los trabajos publicados o adaptaciones del Contenido Digital. Usted se abstendrá de duplicar o de cualquier otra forma reproducir (incluyendo, pero no limitado a grabar) el Contenido Digital, o cualquier parte del mismo, en cualquier medio físico o memoria o dispositivo (salvo respecto de su dispositivo personal autorizado para descargar Contenido Digital hasta por un periodo de treinta días), incluyendo sin limitar a CDs, DVDs, computadoras u otro hardware, o cualquier otro medio conocido o en el presente o en el futuro.

## E) Sitio Web OMICS

- ❖ Uso del Sitio Web OMICS. Usted podrá utilizar el Sitio Web OMICS, incluyendo el software asociado a ellos. Usted deberá abstenerse de separar componentes individuales de dicho software para utilizarlo en otro dispositivo o computadora, no podrá transferirlo para usarlo en otro dispositivo o computadora o usarlo, total o parcialmente, a través de una red, y no podrá vender, prestar, distribuir o sublicenciar, o de cualquier otra forma ceder, cualquier derecho sobre el software, ya sea total o parcialmente.
- ❖ Actualizaciones. Para mantener el Sitio Web OMICS actualizados, OMICS Digital Reader puede poner a disposición actualizaciones / mejoras al Sitio Web OMICS. Si Usted no descarga dichas actualizaciones, es posible que no pierda el beneficio de ciertas características, funciones o avisos del Sitio Web OMICS.
- ❖ Prohibición de Aplicación de Ingeniería Inversa, Desagregación, Desmontaje, o Contravención. Usted deberá abstenerse de modificar, aplicar ingeniería inversa, descompilar, reproducir, copiar, desmontar el Sitio Web OMICS, o de cualquier otra forma reducir el archivo de computadora en el que el Contenido Digital se almacena, a una forma legible, ya sea en su totalidad o en parte, crear cualquier obra derivada del Sitio Web OMICS, u omitir, modificar, afectar, o alterar o eludir

cualquiera de las funciones o protecciones del Sitio Web OMICS; por ejemplo, modificando, reduciendo, aumentando, o sustituyendo cualquier función de gestión de derechos digitales.

## F) Condiciones de Uso

- ❖ Límite de Edad para los Usuarios. El usuario del Servicio debe tener por lo menos catorce (14) años (la "Edad Mínima"). El Uso del Servicio por cualquier persona que no haya cumplido la Edad Mínima no está autorizado y respecto de dichas personas, el equipo de Omics no le concede licencia alguna. Si no cumple con este requisito, Usted no deberá utilizar el Servicio.
- ❖ Si Usted tiene entre 14 y 18 años (o está entre la Edad Mínima y la mayoría de edad legal en su respectiva jurisdicción, un "Menor(es)"), solo podrá utilizar el Servicio bajo la supervisión de sus padres o de quien ejerza su representación legal (el/los "Representante(s)" los cuales quedan obligados bajo el Contrato.
- ❖ En caso de que usted sea un Menor, su Representante, declara al equipo de Omics que el Usuario (1) tiene, al menos, la Edad Mínima y (2) en caso de ser un Menor, solo utilizará el Servicio con el consentimiento y bajo la supervisión de un Representante que queda obligado con Omics bajo el Contrato, y que el uso del Servicio es para propósitos personales. Podremos solicitarle que nos proporcione evidencia del consentimiento de su Representante o de su edad. Así mismo, podremos dar por terminado el uso del Servicio si llegara a ser de nuestro conocimiento que cualquiera de las declaraciones que usted hace no es correcta. Usted, y si es Menor su Representante, declara que, ha leído y aceptado estas Condiciones de Uso y la Política de Privacidad antes de iniciar el uso del Servicio o cualquier otro producto o servicio que OMICS Digital Reader ofrece. Sugerimos a los Representantes que permiten a sus representados utilizar un servicio interactivo, que conversen con ellos sobre la seguridad en línea, así como de la clasificación de contenidos. Los Representantes que permitan a Menores utilizar cualquier servicio interactivo deberán informar a sus representados de los riesgos potenciales que esto representa. Los Representantes serán los responsables de las acciones u omisiones de los Menores que representen.
- ❖ El equipo de Omics no recopila, intencional o accidentalmente, datos de niños o Menores en los casos en que los Representantes no han consentido que el Menor utilice el Servicio. Sin embargo, si un Representante descubre que un Menor nos ha proporcionado información personal sin

el permiso del Representante respectivo, el Representante deberá comunicarse con el equipo de Omics a través de [OMICS.CONTACTO@GMAIL.COM](mailto:OMICS.CONTACTO@GMAIL.COM). Tras la solicitud del Representante, el equipo de Omics eliminará cualquier información personal de dicho Menor sin necesidad de actos posteriores del Representante. El Representante deberá indemnizar y sacar a salvo al equipo de Omics de cualquier obligación, daños, demandas o cualquier otra acción legal relacionada con la transmisión de datos personales por el Menor.

- ❖ Cumplimiento de la Ley y Reserva de Derechos. Usted deberá utilizar el Sitio Web OMICS y/o el Contenido Digital de conformidad con todas las leyes aplicables, incluyendo, sin limitar, las de su jurisdicción. Las licencias otorgadas bajo este Contrato son no exclusivas y cualquier derecho que no esté expresamente concedidos en este Contrato quedarán reservados para OMICS Digital Reader.

## G) Privacidad

La Empresa respeta su privacidad y ha establecido ciertas políticas y procedimientos relativos a la recopilación y uso de sus datos personales en relación con el uso del Servicio. Nuestra Política de Privacidad se encuentra disponible en el Sitio Web OMICS y en el apartado 6 del Manual de usuario, y es aplicable al uso que Usted haga del Servicio. Al utilizar el Servicio y/o indicar su consentimiento a la Política de Privacidad, acepta a los términos de nuestra Política de Privacidad, según ésta sea modificada de tiempo en tiempo.

No recopilamos a sabiendas información personal de niños menos de 14 años. Cualquier padre o tutor que crea que un niño puede haber proporcionado información personal debe comunicarse con nosotros en Atención al Cliente. En tal caso, tomaremos las medidas necesarias para eliminar inmediatamente la información personal de niños menores de 14 años.

## H) Propiedad Intelectual y Otros Derechos

El Servicio es propiedad de y es operado por OMICS Digital Reader, y el Contenido Digital ha sido otorgado bajo licencia por el propietario de los derechos de autor a OMICS Digital Reader. El Contenido Digital, el software, las interfaces visuales, los elementos interactivos, las características, la información, los gráficos, los diseños, la compilación, el código de computadora, los productos, la apariencia y todos los demás elementos del Servicio, están protegidos por las leyes de la República de Argentina relativas a la protección de propiedad intelectual y los derechos de propiedad, incluyendo las leyes en materia de derechos de autor, patentes, imagen comercial y

marcas registradas, así como las leyes relacionadas con la protección y seguridad de información. Todo el Contenido Digital y el Servicio, incluyendo los derechos de propiedad intelectual aquí contenidos y relativos a los anteriores, son propiedad de OMICS Digital Reader o sus subsidiarias o afiliadas y/o terceros licenciantes. Usted deberá abstenerse de reproducir, distribuir, rentar, arrendar, vender, licenciar, copiar, modificar, realizar o exhibir públicamente, transferir, transmitir, publicar, editar, adaptar, preparar trabajos derivados o basados en, o de otra manera usar Contenido Digital, excepto que esté expresamente autorizado por este Contrato. OMICS Digital Reader se reserva todos los derechos no otorgados expresamente en este Contrato.

## **I) Publicidad**

La Empresa puede proporcionarle anuncios propios o de terceros en línea, entre otros, en el Sitio Web OMICS de conformidad con el Contrato. Los anuncios incluyen, pero no están limitados, entre otros, la colocación de mensajes promocionales (incluida cualquier tecnología relacionada) con el fin de publicitar los productos o servicios de un anunciante externo. Los anuncios pueden consistir en texto, gráficos, audio y/o video o cualquier combinación de estos y podrá direccionar al usuario a un enlace externo (por ejemplo, una página destino).

## **J) Suspensión de Acceso**

Podemos modificar, suspender o discontinuar el acceso a cierto Contenido Digital en caso de cualquier eventualidad relativa a derechos o cualquier otra incidencia de negocios o legal.

## **K) Terminación**

Sus derechos bajo este Contrato terminarán automáticamente si no cumple con los términos de este Contrato. En caso de dicha terminación, deberá suspender el uso del Servicio y/o Contenido Digital y el equipo de Omics se reserva el derecho de revocar inmediatamente su acceso al Servicio y/o Contenido Digital. La falta de ejercicio de cualquier derecho o condición de este Contrato por el equipo de Omics, no constituirá renuncia alguna a dicho derecho o condición ni a cualquier derecho o recurso disponible para el mismo.

## **L) Ausencia de Garantías**

El uso del sitio web *Omics* y/o el contenido digital es bajo su propio riesgo. Ninguna información oral o escrita ni ningún consejo dado por el equipo o cualquier representante autorizado de *Omics* constituye una garantía. El sitio web *Omics*, y/o el contenido digital se proporcionan en los términos en que se encuentran, con todas las

fallas y sin garantía de ningún tipo, y ni la empresa ni sus licenciantes otorgan garantías expresas, implícitas o legales, incluyendo sin limitar, garantías implícitas o condiciones de comerciabilidad, idoneidad para un propósito particular, exactitud, uso inalterado de posesión, calidad satisfactoria, no infracción de derechos de terceros y sus equivalentes bajo las leyes de cualquier jurisdicción. La empresa no hace declaraciones o garantías respecto a que el servicio o el acceso y uso del servicio por usted será ininterrumpido o sin errores, libre de virus, de códigos maliciosos o de cualquier otro componente dañino o que es seguro. El equipo de *Omics* no es responsable de ningún daño a su dispositivo móvil, Tablet, computadora portátil o de escritorio, sistema informático, otro hardware o software, o de cualquier pérdida o daño a datos, por lesiones personales o daños a la propiedad de cualquier naturaleza que puedan resultar del uso del servicio. *Omics* no hace declaraciones o garantías sobre la exactitud o integridad del contenido en el servicio o el contenido de cualquier sitio vinculado a este sitio y no asume obligación o responsabilidad alguna por cualquier error, fallo, o inexactitudes, cualquier error u omisión en cualquier contenido o por cualquier pérdida o daño de cualquier tipo como resultado del uso del contenido publicado, enviado por correo electrónico, transmitido o de otra manera disponible a través de los servicios. Además, no garantiza, avala o asume responsabilidad por cualquiera de los productos o servicios publicitados u ofrecidos por un tercero a través de los servicios o cualquier servicio hipervinculado o presentado en cualquier banner u otra publicidad, y el equipo *Omics* no será parte de, o de cualquier forma responsable de, monitorear cualquier transacción entre usted y proveedores de productos o servicios terceros. Como con la compra de un producto o servicio a través de cualquier medio o en cualquier ambiente, usted deberá utilizar su mejor juicio y ser precavido cuando lo amerite.

## M) Limitación de Responsabilidad

En la medida en que lo permite la ley, el equipo de *Omics* y sus licenciatarios y afiliados no serán responsables por ningún daño incidental o como consecuencia de un incumplimiento de cualquier garantía expresa o implícita, incumplimiento de contrato, negligencia, responsabilidad estricta o cualquier otra teoría legal relacionada al sitio web *Omics* y/o el contenido digital, tales como daños derivados de la pérdida de ganancias, ingresos, datos, uso del sitio web *Omics* y/o el contenido digital o cualquier producto asociado, o cualquier error, o inexactitud de contenido, cualquier error u omisión en cualquier contenido o por cualquier pérdida o daños de cualquier tipo como consecuencia del uso del contenido publicado, enviado por correo electrónico, transmitido o de otro modo disponible a través de los servicios, incluso si el equipo ha sido notificado de la posibilidad de esos daños. El uso del sitio web *Omics* y/o el contenido digital y el software es bajo su propio riesgo. Para el servicio gratuito, el equipo y sus

licenciatarios y afiliadas no serán responsables de cualquier daño o pérdida directa, indirecta, incidental o consecuencial.

## **N) Indemnización**

Usted se obliga a indemnizar, defender y liberar de responsabilidad al equipo Omics y sus afiliadas contra cualquier pérdida, responsabilidad, demanda, acción, y gasto (incluyendo sin limitar, honorarios razonables de abogados y costos de investigación, previos o posteriores a la presentación de una demanda formal y gastos de litigio razonables) que surjan de, o estén relacionados con, el uso del Servicio, sus Publicaciones de Usuario, la violación a este Contrato o a cualquier ley o regulación aplicable.

