


UTN * SANTA FE

PROYECTO FINAL DE CARRERA

INGENIERÍA EN SISTEMAS DE INFORMACIÓN

Sistema de Gestión de Consultorios “HEALMATE”

Informe Final de Proyecto

Integrantes:

Inalbon, Nelson

Milesi, Matías

Rosset, Víctor

Director:

Ing. Rodrigo Ledesma

Índice

Índice	2
1. Introducción	9
1.1 Contexto	9
1.2 Análisis de la necesidad	9
1.3 Objetivos	10
1.4 Método de Comercialización	12
1.5 Aportes esperados	12
1.6 Organización del informe	12
2. Marco Teórico	14
2.1 Interacción entre las Herramientas	14
2.2 Diseño de Arquitectura	15
2.2.1 Capa de presentación	15
2.2.2 Capa de negocio	16
2.2.3 Capa de persistencia	16
2.2.4 Integración de Capas	16
3. Gestión de Proyecto	18
3.1 Metodología Utilizada	18
3.1.1 Aspectos buscados en la Metodología a Utilizar ad hoc	18
3.1.2 Fundamentos en la elección de la Metodología	19
Kanban	19
Extreme Programming	20
SCRUM	20

3.2 Implementación de la Metodología ad hoc	20
3.2.1 Story Points	22
3.2.2 Implementación de Programación en parejas:	22
3.2.3 Documentación	22
3.2.4 Reuniones de Planificación:	23
3.2.5 Reuniones Diarias:	23
3.2.6 Reunión de Retrospectiva:	24
3.2.7 Reunión de Revisión:	24
3.3 Implementación	24
3.3.1 Reunión Inicial	25
3.3.2 Diagramas de Flujo	26
Asignación de Turnos	26
Recurso	26
Tratamiento	27
Tarifa de Servicio (Aranceles)	27
3.3.3 Aclaraciones sobre las capturas de Análisis de Requerimientos Iniciales:	30
3.4 Sprints	31
3.4.1 Primer sprint	31
3.4.2 Segundo sprint	33
3.4.3 Tercer sprint:	37
3.4.4 Cuarto sprint	41
3.4.5 Quinto sprint:	44
3.4.6 Sexto sprint:	47
3.4.7 Séptimo sprint:	50

3.4.8 Octavo sprint:	53
3.5 Burn-up Chart del Proyecto	57
3.6 Testing	57
3.6.1 Pruebas de Unidad	58
3.6.2 Pruebas Funcionales	59
3.6.3 Pruebas de Aceptación	61
3.6.4 Detalle de Pruebas	61
3.7 Integración Continua	68
3.8 Gestión del Código	69
3.8.1 Branching o manejo de ramas	71
3.8.2 Implementación de Branching	72
4 Marco de Trabajo	75
4.1 Implementación de la interfaz de usuario	75
4.1.1 Página de Bienvenida	75
4.1.2 Inicio de Sesión	76
4.1.3 Barras de MENU	77
4.1.3.a Barra Lateral	77
4.1.3.b Barra Superior	78
4.1.4 Funcionamiento en Dispositivos Móviles	79
4.1.5 Paletas de Colores	80
4.1.6 Iconografía	81
4.1.7 Tipografía	82
4.1.8 Mensajes y Alertas	82
4.1.9 Calendario	83

4.2 Presentación del Sistema	85
4.2.1 Usuarios	85
4.2.2 Ingreso al sistema	87
4.2.3 Consultorio, Kinesiólogos y Especialidades	87
4.2.4 Recursos	89
4.2.5 Tratamientos	91
4.2.6 Obras Sociales	92
4.2.7 Pacientes	93
4.2.8 Turnos	94
4.2.9 Rehabilitaciones	94
4.2.10 Orden Médica	95
4.2.11 Notificaciones	97
4.2.12 Informes	97
5. Conclusiones y Trabajos Futuros	100
5.1 Principales aportes	100
5.2 Experiencias	100
5.3 Trabajos Futuros	100
6. Glosario	102
6.1 Técnico de Consultorio	102
6.2 Técnico Software	103
7. Bibliografía	106
8 Anexos	107
8.1 Análisis Inicial de Habilidades	107
8.2 Análisis para elección de Calendario	108

8.3 Ejemplos de Minutas de Reunión:	108
8.4 Historia Clínica vs Ficha de Turno	110
8.5 Tratamientos VS Plantilla Tratamientos	111
8.6 Requerimientos funcionales	112
8.6.01 Modelo Básico de Interfaz	112
8.6.02 ABM de Consultorios	113
8.6.03 ABM de Usuario	115
8.6.04 ABM de Pacientes	116
8.6.05 ABM de Turnos	118
8.6.06 ABM de Kinesiólogos	120
8.6.07 ABM de Patologías	122
8.6.08 ABM Tipo de Recursos	124
8.6.09 Módulo de Recursos	125
8.6.10 ABM de Obras Sociales	126
8.6.11 Gestión por calendario funcional	128
8.6.12 Notificaciones y Mensajes	129
8.6.13 ABM de Tratamientos	130
8.6.14 Gestión de Aranceles	132
8.6.15 ABM Variables de seguimiento	133
8.6.16 ABM Rehabilitación	134
8.6.17 ABM de Orden Médica	136
8.6.18 ABM Especialidades Médicas	137
8.6.19 Informes de Turnos	138
8.6.20 Módulo de Sesiones de Tratamientos	139