## **O) Restricciones Geográficas y Ley Aplicable; Controversias**

Brindamos los Servicios desde Argentina. No afirmamos que el Sitio Web OMICS, o cualquiera del Servicio sea accesible o apropiada fuera de la Argentina. El acceso al Sitio Web OMICS o los Servicios OMICS puede no estar permitido legalmente para ciertas personas o en ciertos países. Si accede al Sitio Web OMICS fuera de la Argentina, lo hace por iniciativa propia y Usted es responsable del cumplimiento con las leyes de su jurisdicción.

Usted acepta que todos los asuntos relacionados con el Sitio Web OMICS y el Contrato y cualquier controversia o demanda que surja de los mismos, se interpretará de conformidad con las leyes de Argentina, sin dar efecto a disposición alguna en materia de elección o conflicto de leyes. Usted acepta que cualquier demanda legal, acción o procedimiento que pueda surgir de, o relacionado con este Contrato se tramitará exclusivamente en los tribunales federales de la Argentina. Usted renuncia a todas y cada una de las objeciones al ejercicio de la jurisdicción y competencia de dichos tribunales.

## **P) Integridad del Contrato y Divisibilidad**

El Contrato constituye el contrato íntegro entre usted y OMICS Digital Reader respecto al Servicio y/o Contenido Digital y reemplaza todos los entendimientos previos sobre dicho tema. Si algún término o condición del Contrato se considera inválido, nulo o por cualquier razón inaplicable, esa parte se considerará separable y no afectará la validez y aplicabilidad de cualquier otro término o condición.

## **Q) Modificación**

Podemos modificar cualquiera de los términos del Contrato a nuestro exclusivo criterio publicando el Contrato modificado en el Sitio de Web OMICS. El uso continuado del Servicio y/o el Contenido Digital después de la fecha en que surta efectos dicha

modificación constituye su consentimiento para quedar obligado en términos de dicha modificación.

## **R) Información de Contacto**

Para obtener ayuda con el Servicio y/o Contenido Digital, favor de comunicarse con el equipo de OMICS en [OMICS.CONTACTO@GMAIL.COM](mailto:OMICS.CONTACTO@GMAIL.COM)



## 8. Backup

En este apartado se contempla tanto el realizado de backup, ya sean automáticos como manuales, y, además, a la ejecución de migraciones para actualizar los esquemas del motor de base de datos relacional utilizado: PostgreSQL.

### Para la administración y mantenimiento de la plataforma:

- ❖ Equipos para cada puesto, pertenecientes a los Administradores de Omics.
- ❖ Conexión con la red y acceso a internet para la conexión al servidor, provisto por el coworking.
- ❖ Sistema operativo: Windows y Linux.
- ❖ Insomnia para hacer testing a la API.
- ❖ Editor de código fuente: Visual Studio Code.

### Para los Administradores de la plataforma:

- ❖ Puestos de trabajo en el coworking.
- ❖ Acceso a internet en el coworking.

### 8.1. Requisitos

- ❖ Tener el motor de base de datos PostgreSQL instalado.
- ❖ Tener Docker instalado.
- ❖ Preferentemente, correr en entorno GNU/Linux.
- ❖ Tener **make** instalado.

### 8.2. Carga inicial

En la carga de datos se identifican tres partes diferentes:

#### Carga inicial:

- ❖ Roles: usuario, gestor de contenido y admin.
- ❖ Categorías: Drama, Romance, Comedia, Terror, Superhéroes, Fantasía, Vida diaria, Sci-Fi, Acción.
- ❖ Usuario admin por defecto: admin-1.
- ❖ Usuario gestor de contenido por defecto: content-manager-1.
- ❖ Estados de entidades (Publicación, Suscripción y Contrato).
- ❖ Configuración general de la plataforma (Roles, etc.).

#### Conversión de datos:

Este ítem no aplica inicialmente ya que la plataforma se implementa por primera vez, y no hay datos que cambien de formato.

En caso de que se requieran conversiones entre tipos de datos, las migraciones se encargarán de llevarlos a cabo. [VER SECCIÓN 8.3.4. MIGRACIONES.](#)

**Depuración:**

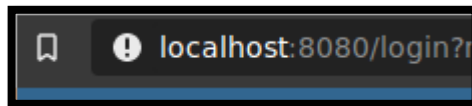
Para realizar la depuración se debe asegurar:

- ❖ Se debe asegurar de hacer un backup de los datos de desarrollo.
- ❖ Que, si faltan datos, se los agregue.
- ❖ Que, si hay datos incorrectos, se los corrija.
- ❖ Que no haya datos duplicados.
- ❖ Que los datos sean válidos en forma lógica (ejemplo: una dirección que no existe no se debería permitir).
- ❖ Integridad de las referencias en las bases de datos.

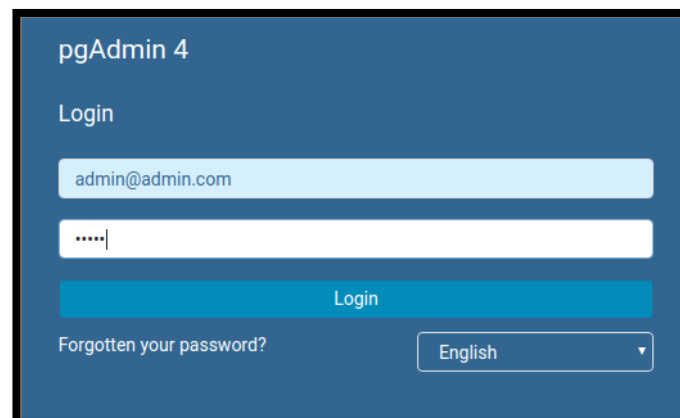
**8.2.1. Creación de la base de datos**

En caso de que la base de datos aún no haya sido creada, debe crearse manualmente. Para esto se accede al panel de administración pgAdmin4, utilizado para gestionar la base de datos de forma gráfica y remota.

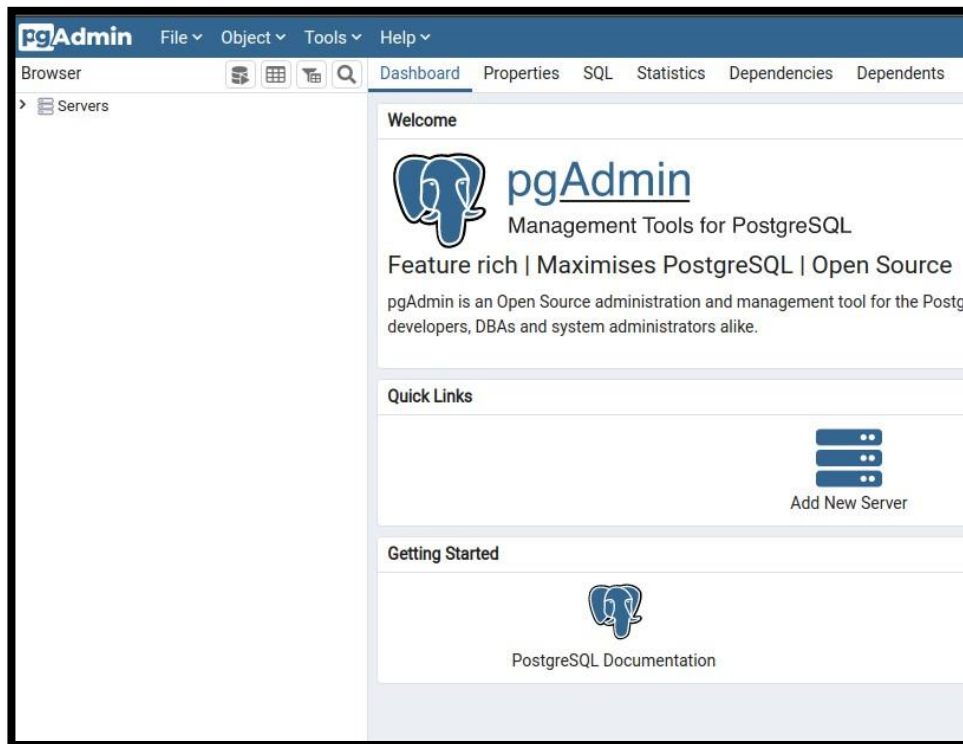
Las imágenes a continuación están a modo ilustrativo. La URL de acceso y el puerto pueden variar acorde al entorno en el que se encuentre corriendo el sistema. En caso de ser así, consultar con un administrador.



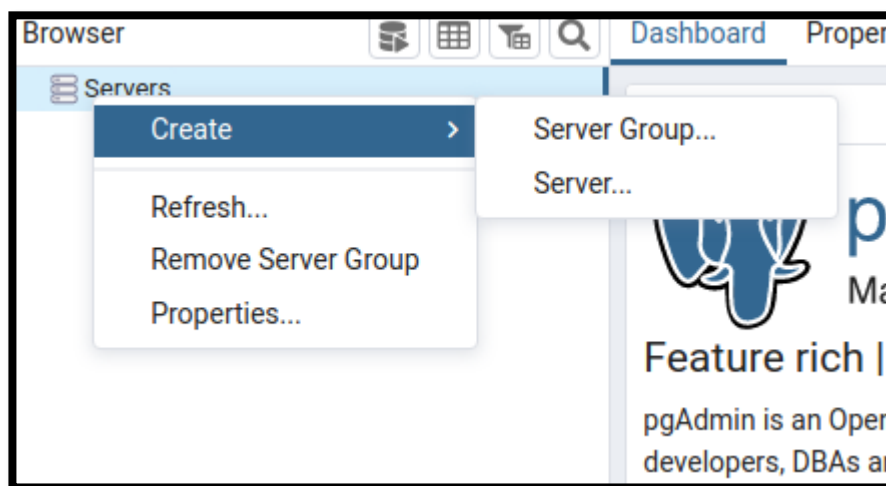
Anexo N°5 - Figura 1. 38 Ingreso a localhost:8080.



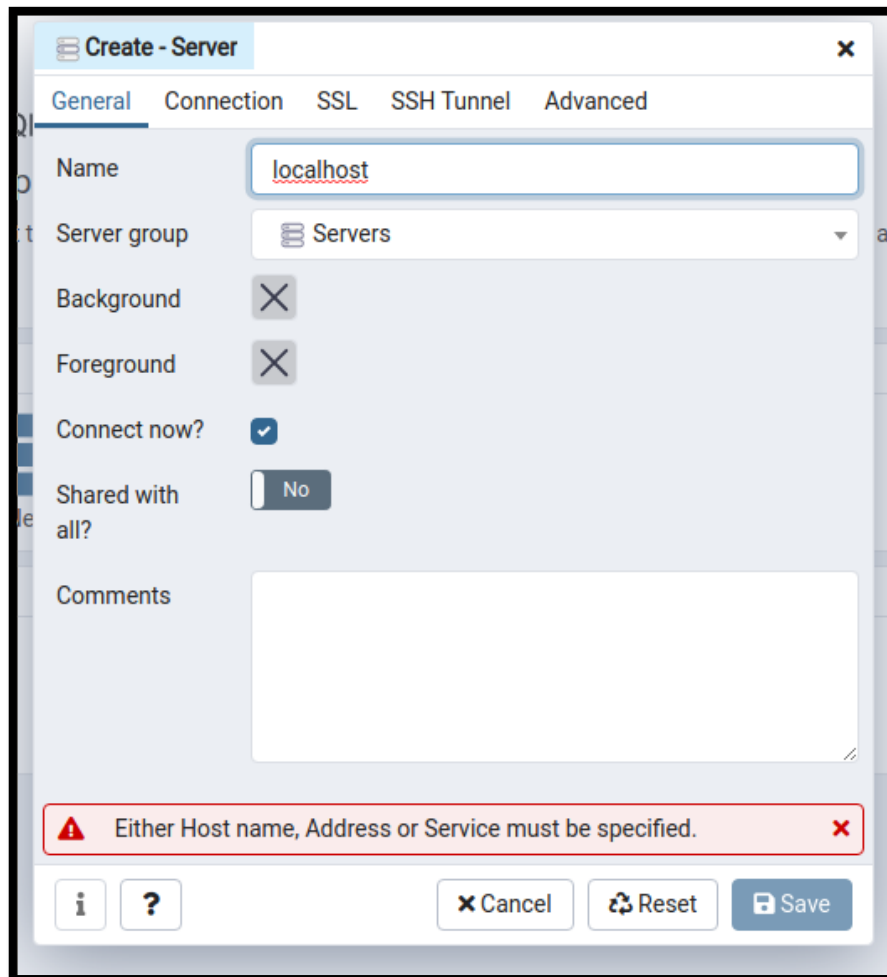
Anexo N°5 - Figura 1. 39 Autenticación del administrador. Estos datos pueden variar por razones de seguridad.



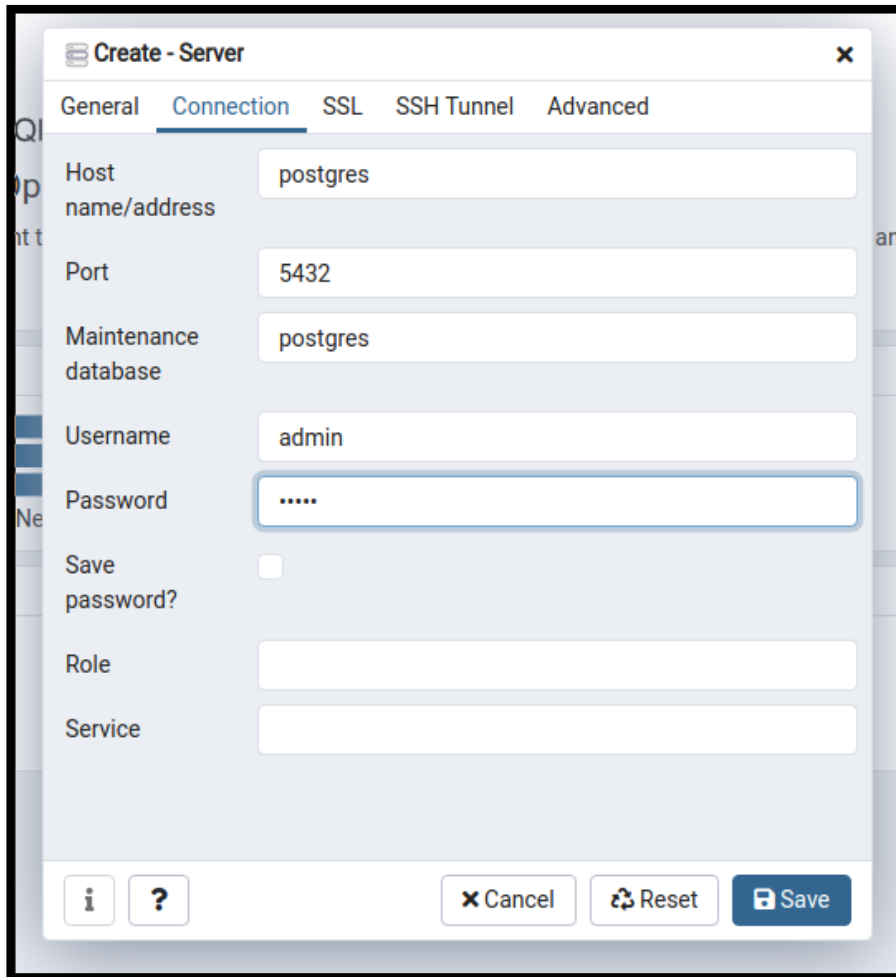
Anexo N°5 - Figura 1. 40 Tablero de administración de PostgreSQL mediante pgAdmin.



Anexo N°5 - Figura 1. 41 Creación de la conexión con el motor de base de datos.

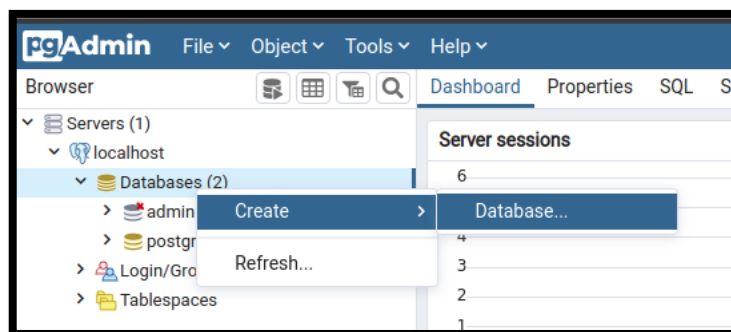


Anexo N°5 - Figura 1. 42 Ingreso de dirección IP donde se encuentra el SGBD corriendo.  
Puerto por defecto.

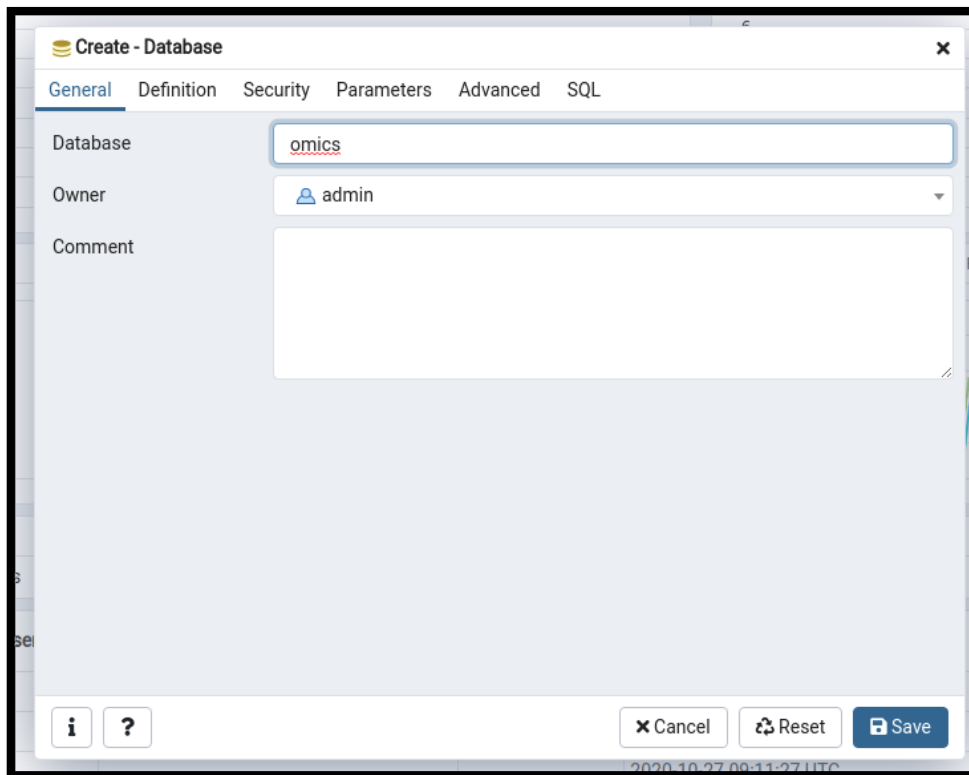


Anexo N°5 - Figura 1. 43 Ingreso de datos de credenciales para acceso a base de datos.

Es necesario considerar que se utiliza la red creada por Docker para lograr la interacción entre los diversos servicios utilizados. Es por esto que utilizamos "postgres" como *hostname*.



Anexo N°5 - Figura 1. 44 Creamos la base de datos.



Anexo N°5 - Figura 1. 45 Base de datos de nombre "omics". Es importante respetar el nombre.

### 8.2.2. Correr las migraciones

Ya que se utiliza un motor de base de datos relacional, es necesario crear los esquemas principales que rigen las tablas de la base de datos. Para esto se hace uso de los scripts desarrollados para tal fin.

La plataforma utiliza el comando **make** y *Makefiles* (archivos de scripts) para automatizar las diversas tareas.

Desde el repositorio principal, raíz del proyecto, correr:

```
$ make clean-db migrate
```

```
+ > make clean-db migrate
make -C scripts clean-db
make[1]: se entra en el directorio '/home/kiriost/dev/omics/scripts'
npm run clean-db

> api@1.0.0 clean-db /home/kiriost/dev/omics/scripts
> node scripts/clean-db

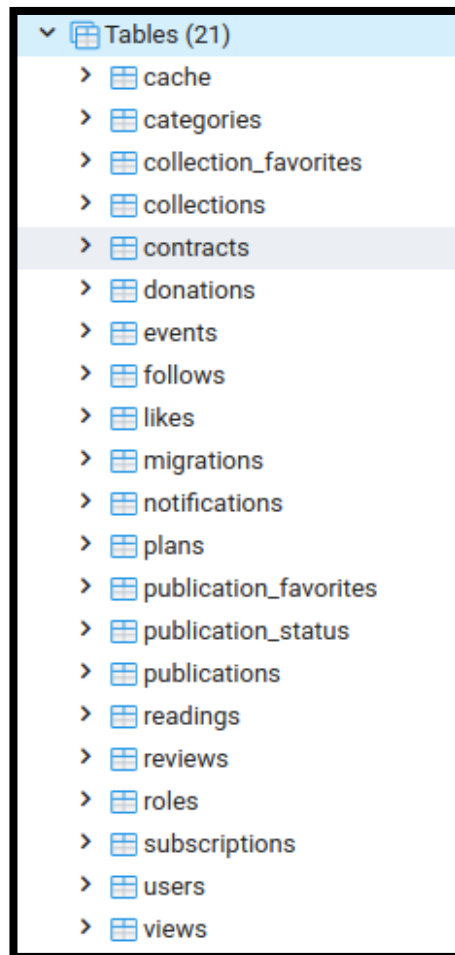
[ CLEAN DATABASE ]
Dropping tables...READY
make[1]: se sale del directorio '/home/kiriost/dev/omics/scripts'
make -C scripts migrate
make[1]: se entra en el directorio '/home/kiriost/dev/omics/scripts'
npm run migrate

> api@1.0.0 migrate /home/kiriost/dev/omics/scripts
> node scripts/migrate

[ MIGRATE ]
Migrating...
0000_init.sql: RUN
0001_populate.sql: RUN
READY
make[1]: se sale del directorio '/home/kiriost/dev/omics/scripts'
```

Anexo N°5 - Figura 1. 46 Ejecución del comando `make clean-db migrate`.

**make** permite correr más de un script al mismo tiempo y definir las dependencias entre ellos. Con lo anterior se ejecuta el script encargado de limpiar la base de datos y otro, más complejo, para correr las migraciones sobre la base de datos. Si volvemos a correr el script, las migraciones ya habrán impactado sobre la base de datos por lo que no volverán a ejecutarse.



Anexo N°5 - Figura 1. 47 Tablas de la base de datos de Omics

Así, obtenemos esta estructura de base de datos.

Cabe aclarar que **make migrate** además de generar los esquemas de las tablas también poblará la base de datos con los datos básicos iniciales: reglas de negocio, roles, usuario administrador, usuario gestor de contenido, categorías, planes básicos.

### 8.2.3. Datos de prueba

En caso de que se necesiten datos de prueba, generados al azar, con el propósito de probar la plataforma y la integración entre partes, puede ejecutarse:

```
$ make populate
```

Esto ejecutará un script desarrollado en NodeJS, encargado de poblar la base de datos con datos aleatorios, y, principalmente, consistentes.



```
> make populate
make -C scripts populate
make[1]: se entra en el directorio '/home/kiriost/dev/omics/scripts'
npm run populate

> api@1.0.0 populate /home/kiriost/dev/omics/scripts
> node scripts/populate

[ POPULATE ]
Populating DB...
READY
make[1]: se sale del directorio '/home/kiriost/dev/omics/scripts'
```

Anexo N°5 - Figura 1. 48 Población de tablas con datos aleatorios con comando *make populate*

### 8.3. Realización de backup

Estrategia para nombrar un archivo de backup

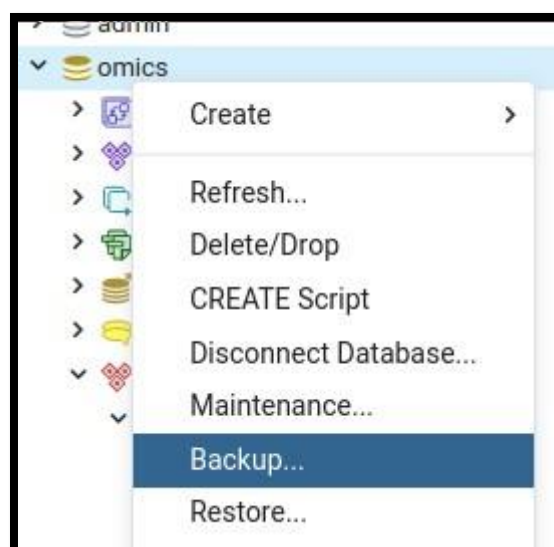
Es importante respetar la estandarización de nombres para generar archivos de backup.

Debe respetarse el siguiente formato: *YYYYMMDD\_backup.sql*.