8.6.21 Informe: Variable de Seguimiento	140
8.6.22 ABM Lugar de Guardado	141
8.6.23 Diseño de Dashboard	141
8.7 Herramientas	143
Jira Software	143
Bitbucket	144
Docker	144
MySql	145
NGINX	146
Symfony	146
PHPUnit	147
BOUML	147
LucidChart	147
PHP	148
JavaScript	148
HTML	148
8.9 Bundles en Symfony	149
TwigBundle	149
DoctrineBundle	149
SensioFrameworkExtraBundle	150
FOSUserBundle	150
JMSSerializerBundle	151
AdminLTEBundle	151
KnpgaufretteBundle	152

TetranzSelect2EntityBundle	152
DigitalAsceticBaseEntityBundle	152
MobileDetectBundle	152
CalendarBundle	152
8.10 Aplicación de las herramientas	152
Configuración de las estaciones de trabajo:	152
8.11 Testing – Casos de Prueba	155
8.11.1 Casos de Prueba Automatizados.	155
Sprint 2	156
Sprint 3	163
Sprint 4	170
8.11.2 Casos de Prueba de Aceptación:	172
Sprint 2	172
Sprint 3	176
Sprint 4	182
Sprint 5	195
Sprint 6	204
Sprint 7	210
Sprint 8	214

1. Introducción

El objetivo de este Informe es presentar el proyecto final de carrera titulado “Healmate - Sistema de gestión de Consultorios”. Para hacerlo realizaremos una explicación del contexto y un análisis de la necesidad del consultorio de kinesiología y de los objetivos perseguidos en el proyecto. Definiremos luego el alcance del mismo y la forma en la que se presentarán los contenidos del informe.

1.1 Contexto

Biokinetic Sport es un consultorio de Kinesiología cuya actividad es la evaluación, tratamiento y rehabilitación de pacientes a través del seguimiento de su terapia y entrenamiento mediante historia clínica.

Su misión es mejorar la calidad de vida de sus pacientes, esforzándose en lograr una mejora diaria de los tratamientos aplicados.

1.2 Análisis de la necesidad

Los profesionales de Kinesiología de *Biokinetic Sport* carecen de una rápida disponibilidad de información histórica de sus pacientes, esto impide realizar búsquedas específicas según tipo de tratamiento, grado de lesión y otros parámetros que podrían ser configurables y registrados durante los tratamientos que realizan. Según William Thomson Kelvin (1824 – 1907): “*Lo que no se define no se puede medir. Lo que no se mide, no se puede mejorar. Lo que no se mejora, se degrada siempre*”. Por lo tanto una necesidad que surge es mejorar la información almacenada de los tratamientos aplicados (al medir los resultados con variables representativas) lo que permitirá a *Biokinetic Sport* su análisis posterior de cada tratamiento, y desarrollar rehabilitaciones de los pacientes cada día más precisas.

Los profesionales de Kinesiología de *Biokinetic Sport* se enfrentan al difícil trabajo de planificar el cobro de sus servicios a las obras sociales. La cantidad de variables para planificar el cobro depende de la obra social y la región geográfica. Lo hace que este proceso sea desgastante.

Biokinetic Sport tiene diferentes equipamientos en sus consultorios (camillas, ultrasonidos, equipamiento de gimnasio, etc.), los cuales permiten que, según el tratamiento, pueda personalizar el turno para optimizar los recursos disponibles y el porcentaje de utilización de dichos recursos. Esta personalización se podría realizar estableciendo diferentes tipos de tratamientos según los diagnósticos.

1.3 Objetivos

Los objetivos planteados para este proyecto están asociados directamente a solucionar las problemáticas planteadas en el Análisis de la necesidad. Desarrollando un producto software que permita a los profesionales de *Biokinetic Sport* gestionar los turnos, historias clínicas y liquidación de pagos de obras sociales en forma eficaz.

Para ello se deberá generar una herramienta que:

- Vincule las Obras Sociales y los Kinesiólogos de *Biokinetic Sport* optimizando los procesos de seguimiento y cobro de las órdenes.
- Permita a los Kinesiólogos generar plantillas de *historias clínicas* personalizadas y utilizarlas en diferentes tipos de tratamientos.
- Permita personalizar los tratamientos teniendo en cuenta los siguientes recursos para cada uno:
 - Duración de los turnos.
 - Espacio necesario.
 - Equipos a utilizar.
 - Porcentaje de utilización de Kinesiólogo.
- Asigne turnos mediante el análisis de los recursos disponibles para los tratamientos optimizando la utilización de estos recursos.
- Permita la administración de los datos de los Consultorios, Kinesiólogos, Obras Sociales y Pacientes de *Biokinetic Sport*.