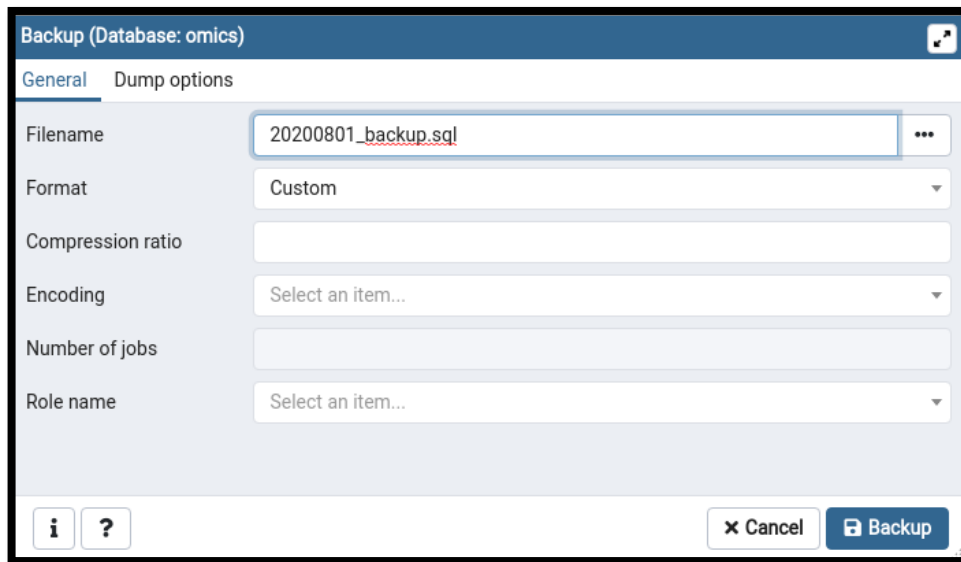
Siendo YYYY el año, MM el mes y DD el día actual. Seguido del sufijo "\_backup".

#### 8.3.1. Manual

De **forma gráfica**, dirigirse al panel de administración pgAdmin. Autenticarse. Hacer click derecho sobre la base de datos "omics" y seleccionar "Backup...".



Anexo N°5 - Figura 1. 49 Opción de Backup del Panel de administración de pgAdmin.



Anexo N°5 - Figura 1. 50 Realización de backup por Interfaz gráfica.

Es recomendable realizar backups de forma manual a través de la consola, ya que se hace uso de un script generado para tal fin, el cual genera automáticamente los nombres de archivo.

Para esto último, se ejecuta:

```
$ make backup
```

Dicho comando ejecutará el script localizado en "omics/scripts/scripts/backup.sh". Así, de forma automática, se conectará a la base de datos y hará un backup completo de la base de datos "omics".

### 8.3.2. Automático

Se genera una nueva entrada en **cron**, comando de GNU/Linux, en el servidor donde Omics se ejecute para producción.

Dicho comando únicamente debe correr **make backup** periódicamente. En caso de no estar disponible *make*, puede correr directamente el script que ejecuta:

```
$ ./omics/scripts/scripts/backup.sh
```

### 8.3.3. Volcado de SQL

La idea detrás de este método de volcado es generar un archivo de texto con comandos SQL que, cuando se realimenta al servidor, recreará la base de datos en el

mismo estado que tenía en el momento del volcado. PostgreSQL proporciona el programa de utilidad **pg\_dump** para este propósito.

### Por CLI (Cliente):

El uso básico de este comando es:

```
pg_dump -U usuario -W -h host basename > basename.sql  
pg_dump -U postgres -W -h localhost omics > 20201006_backup.sql
```

Se agrega la fecha del backup adelante del archivo sql de backup generado.

El comando anterior se automatiza mediante un proceso de cron para que corra todos los días a las 02:00 am.

### Explicación de los comandos:

- ❖ **U:** Se refiere al Usuario, en este caso puede ser el usuario propietario de la base de datos o el usuario postgres.
- ❖ **W:** Con este parámetro conseguiremos que nos solicite el password del usuario antes especificado.
- ❖ **h:** Con este indicamos cuál es el servidor PostgreSQL al que nos conectaremos para obtener nuestro dump, si estamos local podemos colocar localhost sino ponemos la IP del servidor PostgreSQL.

**basename:** Este es el último parámetro realmente en esta línea de comando, por esa razón no tiene alguna letra que indique que el siguiente parámetro es el nombre de la base de datos.

**basename.sql:** Esta parte en realidad solo indica que la salida de nuestro comando pg\_dump la guarde en un archivo basename.sql.

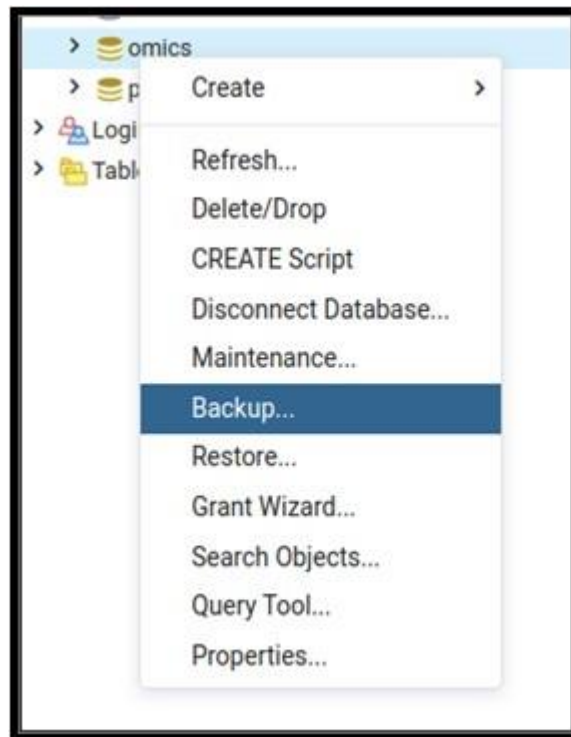
### Por GUI (Interfaz):

Se utiliza el panel de gestión de PostgreSQL pgAdmin4. Este dashboard es online, y se accede mediante el navegador web (editado).

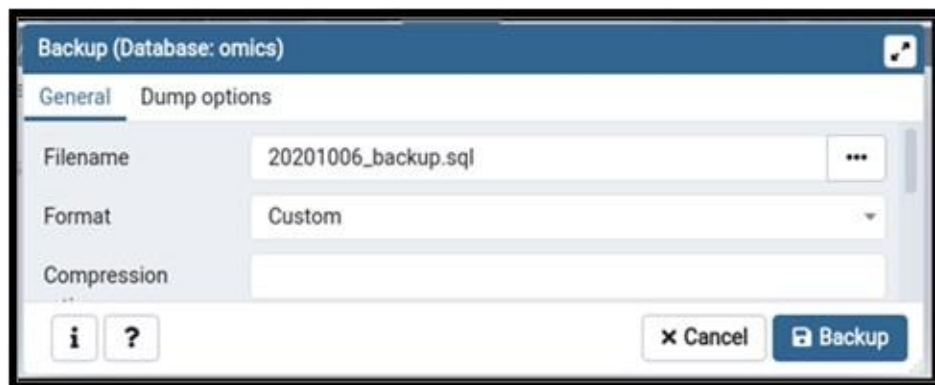
A través de la GUI se pueden realizar backup de la DB manualmente, como se observa en las siguientes figuras:



Anexo N°5 - Figura 1. 51 Interfaz gráfica de PgAdmin.



Anexo N°5 - Figura 1. 52 Realizar backup por PgAdmin.



Anexo N°5 - Figura 1. 53 Opciones de Backup.

### 8.3.4. Migraciones

Por cada nueva versión de la plataforma que se quiera poner en producción, e incluso en entornos de desarrollo y testing, debe ejecutarse:

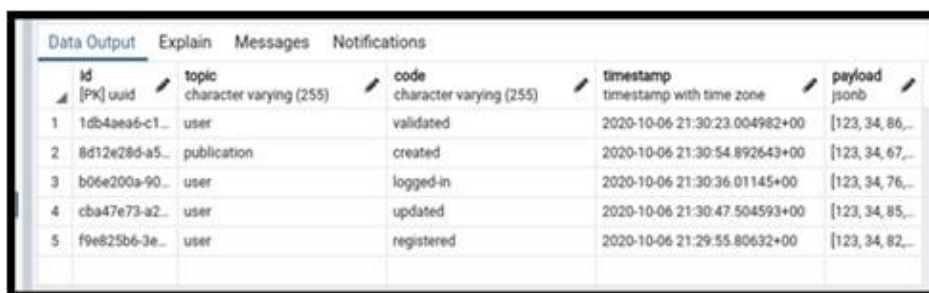
**\$ make migrate**

Lo que provocará que las nuevas migraciones, aún no impactadas en la base de datos sobre la que se trabaja, corran.

### 8.3.5. Eventos

Ya que la plataforma de Omics utiliza Event-Sourcing como metodología, donde cada cambio en las entidades emite un evento, y cada evento es almacenado en la DB a modo de event store, se puede realizar un backup únicamente de los eventos ocurridos.

De esta forma, si los eventos ocurridos desde el comienzo de la plataforma (corriendo en producción) se vuelven a aplicar a todos los modelos o entidades, se consigue el mismo estado en el que se encuentra actualmente.



id	topic	code	timestamp	payload
1db4ee6-c1...	user	validated	2020-10-06 21:30:23.004982+00	[123, 34, 86, ...]
8d12e28d-a5...	publication	created	2020-10-06 21:30:54.892643+00	[123, 34, 67, ...]
b06e200a-90...	user	logged-in	2020-10-06 21:30:36.01145+00	[123, 34, 76, ...]
cba47e73-a2...	user	updated	2020-10-06 21:30:47.504593+00	[123, 34, 85, ...]
f9e825b6-3e...	user	registered	2020-10-06 21:29:55.80632+00	[123, 34, 82, ...]

Anexo N°5 - Figura 1. 54 Backups realizados.

De esta forma, si se vuelve a correr el gestor de migraciones (un script especial desarrollado en JavaScript) sólo se ejecutarán y su resultado será reflejado sobre el esquema de DB aquellos que no hayan sido ya ejecutados.

### 8.3.6. Métodos de Replicación

El sistema de gestión de bases de datos elegido "PostgreSQL" posee un método de replicación.

Es la transmisión de información derivada de las operaciones DML (Data Manipulation Language) de una Base de Datos a otra, es decir, se transmite a otra Base de Datos las instrucciones INSERT, UPDATE y DELETE que se realicen en una Base de datos de forma que ambas BBDD tengan la misma información, así pues, se obtiene una redundancia de datos.

#### Replicación en PostgreSQL:

Uno de los métodos más usados para conseguir una alta disponibilidad en nuestro servidor Postgres es implementar la Replicación. Con ello nos aseguramos de que nuestra plataforma Omics, esté activa el 99,9% del año.

PostgreSQL dispone de varias herramientas y métodos para implementar:

#### Tipos:

- ❖ Basada en triggers
- ❖ Basada en ficheros WAL (Write Ahead Log).

### **Técnicas:**

- ❖ Archivado continuo (WAL Shipping).
- ❖ Streaming Replication.

### **Forma de transmisión:**

- ❖ Asíncrona.
- ❖ Síncrona.

### **Tipos de replicación:**

Un trigger (Disparador), es una acción que se ejecuta con cada operación DML. Utiliza una cola para almacenar los cambios y enviarla a la Base de Datos remota de forma asíncrona. Su ventaja, se puede seleccionar un subconjunto de una o varias tablas de las Bases de Datos.

Un fichero WAL es un registro con los INSERT, UPDATE y DELETES que se han realizado en una Base de Datos. Los ficheros WAL se generan para garantizar la durabilidad de la Base de Datos.

### **Técnicas de implementación:**

PostgreSQL puede enviar la información de los registros WAL mediante un archivado continuo (WAL Shipping) entre el servidor maestro y el servidor esclavo. También se puede configurar para que se envíen los ficheros WAL con las operaciones DML de una Base de Datos a otra mediante la red (Streaming Replication). El modo de envío de estos ficheros puede ser de forma asíncrona o síncrona.

### **BACKUP**

La base de dato PostgreSQL debe ser respaldada regularmente.

Existen tres enfoques fundamentales diferentes para hacer una copia de seguridad de los datos de PostgreSQL:

- ❖ Volcado de SQL.
- ❖ Copia de seguridad a nivel de sistema de archivos.
- ❖ Archivo continuo.

Cada uno tiene sus propias fortalezas y debilidades. Por el momento se va a utilizar el tipo "Volcado de SQL".

## 8.4. Recuperación

Siguiendo el mismo método del apartado anterior para la restauración se va a aplicar el siguiente comando:

```
psql -U username -W -h host basename < basename.sql  
psql -U postgres -W -h localhost omics < omicsBackup.sql
```

### **Las tareas para realizar la Recuperación son las siguientes:**

Se selecciona el backup más reciente, lo cual puede ser determinado gracias al nombre del archivo del backup generado con anterioridad, como, por ejemplo: "20200501\_backup.sql", el cual es el archivo de backup correspondiente al 01/05/2020. Antes de cargar el archivo, se limpia la base de datos completa y se ejecutan las migraciones para reconstruir los esquemas necesarios, como ya se comentó: "make clean-db migrate".

Una vez realizado esto, se accede al dashboard de administración de la base de datos principal, PostgreSQL. Y se carga el archivo de backup.

Para comprobar el correcto funcionamiento, existen archivos de script que pueblan la base de datos con datos "fake", es decir, datos de prueba. Para esto se ejecuta: "make populate".

Esto sirve para comprobar que tanto el back-end como la DB están manejando los mismos esquemas de datos. Debería realizarse antes de cargar el backup real para apaciguar dudas.

Para comprobar que se ha cargado correctamente el backup, se puede corroborar desde el dashboard de administración de la DB, pgAdmin. Además, para comprobar que funciona correctamente junto a la lógica del back-end se han creado tests de integración en forma de scripts. Estos tests realizan llamadas a la API del sistema y comprueban que las respuestas que se reciben son las esperadas. Se pueden ejecutar con "make api-test", y corren gracias a NodeJS y el framework de testing Jest.

La plataforma Omics se inserta en el mercado para prestar servicio en dos fases:

La primera tiene como objetivo la generación de la comunidad en torno a la plataforma. Una vez establecida una comunidad lo suficientemente grande, entra en juego la segunda fase que se encarga de asentar las estrategias de negocio y el modelo económico que se quiere adoptar para generar beneficios en la organización, pero manteniendo también los beneficios que representa para todo tipo de usuario de la misma.

Cada fase, a su vez, utiliza distintas estrategias de conversión e implementación, por lo cual la conversión resultante es una combinación de distintos métodos. También, se tiene en cuenta el uso interno del sistema, para el Administrador encargado del "Control de Contenido".

La implementación e inserción al mercado de *Omics*, son críticas para el éxito de la plataforma. Dicha implementación no es tradicional en el sentido de que no conlleva únicamente adaptar a los usuarios para mejorar su eficiencia, sino que también se aprovecha la retroalimentación de estos para mejorar la calidad del servicio brindado por la plataforma, en términos de usabilidad y performance.



## 9. Preguntas Frecuentes

### 9.1. OMICS

#### ❖ ¿Qué es OMICS?

Omics es una plataforma para distribuir contenido de alta calidad, acompañando a los autores que recién comienzan con sus primeros cómics y otros que ya poseen trayectoria, pero que no encuentran un lugar seguro para publicar sus obras.

Es el "Spotify" de los cómics donde pueden darse a conocer artistas que no tienen el alcance de grandes editoriales. Enfatiza su cercanía al usuario, el cual no tiene que invertir en papel, y puede seguir sus gustos según autor, categoría, estilo, etcétera.

#### ❖ ¿La plataforma de OMICS es gratuita?

Sí, es gratuita. Lee gratis los cómics originales exclusivos de OMICS. La plataforma se actualiza constantemente para que puedas leer publicaciones nuevas de tus autores favoritos.

#### ❖ ¿En qué dispositivos funciona la plataforma de OMICS?

Puedes utilizarla en un servidor web tanto en notebook, PC, como en tu celular. Puedes acceder al sitio móvil desde cualquier navegador. Es posible que el funcionamiento de la aplicación varíe según el dispositivo y la conexión que uses. Ten en cuenta que es posible que se apliquen cargos de datos según tu uso.

#### ❖ Tengo problemas con la conexión a la plataforma.

- Verifica tu conexión a internet: En ciertos casos es posible que las páginas de OMICS no se abran o inicien correctamente si tu conexión es inestable. En ese caso, te recomendamos ponerte en contacto directamente con tu ISP (proveedor de servicio de internet).
- Intenta refrescar la página: Presiona la tecla F5 o actualizar en la pestaña superior a la izquierda del navegador por si se ha colgado o las imágenes no cargan.
- Otro: Si a pesar de haber comprobado lo que se menciona anteriormente, el problema persiste, ponte en contacto con el servicio de OMICS para que podamos analizarlo.

### 9.2. Cuentas e inicio de sesión

#### ❖ ¿Cómo puedo reestablecer mi contraseña?

Si te registraste con una dirección de email debes hacer click en ¿No recuerdas la contraseña? En la pantalla de inicio de sesión. Ingresar la dirección de email, haz click en "Enviar" y seguir las instrucciones del correo recibido.

❖ **Apareció un mensaje que me pedía que confirmara mi dirección de email y contraseña.**

Verifica que hayas registrado tu cuenta en OMICS con tu dirección de email. La información de tu cuenta de OMICS distingue entre mayúsculas y minúsculas. Los caracteres de tu nombre de usuario y contraseña deben coincidir exactamente con los de tu cuenta para poder iniciar sesión.

❖ **Quiero eliminar mi cuenta.**

Si quieres eliminar tu cuenta es necesario que lo hagas desde la página web de OMICS. Sigue estos pasos desde el navegador de tu PC o teléfono inteligente:

1. Iniciar sesión en OMICS y dirigirse a la imagen/foto de usuario. Allí se despliegan las funcionalidades.
2. Seleccionar la sección de "Ver Perfil".
3. Seleccionar la opción "Editar Perfil".
4. Luego, seleccionar la opción "Eliminar usuario". Por seguridad te pide que ingreses la contraseña para confirmar la acción que deseas realizar.

❖ **No recuerdo mi cuenta de OMICS.**

Comunícate con el equipo de OMICS para que podamos revisar tu cuenta a [OMICS.CONTACTO@GMAIL.COM](mailto:OMICS.CONTACTO@GMAIL.COM).

❖ **Aún no he recibido el correo de recuperación de contraseña.**

Es posible que hayas recibido el correo de recuperación de contraseña en tu carpeta de spam. Si no está allí, accede a los ajustes de tu email y verifica que el contacto de [OMICS.CONTACTO@GMAIL.COM](mailto:OMICS.CONTACTO@GMAIL.COM) no esté en la lista de remitentes bloqueados.

Si los problemas persisten, comunícate con el equipo de OMICS para que podamos resolver el problema a [OMICS.CONTACTO@GMAIL.COM](mailto:OMICS.CONTACTO@GMAIL.COM).

❖ **¿Tengo que crear una cuenta con mi dirección de email?**

Sí, tienes que crear una cuenta con tu dirección de email para comenzar a utilizar todas las funcionalidades que te ofrece la plataforma de OMICS.

### 9.3. Uso

❖ **Mis "Favoritos" desaparecieron.**

Revisa que no hayas cerrado sesión en tu cuenta de OMICS. Puedes acceder al historial de lectura desde "Favoritos" una vez que inicies sesión en tu cuenta.

❖ **La plataforma se cierra constantemente después de una actualización.**

Comunícate con el equipo de OMICS para que podamos resolver el problema a [OMICS.CONTACTO@GMAIL.COM](mailto:OMICS.CONTACTO@GMAIL.COM).

❖ **¿Cómo me suscribo para poder acceder a contenido premium?**

La Suscripción a la plataforma es mensual, y se puede cancelar en cualquier momento que el usuario así lo desee. Cuando un usuario quiere suscribirse a la plataforma para poder acceder al contenido premium (Publicaciones marcadas con una Estrella), puede realizarlo de 3 formas diferentes:

1. Si el usuario selecciona una Publicación Premium (marcada con una Estrella) se despliega la pantalla con la opción para Suscribirse.
2. El usuario debe ir a la esquina superior derecha y seleccionar su perfil. Se despliega el menú del panel de usuario y se selecciona la opción "Suscribirse". Ahí se observa el Plan Básico, el valor y el botón Suscribirse. Al seleccionar Suscribirse se visualiza la opción Pagar. Al seleccionarlo, el usuario es redirigido a la sección de pago check-out de Mercado Pago.
3. El usuario debe ir a la esquina superior derecha y seleccionar su perfil. Se despliega el menú del panel de usuario y se selecciona la opción "Editar Perfil". Ahí se encuentra la opción de "Suscribirse", se selecciona y se despliega la pantalla con el Plan Básico, el valor y el botón Suscribirse. Al confirmar, se visualiza la opción Pagar. Al seleccionarlo, el usuario es redirigido a la sección de pago check-out de Mercado Pago.

❖ **No puedo conectarme a OMICS.**

Si tienes problemas para conectarte a OMICS, intenta comprobar la configuración de red de tu dispositivo.

Comprueba la opción que se adecúe a tu caso si utilizas teléfono móvil:

1. Usuarios de Android
  - a) Ve a la sección de ajustes de tu dispositivo.
  - b) Pulsa Conexiones > Redes móviles > Nombres de punto de acceso.
  - c) Abre el menú en la parte superior de la pantalla, y selecciona Restablecer valores. (Los nombres y ubicaciones mostradas varían según el dispositivo.)
2. Usuarios de iOS
  - a) Ve a la sección de ajustes de tu dispositivo.
  - b) Pulsa General > Restablecer.
  - c) Pulsa Restablecer la configuración de red.

Ahora, si a pesar de haber comprobado lo mencionado anteriormente, el problema persiste, ponte en contacto con el equipo de OMICS para que podamos resolver el problema a [OMICS.CONTACTO@GMAIL.COM](mailto:OMICS.CONTACTO@GMAIL.COM).

**❖ La plataforma no carga los cómics que quiero leer.**

Revisa tu conexión a internet. Si tu red es inestable, es posible que la plataforma no pueda cargar las imágenes correctamente. Si los problemas persisten, comunícate con el equipo de OMICS para que podamos resolver el problema a [OMICS.CONTACTO@GMAIL.COM](mailto:OMICS.CONTACTO@GMAIL.COM).

**❖ La plataforma quedó trabada en una pantalla blanca apenas la abrí.**

Revisa tu conexión a internet. Si tu conexión es inestable, es posible que la plataforma no cargue el contenido correctamente. Si el problema no se resuelve, comunícate con el equipo de OMICS para que podamos resolver el problema a [OMICS.CONTACTO@GMAIL.COM](mailto:OMICS.CONTACTO@GMAIL.COM).

## 9.4. ¿Cómo publicar en OMICS?

**❖ ¿Cómo empiezo a publicar?**

El botón "Publicar" se encuentra en el sitio de OMICS. Sigue las siguientes recomendaciones:

1. Ir a la esquina superior izquierda, al Menú de despliegue (tres líneas horizontales) y seleccionar "Dashboard".
2. Luego se le muestra la ventana con los botones de:
  - a) Publicaciones.
  - b) Colecciones.
3. Debe seleccionar el botón Publicaciones y se le muestra el botón "Añadir Publicación".
4. Al seleccionar dicho botón se le despliega la ventana para llenar los datos de la Publicación que desea subir a la plataforma.
5. Una vez cargados los datos debe seleccionar el botón Publicar (botón verde). Luego pasa a un control de contenido y una vez aprobado su cómic será publicado con éxito.

**❖ ¿Cómo guardo un borrador?**

Cuando haces una carga de una obra, puedes guardar la misma como borrador con la opción "Borrador" o publicarlo para que luego de un control de contenido se apruebe y comience a compartirse en la plataforma. Guardarlo como borrador cargará el cómic en el servidor de OMICS, pero tus lectores no podrán verlo.

**❖ ¿Qué edad debo tener para ingresar a la plataforma?**

Debes tener por lo menos 14 años para ingresar a la plataforma de OMICS y usarlo.

**❖ ¿Qué edad debo tener para publicar?**

Debes tener por lo menos 14 años para publicar en la plataforma de OMICS.

❖ **¿Qué proceso puedo seguir si me rechazaron la publicación?**

Cuando el Gestor de Contenido rechaza tu publicación, te llega una notificación y un email con el/los motivos. Puedes tomar esas recomendaciones y hacer los cambios necesarios en tu obra para que luego pueda ser publicada en la plataforma. Una vez realizado los cambios, tu cómic pasa a revisión nuevamente y en caso de que esté todo bien, tu publicación es aprobada.

❖ **Otra persona está publicando mis cómics en OMICS sin mi permiso.**

Comunícate con el equipo de OMICS para recibir ayuda y que podamos resolver el problema a [OMICS.CONTACTO@GMAIL.COM](mailto:OMICS.CONTACTO@GMAIL.COM) y bríndale la siguiente información:

1. Nombre del cómic infractor.
2. URL del cómic infractor.
3. Prueba de propiedad de contenido (como un enlace a la publicación original o el archivo PSD). Revisaremos la denuncia y tomaremos las medidas pertinentes.

❖ **Mis archivos no se están cargando.**

Si tienes problemas para conectarte a OMICS, prueba restablecer los ajustes de red.

Si los problemas persisten después de restablecer tus ajustes de red, comunícate con el equipo de OMICS para que podamos resolver el problema a [OMICS.CONTACTO@GMAIL.COM](mailto:OMICS.CONTACTO@GMAIL.COM) y bríndale la siguiente información:

1. Captura de pantalla de la página de error.
2. Cuenta con la que iniciaste sesión.
3. País.
4. Dispositivo (por ejemplo, Samsung Galaxy S8+, iPhone X, iPad 9.7, Kindle Fire HD8).
5. Sistema operativo y versión (por ejemplo, Android 8.0.0, iOS 11.3.1).
6. Versión de la plataforma de OMICS (Configuración> Información> Versión de la plataforma).
7. Conexión a internet.

## 9.5. Otros

❖ **Quiero reportar un plagio.**

En ese caso, por favor ponte en contacto con el equipo de OMICS para que podamos resolver el problema a [OMICS.CONTACTO@GMAIL.COM](mailto:OMICS.CONTACTO@GMAIL.COM) e incluye la siguiente información:

- Título del cómic infractor.
- URL del cómic infractor.

- Pruebas de propiedad de contenido. (URL de la publicación original o el archivo PSD original del contenido en cuestión).
- ❖ **¿Dónde puedo enviar comentarios/sugerencias relacionados con OMICS?**
  - ¡Tus sugerencias, comentarios e ideas siempre son bienvenidas!
  - Comunícate con el equipo de OMICS [OMICS.CONTACTO@GMAIL.COM](mailto:OMICS.CONTACTO@GMAIL.COM) para compartir tus comentarios o sugerencias.

## 10. Videos Tutoriales

Un tutorial es un método para transferir conocimiento, que conduce al usuario a través de las características y funcionalidades más importantes del Sistema.

En general, los videotutoriales son documentos que pretenden guiar el proceso que debe seguir una persona para realizar una o varias actividades en un campo determinado, con una amplitud muy variada y directamente relacionada con el nivel de profundización y de especificación del proceso que se pretende guiar.

Se pretende usar esta metodología para la guía de los usuarios, autores y lectores, en la que la manera de aprendizaje sea sencilla e interactiva. Se explican paso a paso, en un video corto, cómo se deben realizar tareas concretas. Pueden verse todas las veces que sea necesario y detenerse para continuar después cuando sea necesario.

En la sección que sigue se dejan los links de cada video tutorial con su correspondiente explicación, que se encuentran en el canal del Equipo Omics en YouTube.

### **10.1. Agregar a favoritos y like**

<HTTPS://YOUTU.BE/V7MFA-3KC9U>

### **10.2. Aprobar publicación**

<HTTPS://YOUTU.BE/J2GFYMM2QQG>

### **10.3. Cambiar contraseña**

<HTTPS://YOUTU.BE/L3DRUGNKAUM>

### **10.4. Corregir publicación rechazada**

[HTTPS://YOUTU.BE/\\_OAD6V9JYNA](HTTPS://YOUTU.BE/_OAD6V9JYNA)

### **10.5. Crear comentario, ver perfil y seguir autor**

[HTTPS://YOUTU.BE/\\_AZz4HMEJVQ](HTTPS://YOUTU.BE/_AZz4HMEJVQ)

### **10.6. Crear nueva publicación**

<HTTPS://YOUTU.BE/IBVL97ODPR0>

### **10.7. Editar perfil**

<HTTPS://YOUTU.BE/2RNRVOCNzCQ>

## **10.8. Eliminar colección**

[HTTPS://YOUTU.BE/SXNICAXu2Sk](https://youtu.be/SXNICAXu2Sk)

## **10.9. Eliminar publicación**

[HTTPS://YOUTU.BE/DCV\\_TLQ-6IG](https://youtu.be/DCV_TLQ-6IG)

## **10.10. Rechazar publicación**

[HTTPS://YOUTU.BE/AVCGNULZTK8](https://youtu.be/AVCGNULZTK8)

## **10.11. Recuperar contraseña**

[HTTPS://YOUTU.BE/6GAW5H0OZAA](https://youtu.be/6GAW5H0OZAA)

## **10.12. Registrarse**

[HTTPS://YOUTU.BE/OFSFG2AI5S0](https://youtu.be/OFSFG2AI5S0)

## **10.13. Solicitar y aprobar contrato**

[HTTPS://YOUTU.BE/V3P\\_6OfEWGM](https://youtu.be/V3P_6OfEWGM)

## **10.14. Suscribir y desuscribir**

[HTTPS://YOUTU.BE/VOJ61VDCR7A](https://youtu.be/VOJ61VDCR7A)

## **10.15. Ver notificaciones y crear colección**

[HTTPS://YOUTU.BE/I0VBRXODGYI](https://youtu.be/I0VBRXODGYI)

## **10.16. Visualizar publicación aprobada**

[HTTPS://YOUTU.BE/WLHKU8B9NU4](https://youtu.be/WLHKU8B9NU4)



## Bibliografía

- (1) *¿Qué son la integración/distribución continuas (CI/CD)?* (s/f). Recuperado el 27 de octubre de 2020, de <https://www.redhat.com/es/topics/devops/what-is-ci-cd>
- (2) Diferencias.eu. (2019). *√ Diferencia entre comic e historieta | [SOLUCIÓN]*. <https://diferencias.eu/entre-comic-e-historieta/>
- (3) *Ejemplo de uso del tablero o pizarra de tareas (Scrum Taskboard) – Proyectos Ágiles.* (s/f). Recuperado el 27 de octubre de 2020, de <https://proyectosagiles.org/2010/09/26/ejemplo-tablero-pizarra-tareas-scrum-taskboard/>
- (4) Española, R. A. (2019a). *cómic | Definición | Diccionario de la lengua española | RAE - ASALE.* <https://dle.rae.es/cómic>
- (5) Española, R. A. (2019b). *yuxtaposición | Definición | Diccionario de la lengua española | RAE - ASALE.* <https://dle.rae.es/yuxtaposición?m=form>
- (6) Fundación Wikipedia, I. (s/f). *Diseño guiado por el dominio - Wikipedia, la enciclopedia libre.* Recuperado el 27 de octubre de 2020, de [https://es.wikipedia.org/wiki/Diseño\\_guiado\\_por\\_el\\_dominio](https://es.wikipedia.org/wiki/Diseño_guiado_por_el_dominio)
- (7) Fundación Wikipedia, I. (2017). *Toba-e - Wikipedia.* <https://en.wikipedia.org/wiki/Toba-e>
- (8) Fundación Wikipedia, I. (2018). *Tankōbon: Revision history - Wikipedia.* <https://en.wikipedia.org/w/index.php?title=Tankōbon&action=history>
- (9) Fundación Wikipedia, I. (2019). *Historial de «Kanji» - Wikipedia, la enciclopedia libre.* <https://es.wikipedia.org/w/index.php?title=Kanji&action=history>
- (10) Fundación Wikipedia, I. (2020a). *Comics - Wikipedia.* <https://en.wikipedia.org/wiki/Comics>
- (11) Fundación Wikipedia, I. (2020b). *Desarrollo ágil de software - Wikipedia, la enciclopedia libre.* [https://es.wikipedia.org/wiki/Desarrollo\\_ágil\\_de\\_software](https://es.wikipedia.org/wiki/Desarrollo_ágil_de_software)
- (12) Fundación Wikipedia, I. (2020c). *History of manga - Wikipedia.* [https://en.wikipedia.org/wiki/History\\_of\\_manga](https://en.wikipedia.org/wiki/History_of_manga)
- (13) Fundación Wikipedia, I. (2020d). *Kanban - Wikipedia, la enciclopedia libre.* <https://es.wikipedia.org/wiki/Kanban>
- (14) Fundación Wikipedia, I. (2020e). *Manga - Wikipedia, la enciclopedia libre.* <https://es.wikipedia.org/wiki/Manga>
- (15) Fundación Wikipedia, I. (2020f). *OS-tan - Wikipedia.* [https://en.wikipedia.org/wiki/OS-tan#Windows\\_10](https://en.wikipedia.org/wiki/OS-tan#Windows_10)
- (16) Fundación Wikipedia, I. (2020g). *Richard F. Outcault - Wikipedia.* [https://en.wikipedia.org/wiki/Richard\\_F.\\_Outcault](https://en.wikipedia.org/wiki/Richard_F._Outcault)
- (17) Fundación Wikipedia, I. (2020h). *Rodolphe Töpffer - Wikipedia.* [https://en.wikipedia.org/wiki/Rodolphe\\_Töpffer](https://en.wikipedia.org/wiki/Rodolphe_Töpffer)
- (18) Fundación Wikipedia, I. (2020i). *The Yellow Kid - Wikipedia.* [https://en.wikipedia.org/wiki/The\\_Yellow\\_Kid](https://en.wikipedia.org/wiki/The_Yellow_Kid)
- (19) Fundación Wikipedia, I. (2020j). *Webcomic - Wikipedia.* <https://en.wikipedia.org/wiki/Webcomic>
- (20) Giuanta, N. G. (s/f). *LA HISTORIA DEL COMIC EN ARGENTINA.* Recuperado el 26

- de octubre de 2020, de [https://www.todohistorietas.com.ar/historia\\_argentina\\_1.htm#TOP](https://www.todohistorietas.com.ar/historia_argentina_1.htm#TOP)
- (21)History of Stuff. (2017, abril 27). *History of Comic Books - YouTube*. <https://www.youtube.com/watch?v=qipZ6h4S--c&feature=youtu.be>
- (22)*Ley 11723 del Régimen Legal de La Propiedad Intelectual*. (2009). <http://servicios.infoleg.gob.ar/infolegInternet/anexos/40000-44999/42755/texact.htm>
- (23)*Ley 25.326 Protección de los Datos*. (2008). <http://servicios.infoleg.gob.ar/infolegInternet/anexos/60000-64999/64790/texact.htm>
- (24)*Manga Toon - Lee cómics en español*. (s/f). Recuperado el 20 de octubre de 2020, de <https://mangatoon.mobi/es>
- (25)McCloud, S. (1993). *scottmcloud.com - Understanding Comics*. <http://www.scottmcloud.com/2-print/1-uc/index.html>
- (26)Palacio, J. (2015). *Scrum Manager I Las reglas de scrum*. <http://www.streetsofdublin.com/>
- (27)*Read Popular Manga Online - Crunchyroll*. (s/f). Recuperado el 27 de octubre de 2020, de <https://www.crunchyroll.com/comics/manga>
- (28)ROOTINKS. (s/f). Recuperado el 20 de octubre de 2020, de <http://www.rootinks.com/>
- (29)Screen, T. L. B. (2015). *History Of - Astro Boy! - YouTube*. <https://www.youtube.com/watch?v=f6tkNCMhUyY&feature=youtu.be>
- (30)*Sutori: Diferencia entre "Cómico" y "Manga"*. (2012). Sutori Manga, grupo Jesulink. <http://www.sutorimanga.com/2012/07/diferencia-entre-comic-y-manga.html>
- (31)Swidlicka, Z. (s/f). *Astro Boy and the legacy of Osamu Tezuka*. Recuperado el 26 de octubre de 2020, de <https://www.foldmagazine.com/astro-boy-legacy-osamu-tezuka>
- (32)*Tapas: Bite-sized stories you can't find anywhere else*. (s/f). Recuperado el 27 de octubre de 2020, de <https://tapas.io/>
- (33)Topping, S. (2016). *Biography of Joseph Pulitzer - The Pulitzer Prizes*. <https://www.pulitzer.org/page/biography-joseph-pulitzer>
- (34)Valvasuata. (2014). *Diferencia entre novela gráfica y cómic*. <http://diferenciaentre.info/diferencia-entre-novela-grafica-y-comic/>
- (35)*WEBTOON - Cómics en línea*. (s/f). Recuperado el 27 de octubre de 2020, de <https://www.webtoons.com/es/>

# Índice de Figuras y Diagramas

## Figuras

Figura I. Portada del influyente libro de estudios "Understanding Comics: The invisible Art" by Scott McCloud	17
Figura II. Una de las primeras ilustraciones realizados por Rodolphe Töpffer	18
Figura III. The Yellow Kid's R-R-Revenge; Or, How the Painter's Son Got Fresh (09/01/1898) por Richard F. Outcault	19
Figura IV. (izquierda) Una de las ilustraciones en el libro Toba Ehon (derecha) Una página ilustrativa del primer número de Astroboy.	19
Figura V. (izquierda) Ilustración de César Bacle (derecha) Portada de una de las historietas de Las Andanzas de Patoruzú.	20
Figura VI. Portada famosa del primer cómic de "Action Comics" donde aparece Superman.	21
Figura VII. Dilbert, historieta conocida en el mundo de la informática y sistemas.	22
Figura VIII. Watchmen, novela gráfica famosa que revolucionó la industria del cómic.	23
Figura IX. Es común encontrar guías de cómo se lee un manga cuando son traducidos y exportados a Occidente.	23
Figura X. Es tan fuerte la cultura manga en oriente que incluso los utilizan para presentar productos, guías para aprender a programar y lógica de las computadoras.	24
Figura 1. 1 Publicación y remuneración publicando en Mangatoon	28
Figura 1. 2 Contrato de obras populares.	29
Figura 1. 3 Respuesta del servidor de Mangatoon.	32
Figura 1. 4 Carga de imágenes secuenciales en lector de Mangatoon.	32
Figura 1. 5 Opciones de registro y login en Mangatoon. Panel de gestión de cuenta.	34
Figura 1. 6 Pantalla de inicio de Mangatoon.	35
Figura 1. 7 Catálogo en dispositivos móviles (izquierda) y menú de navegación (derecha).	36
Figura 1. 8 Catálogo de Mangatoon.	36
Figura 1. 9 Detalle de un cómic.	37
Figura 1. 10 Lector de cómics en Mangatoon.	38
Figura 1. 11 Destaca botón para publicar obra en Mangatoon.	38
Figura 1. 12 Publicación de obras en Mangatoon.	39
Figura 1. 13 Datos de obra a publicar en Mangatoon.	39
Figura 1. 14 Subida de imágenes que representan las viñetas en Mangatoon.	40
Figura 1. 15 Tamaño máximo de imagen a subir en Mangatoon.	40
Figura 1. 16 Beneficios del contrato en Mangatoon.	41

Figura 1. 17 Publicar en Tapas.	45
Figura 1. 18 Monetizar publicaciones en Tapas.	46
Figura 1. 19 Perfil de usuario en Tapas.	49
Figura 1. 20 Autenticación (ingreso y registro) en Tapas.	50
Figura 1. 21 Confirmación de email luego de registro en Tapas.	50
Figura 1. 22 Recuperación de contraseña mediante dirección de correo electrónico en Tapas.	51
Figura 1. 23 Cambio de contraseña.	52
Figura 1. 24 Pantalla inicial de Tapas.	52
Figura 1. 25 Catálogo de Tapas.	53
Figura 1. 26 Catálogo de Tapas	53
Figura 1. 27 Detalle de obra publicada.	54
Figura 1. 28 Lector digital de Tapas.	55
Figura 1. 29 Publicar en Tapas.	55
Figura 1. 30 Publicar nueva obra en Tapas.	56
Figura 1. 31 Comprar tinta en Tapas para realizar donaciones.	56
Figura 1. 32 Marcadores en Tapas.	57
Figura 1. 33 Notificaciones en Tapas.	57
Figura 1. 34 Recomendación por día de la semana en Webtoon.	62
Figura 1. 35 Detalle de una obra en Webtoon, por episodios.	62
Figura 1. 36 Información del creador en Webtoon.	63
Figura 1. 37 Barra de navegación en lector de Webtoon.	63
Figura 1. 38 Siguiendo en Webtoon.	64
Figura 1. 39 Calificación de cómic en Webtoon.	64
Figura 1. 40 Registro de usuario en Webtoon.	65
Figura 1. 41 Recuperación de contraseña en Webtoon.	65
Figura 1. 42 Logo de Crunchyroll.	67
Figura 1. 43 Inicio de Crunchyroll Manga. Izquierda: Vista en forma cuadrículada. Derecha: Vista en forma lista.	71
Figura 1. 44 Filtros de búsqueda por coincidencia de texto de Crunchyroll.	72
Figura 1. 45 Filtros de diferentes formas de mostrar el catálogo en Crunchyroll.	73
Figura 1. 46 Progreso en lecturas en Crunchyroll.	74
Figura 1. 47 La forma de leerse los mangas en Crunchyroll: Completa y siguiendo una guía que resalta la viñeta, una por una.	75
Figura 1. 48 (Izquierda) Lista de páginas marcadas. (Derecha) Lista de favoritos.	75
Figura 1. 49 Todas las opciones e información que aparece en la pantalla Configuraciones.	76
Figura 1. 50 Pantalla de bienvenida a la página web Rootinks.	81
Figura 1. 51 Pantalla de login de usuarios en Rootinks.	84

Figura 1. 52 Usuario registrado en Rootinks.	85
Figura 1. 53 Pantalla de cómics y categorías.	85
Figura 1. 54 Pantalla que muestra la información de las publicaciones del autor en Rootinks.	86
Figura 1. 55 Pantalla de marcadores, biblioteca del lector.	87
Figura 1. 56 Pantalla de visualización de cómics en Rootinks.	88
Figura 1. 57 Pantalla de error de logueo con Facebook.	88
Figura 1. 58 Pantalla N° 1: Pantalla de Bienvenida a Omics.	159
Figura 1. 59 Pantalla N° 2: Pantalla de Inicio Con Barra Lateral. Sin loguearse.	160
Figura 1. 60 Pantalla N° 3: Pantalla de Inicio Sin Barra Lateral. Sin loguearse.	160
Figura 1. 61 Pantalla N° 4: Pantalla para Registrarse.	161
Figura 1. 62 Pantalla N° 5: Pantalla de Login.	161
Figura 1. 63 Pantalla N° 6: Pantalla de Enviar Recuperar Contraseña.	162
Figura 1. 64 Pantalla N° 7: Pantalla de Nueva Contraseña.	162
Figura 1. 65 Pantalla N° 8: Pantalla de Usuarios Logueados.	163
Figura 1. 66 Pantalla N° 9: Pantalla de Permisos de Usuario, con menú desplegable (lo de fondo violeta es los de acceso de administradores, y lo de azul para usuarios lectores y/o autores).	163
Figura 1. 67 Pantalla N° 10: Pantalla de Editar Perfil.	164
Figura 1. 68 Pantalla N° 11: Pantalla de Expandir Opciones de Perfil.	164
Figura 1. 69 Pantalla N° 12: Pantalla de Ver Perfil.	165
Figura 1. 70 Pantalla N° 13: Pantalla de Gestionar Medio de Pago.	165
Figura 1. 71 Pantalla N° 14: Pantalla de Biblioteca General de Obras.	166
Figura 1. 72 Pantalla N° 15: Pantalla de Colecciones.	166
Figura 1. 73 Pantalla N° 16: Pantalla de Ver Detalle de Colección.	167
Figura 1. 74 Pantalla N° 17: Pantalla de Ver Detalle de Publicación.	167
Figura 1. 75 Pantalla N° 18: Pantalla de Ver Favoritos Publicaciones.	168
Figura 1. 76 Pantalla N° 19: Pantalla de Lector de Cómics.	168
Figura 1. 77 Pantalla N° 20: Pantalla de Realizar Reseña.	169
Figura 1. 78 Pantalla N° 21: Pantalla de Donar a Autor.	169
Figura 1. 79 Pantalla N° 22: Pantalla de Suscripción a la plataforma.	170
Figura 1. 80 Pantalla N° 23: Pantalla de Cancelar Suscripción.	170
Figura 1. 81 Pantalla N° 24: Pantalla de Introducir Pago.	171
Figura 1. 82 Pantalla N° 25: Pantalla de Solicitar Publicar Obra.	171
Figura 1. 83 Pantalla N° 26: Pantalla de Publicación Espera Aprobación.	172
Figura 1. 84 Pantalla N° 27: Pantalla de Editar Publicación.	172
Figura 1. 85 Pantalla N° 28: Pantalla de Gestionar Colecciones.	173
Figura 1. 86 Pantalla N° 29: Pantalla de Crear Colecciones.	173
Figura 1. 87 Pantalla N° 30: Pantalla de Editar Colecciones.	174

Figura 1. 88 Pantalla N° 31: Pantalla de Aceptar/Rechazar Publicaciones.	174
Figura 1. 89 Pantalla N° 32: Pantalla de Ver Estadísticas de Publicación.	175
Figura 1. 90 Pantalla N° 33: Pantalla de primera publicación aprobada.	175
Figura 1. 91 Pantalla N° 34: Pantalla de Notificaciones.	176
Figura 1. 92 Pantalla N° 35: Pantalla de Publicación Sin Contrato.	177
Figura 1. 93 Pantalla N° 36: Pantalla de Mis Comics.	177
Figura 1. 94 Pantalla N° 37: Pantalla de Comentarios.	177
Figura 1. 95 Pantalla N° 38: Pantalla de Publicación Con Contrato.	178
Figura 1. 96 Pantalla N° 39: Pantalla de Billetera Autor.	178
Figura 1. 97 Pantalla N° 40: Pantalla de Publicación Ver Resúmenes Generados.	179
Figura 1. 98 Pantalla N° 41: Pantalla de Aprobar/Rechazar Contrato.	179
Figura 1. 99 Pantalla N° 42: Pantalla de Reportes (a).	180
Figura 1. 100 Pantalla N° 43: Pantalla de Reportes (b).	180
Figura 1. 101 Pantalla N° 44: Gestionar Reglas de Negocios.	181
Figura 1. 102 Pantalla N° 45: Pantalla de Administrador de Gestionar Categorías.	181
Figura 1. 103 Pantalla N° 46: Gestionar Roles y Visualizar Tabla de Roles Disponibles.	182
Figura 1. 104 Pantalla N° 47: Crear Nuevo Rol.	182
Figura 1. 105 Pantalla N° 48: Editar Rol Existente.	183
Figura 1. 106 Pantalla N° 49: Gestionar Usuarios y sus Roles Asignados.	184
Figura 1. 107 Pantalla N° 50: Cambiar Rol a un Usuario Existente.	185
Figura 1. 108 Arquitectura limpia (Clean Architecture).	263
Figura 1. 109 Esquema DDD.	264
Figura 1. 110 Ejemplo de intercambio de mensaje de Notificaciones.	265
Figura 1. 111 Ejemplo de intercambio de mensaje de suscripción.	266
Figura 1. 112 Test del Núcleo del sistema	279
Figura 1. 113 Test de Identidad: autenticación, autorización, usuario y roles.	280
Figura 1. 114 Test Utilitarios para los controladores de la API REST	280
Figura 1. 115 Test de pagos	281
Figura 1. 116 Test Publicaciones, colecciones, autores, lectores y categorías.	281
Figura 1. 117 Página de Facebook de Omics.	323
Figura 1. 118 Interfaz gráfica de PGAdmin de las tablas que componen la Base de Datos de Omics.	325
Figura 1. 119 Ingreso a localhost:8080.	328
Figura 1. 120 Autenticación del administrador. Estos datos pueden variar por razones de seguridad.	328
Figura 1. 121 Tablero de administración de PostgreSQL mediante pgAdmin.	329
Figura 1. 122 Creación de la conexión con el motor de base de datos.	329

Figura 1. 123 Ingreso de dirección IP donde se encuentra el SGBD corriendo. Puerto por defecto.	330
Figura 1. 124 Ingreso de datos de credenciales para acceso a base de datos.	331
Figura 1. 125 Creamos la base de datos.	331
Figura 1. 126 Base de datos de nombre "omics". Es importante respetar el nombre.	332
Figura 1. 127 Ejecución del comando make clean-db migrate.	333
Figura 1. 128 Tablas de la base de datos de Omics	334
Figura 1. 129 Población de tablas con datos aleatorios con comando make populate.	335
Figura 1. 130 Opción de Backup del Panel de administración de pgAdmin.	335
Figura 1. 131 Realización de backup por Interfaz gráfica.	336
Figura 1. 132 Interfaz gráfica de PgAdmin.	337
Figura 1. 133 Realizar backup por PgAdmin.	338
Figura 1. 134 Opciones de Backup.	338
Figura 1. 135 Backups realizados.	339
Figura 1. 136 Pantalla del Administrador Opción Backups.	342
Figura 1. 137 Pantalla ventana de Backups.	343
Figura 2. 1 Planificación a partir de la definición del alcance de proyecto.	348
Figura 2. 2 Lista de Planificación.	362
Figura 2. 3 Lista de Definición de Requerimientos.	363
Figura 2. 4 Lista de Diseño.	363
Figura 2. 5 Lista de Desarrollo (a).	364
Figura 2. 6 Lista de Desarrollo (b).	365
Figura 2. 7 Lista de Desarrollo – Revisión de módulos del sistema (c).	366
Figura 2. 8 Lista de Desarrollo – Ejecución de pruebas unitarias (d).	366
Figura 2. 9 Lista de Desarrollo – Ejecución de pruebas (e).	367
Figura 2. 10 Lista de Desarrollo – Ejecución de pruebas y Resultados (f).	368
Figura 2. 11 Lista de Desarrollo – Revisión de módulos del sistema (g).	368
Figura 2. 12 Lista de Implementación.	369
Figura 2. 13 Lista Realización del Póster.	369
Figura 2. 14 Lista de Capacitación.	370
Figura 2. 15 Entregables y revisiones finales.	370
Figura 2. 16 Ejemplo de un Tablero Kanban	379
Figura 2. 17 Proceso de Scrum	380
Figura 2. 18 Diagrama de Burndown	382
Figura 2. 19 Logo de Discord.	388
Figura 2. 20 Logo de Telegram.	388
Figura 2. 21 Logo de Trello	389
Figura 2. 22 Logo de TeamGantt y Project de Microsoft.	389

Figura 2. 23 Gráfico de recursos Arquitecto de Software y Scrum Master	416
Figura 2. 24 Gráfico de recursos Product Manager y Dev Front-end	416
Figura 2. 25 Gráfico de recursos Dev Back-end y Tester	417
Figura 2. 26 Incremento del número de usuarios mensualmente.	430
Figura 2. 27 Cantidad de meses en que se recupera la inversión en función del número de usuarios de la plataforma y el monto de la donación/suscripción: \$ 30 (azul), \$50 (rojo), \$75 (amarillo), \$100 (verde), \$150 (naranja).	437
Figura 2. 28 Ingresos mensuales mínimo (rojos), máximos (amarillo) y promedios (azul) teniendo en cuenta la cantidad de usuarios realizando aportes, ya sea por donación o suscripción.	438
Figura 2. 29 Cantidad de meses necesarios en función del número de usuarios realizando aportes, teniendo en cuenta variabilidad del porcentaje de usuarios y del valor de las donaciones/suscripciones.	439
Figura 4. 1 Estructura organizativa de T.I., Data Center, áreas relacionadas y áreas de control.	494
Figura 4. 2 Ejemplo de Organigrama de Tipo Departamental.	495
Figura 4. 3 Ejemplo de Tablero de Control Operativo (a)	511
Figura 4. 4 Ejemplo de Tablero de Control Operativo (b)	511
Figura 4. 5 Ejemplo de Tablero de Control del área de Nuevos Proyectos de T.I.	512
Figura 4. 6 Ejemplo de tablero de control del área Nuevos Proyectos de T.I.	512
Anexo Nº 2 - Figura 1. 1 Diagrama de Recursos de Arquitecto de Software.	522
Anexo Nº 2 - Figura 1. 2 Diagrama de Recursos de Dev Back-end	523
Anexo Nº 2 - Figura 1. 3 Diagrama de Recursos de Dev Front-end.	524
Anexo Nº 2 - Figura 1. 4 Diagrama de Recursos de Product Manager.	525
Anexo Nº 2 - Figura 1. 5 Diagrama de Recursos de Scrum Master.	526
Anexo Nº 2 - Figura 1. 6 Diagrama de Recursos de Tester.	527
Anexo Nº5 - Figura 1. 1 Pantalla de Login del Administrador.	537
Anexo Nº5 - Figura 1. 2 Pantalla de logueo exitoso.	537
Anexo Nº5 - Figura 1. 3 Pantalla de opciones del Administrador general.	538
Anexo Nº5 - Figura 1. 4 Pantalla de opciones del Content Manager.	539
Anexo Nº5 - Figura 1. 5 Pantalla Gestionar Reglas de Negocio.	540
Anexo Nº5 - Figura 1. 6 Pantalla regla de negocio modificada, mensaje de éxito.	541
Anexo Nº5 - Figura 1. 7 Pantalla de Reportes.	542
Anexo Nº5 - Figura 1. 8 Pantalla Agregar una Categoría.	543
Anexo Nº5 - Figura 1. 9 Pantalla para Editar una Categoría.	543
Anexo Nº5 - Figura 1. 10 Pantalla de éxito, Categoría modificada.	544
Anexo Nº5 - Figura 1. 11 Pantalla de éxito, Categoría eliminada.	545
Anexo Nº5 - Figura 1. 12 Pantalla Gestionar Publicaciones del Content Manager.	545
Anexo Nº5 - Figura 1. 13 Pantalla Detalles de la Publicación.	546



Anexo N°5 - Figura 1. 14 Pantalla Publicación Aprobada.	547
Anexo N°5 - Figura 1. 15 Pantalla Rechazar la Publicación.	548
Anexo N°5 - Figura 1. 16 Pantalla Gestionar Contratos.	549
Anexo N°5 - Figura 1. 17 Pantalla mensaje Contrato Aprobado.	550
Anexo N°5 - Figura 1. 18 Pantalla mensaje Contrato Rechazado.	550
Anexo N°5 - Figura 1. 19 El usuario no está disponible	551
Anexo N°5 - Figura 1. 20 El correo no está disponible	551
Anexo N°5 - Figura 1. 21 Pantalla de logueo erróneo.	552
Anexo N°5 - Figura 1. 22 Nombre de usuario erróneo.	552
Anexo N°5 - Figura 1. 23 Pantalla ingreso de contraseña incorrecta.	553
Anexo N°5 - Figura 1. 24 Las contraseñas ingresadas no coinciden.	553
Anexo N°5 - Figura 1. 25 Pantalla mensaje de error al editar perfil.	554
Anexo N°5 - Figura 1. 26 Usuario ingresa contraseña incorrecta, no puede eliminar la cuenta	555
Anexo N°5 - Figura 1. 27 Pantalla cuenta eliminada, el usuario ya no puede acceder.	555
Anexo N°5 - Figura 1. 28 Nombre ingresado no es igual al que figura en la tarjeta.	556
Anexo N°5 - Figura 1. 29 Número de tarjeta ingresado incorrecto.	556
Anexo N°5 - Figura 1. 30 Código de verificación tiene fecha actual.	557
Anexo N°5 - Figura 1. 31 Saldo de la tarjeta insuficiente	557
Anexo N°5 - Figura 1. 32 Resolución inadecuada a la permitida para imágenes.	558
Anexo N°5 - Figura 1. 33 No seleccionó página alguna publicación a subir	558
Anexo N°5 - Figura 1. 34 Resolución inadecuada a la permitida para portada.	558
Anexo N°5 - Figura 1. 35 Resolución inadecuada a la permitida para portada.	559
Anexo N°5 - Figura 1. 36 Ingreso a una ruta que no tiene permisos.	559
Anexo N°5 - Figura 1. 37 No se puede eliminar Categoría.	560
Anexo N°5 - Figura 1. 38 Ingreso a localhost:8080.	578
Anexo N°5 - Figura 1. 39 Autenticación del administrador. Estos datos pueden variar por razones de seguridad.	578
Anexo N°5 - Figura 1. 40 Tablero de administración de PostgreSQL mediante pgAdmin.	579
Anexo N°5 - Figura 1. 41 Creación de la conexión con el motor de base de datos.	579
Anexo N°5 - Figura 1. 42 Ingreso de dirección IP donde se encuentra el SGBD corriendo. Puerto por defecto.	580
Anexo N°5 - Figura 1. 43 Ingreso de datos de credenciales para acceso a base de datos.	581
Anexo N°5 - Figura 1. 44 Creamos la base de datos.	581
Anexo N°5 - Figura 1. 45 Base de datos de nombre "omics". Es importante respetar el nombre.	582
Anexo N°5 - Figura 1. 46 Ejecución del comando make clean-db migrate.	583

Anexo N°5 - Figura 1. 47 Tablas de la base de datos de Omics	584
Anexo N°5 - Figura 1. 48 Población de tablas con datos aleatorios con comando make populate	585
Anexo N°5 - Figura 1. 49 Opción de Backup del Panel de administración de pgAdmin.	585
Anexo N°5 - Figura 1. 50 Realización de backup por Interfaz gráfica.	586
Anexo N°5 - Figura 1. 51 Interfaz gráfica de PgAdmin.	587
Anexo N°5 - Figura 1. 52 Realizar backup por PgAdmin.	588
Anexo N°5 - Figura 1. 53 Opciones de Backup.	588
Anexo N°5 - Figura 1. 54 Backups realizados.	589

## Diagramas

Diagrama 1. 1 Diagrama de contexto de interfaces detectadas en Mangatoon.	29
Diagrama 1. 2 Modelo de entidades de Mangatoon.	41
Diagrama 1. 3 Diagrama de Casos de Uso general de Mangatoon.	42
Diagrama 1. 4 Diagrama de contexto de interfaces detectadas en Tapas.	46
Diagrama 1. 5 Modelo de entidades de Tapas.	57
Diagrama 1. 6 Diagrama de casos de uso de Tapas	58
Diagrama 1. 7 Diagrama de contexto de interfaces detectadas en Webtoon.	60
Diagrama 1. 8 Diagrama de entidades de Webtoon	66
Diagrama 1. 9 Diagrama de contexto de interfaces detectadas en Crunchyroll.	68
Diagrama 1. 10 Modelo de entidades de Crunchyroll.	77
Diagrama 1. 11 Diagrama de Casos de Uso general de Crunchyroll.	78
Diagrama 1. 12 Diagrama de contexto de interfaces detectadas en Rootinks.	82
Diagrama 1. 13 Modelo de entidades de Rootinks.	88
Diagrama 1. 14 Diagrama de Casos de Uso.	119
Diagrama 1. 15 Diagrama de Estado para Publicación.	203
Diagrama 1. 16 Diagrama de Estado para Contrato.	204
Diagrama 1. 17 Diagrama de Estado para Suscripción.	204
Diagrama 3. 1 Proceso de Integración	470
Anexo N°5 - Diagrama 1. 1 Flujo de Publicación de Obra	561

## Tablas

Tabla 1. 1 Comparación de funcionalidades de sistemas relevados.	91
Tabla 1. 2 Correspondencia entre las salidas del sistema, las historias de usuarios y las pantallas.	196
Tabla 1. 3 Tecnologías utilizadas en el desarrollo	209

Tabla 1. 4 Plantilla base de documentación de código.	210
Tabla 1. 5 Plantilla de documentación de código de "Publicar Obra".	261
Tabla 1. 6 Planilla base de pruebas.	274
Tabla 1. 7 Caso de Prueba de Validación de Ingreso de Datos 01.	284
Tabla 1. 8 Caso de Prueba de Validación de Ingreso de Datos 02.	286
Tabla 1. 9 Caso de Prueba de Validación de Ingreso de Datos 03.	289
Tabla 1. 10 Caso de Prueba de Lógica de los Módulos Principales 01.	292
Tabla 1. 11 Caso de Prueba de Lógica de los Módulos Principales 02.	293
Tabla 1. 12 Caso de Prueba de Lógica de los Módulos Principales 03.	296
Tabla 1. 13 Caso de Prueba de integración entre los Módulos del Sistema 01.	299
Tabla 1. 14 Caso de Prueba de integración entre los Módulos del Sistema 02.	301
Tabla 1. 15 Caso de Prueba de integración entre los Módulos del Sistema 03.	305
Tabla 1. 16 Caso de Prueba de Seguridad por Niveles de Usuario 01.	308
Tabla 1. 17 Caso de Prueba de Seguridad por Niveles de Usuario 02.	309
Tabla 1. 18 Caso de Prueba de Seguridad por Niveles de Usuario 03.	313
Tabla 1. 19 Caso de Prueba de Carga 01.	314
Tabla 1. 20 Caso de Prueba de Carga 02.	316
Tabla 1. 21 Caso de Prueba de Carga 03.	317
Tabla 1. 22 Caso de Prueba de Carga 04.	318
Tabla 2. 1 División del proyecto en Sprints.	376
Tabla 2. 2 Roles y asignaciones de integrantes del equipo.	385
Tabla 2. 3 Recursos para cada una de las actividades.	405
Tabla 2. 4 Cantidad de horas mensuales y totales por recurso.	406
Tabla 2. 5 Cantidad de horas por recurso y tarea.	415
Tabla 2. 6 Utilización de los recursos	415
Tabla 2. 7 Capacidades y precios de VPS (servidor privado virtual) de DigitalOcean. Recuperado de: <a href="https://www.digitalocean.com/pricing">DIGITALOCEAN.COM/PRICING</a> Fecha: 14/04/2020	420
Tabla 2. 8 Porcentaje de incremento de usuarios mensualmente	430
Tabla 2. 9 Costos recurrentes durante 9 meses.	434
Tabla 2. 10 Meses necesarios para recuperar la inversión en función de los usuarios.	436
Tabla 2. 11 Equipo disponible para operar.	442
Tabla 2. 12 Nivel del equipo en cada tecnología.	443
Tabla 2. 13 Costos totales del proyecto.	446
Tabla 2. 14 Costos desagregados por mes desde Marzo a Julio.	447
Tabla 2. 15 Costos desagregados por mes desde Agosto a Noviembre.	447
Tabla 2. 16 Costo total.	447
Tabla 2. 17 Tabla de Riesgos.	451
Tabla 2. 18 Ranking de Riesgos.	451
Tabla 2. 19 Análisis de riesgos-Medida Preventiva.	453

Tabla 2. 20 Análisis de riesgos-Medida Correctiva.	454
Tabla 2. 21 Impacto ambiental - Acción: Plataforma "Omics".	456
Tabla 2. 22 Impacto ambiental - Acción: Módulo publicación de obras.	456
Tabla 3. 1 Tabla de referencia de Probabilidad de ocurrencia e impacto de riesgo.	459
Tabla 3. 2 Tabla de ponderación riesgos (de mayor a menor consecuencia).	460
Tabla 3. 3 Tabla de consecuencia y tratamiento de riesgo.	464
Tabla 3. 4 Funciones del Director del Proyecto	473
Tabla 3. 5 Funciones principales del Director del Proyecto en fase de ejecución.	475
Tabla 4. 1 Actividades del primer año.	516
Tabla 4. 2 Actividades del segundo año.	518