Los requerimientos no funcionales que se detallan a continuación no se refieren directamente a las funciones específicas que entrega el sistema, sino a las propiedades emergentes de éste como la fiabilidad, la respuesta en el tiempo, la capacidad de almacenamiento y características como atributos de calidad principales que se considera que debe tener el sistema. Son las condiciones que impone el cliente al programa que necesita.

Se pueden agrupar en dos categorías generales:

1) Requisitos de calidad de ejecución:

- a) **Seguridad:** Al tratarse de un sistema que maneja datos médicos personales de los tratamientos y evolución de los pacientes, deberemos asegurar que el sistema sea utilizado sólo por las personas autorizadas por el director de *Biokinetic*.
- b) **Usabilidad:** Mantener una interfaz (páginas web) de usuario intuitiva y fácil de usar, para facilitar la capacitación inicial, mejorar la interacción con los usuarios y evitar errores de carga.
- c) **Disponibilidad:** Detener la ejecución del sistema ante cualquier falla que se presente en el mismo y notificar al usuario sobre esta situación. Con esto se permitiría una recuperación rápida, ya sea por reingreso al proceso en el cual se realiza la falla o bien, mediante recuperación de respaldos para que el usuario pueda volver a operar el sistema con normalidad.

2) Requisitos de calidad de evolución:

- a) **Testeabilidad:** Se refiere a que el sistema sea testeable. Esto es poder crear mocks de las dependencias, testear y asegurar el correcto funcionamiento de nuestro código; de forma de poder probar todo el sistema y asegurarnos de obtener la calidad deseada.

- b) **Escalabilidad:** Es la propiedad de aumentar la capacidad de trabajo o de tamaño de un sistema sin comprometer el funcionamiento y calidad normales del mismo. El sistema es escalable cuando puede aumentar el número de usuarios, de datos que procesa o de solicitudes que recibe, sin que se afecte significativamente su velocidad de respuesta.
- c) **Modificabilidad:** Facilitar la adecuación a futuros cambios que puedan requerirse.

1.4 Método de Comercialización

Planteamos un Modelo de comercialización *Freemium*. Esta siglas provienen de la contracción de las palabras en inglés “Free” y “Premium”, el cual es un modelo de negocio donde se brinda un servicio gratuito y funcional básico y completo (en éste ofreceremos el sistema de turnos, patologías, pacientes y variables de recuperación, así como su potente interfaz para dispositivos móviles) intentando captar clientes de forma masiva, con esto se espera que un porcentaje de ellos opte por elegir las funcionalidades premium como las Historias Clínicas y control de Órdenes Médicas y sus cobros. Pensamos que este método nos permitirá llegar a más potenciales clientes, generando una colaboración con el consultorio de Healmate de WIN-WIN, que tiene como objetivo que todas las partes salgan ampliamente beneficiadas.

1.5 Aportes esperados

Se espera brindar una solución de calidad en base a la experiencia y el aprendizaje, construyendo un producto orientado a la medida de las necesidades ya mencionadas del consultorio, con la finalidad de contribuir al éxito y al manejo eficiente de las actividades que este desempeña.

Asimismo, se intenta orientar a la empresa a una nueva era de información mejorando su eficiencia tanto en la administración de sus tiempos y tareas, como así también la eficiencia de sus servicios, permitiéndole mejorar día a día sus tratamientos.

1.6 Organización del informe

1. La elaboración del informe se realizó presentando las principales funcionalidades del sistema que satisfacen las necesidades planteadas. Procedimos luego a explicar de qué forma se alcanzarán los objetivos planteados.

2. Conceptos y Herramientas: Contiene los conceptos y terminologías principales que se mencionan en el informe, la descripción de las herramientas y tecnologías aplicadas en el proyecto, y la manera en que se integran estas últimas

3. Gestión del proyecto: Contiene una descripción de la metodología de trabajo adoptada, detalle de cada *Sprint* y de sus respectivas tareas y estimaciones.

4. Solución: define la forma y las pautas de trabajo adoptadas por el equipo y una descripción de la infraestructura de desarrollo, de la arquitectura del sistema, de los aspectos de configuración aplicados y una presentación final del sistema como solución.

5. Conclusiones y Trabajos Futuros: se concluye sobre la metodología de trabajo adoptada, las herramientas y tecnologías aplicadas, el equipo del proyecto y los aportes a la empresa.

6. Glosario: es un catálogo de las palabras poco frecuentes y terminologías utilizadas en el proyecto.

7. Bibliografía: contiene todas las referencias bibliográficas citadas a lo largo del informe.

8. Anexos: contiene la información complementaria y técnica que aporta más detalle a la hora de profundizar sobre aspectos relevantes en el desarrollo del proyecto.

2. Marco Teórico

En esta sección daremos una breve descripción de los conceptos y terminologías para que en el desarrollo del proyecto se logre entender fácilmente el lenguaje y las palabras utilizadas por personas no afines de la Kinesiología. Seguidamente definiremos aspectos relacionados a las Herramientas utilizadas, la aplicación de las mismas, la lógica del negocio y luego presentaremos aspectos del desarrollo de software.

La terminología utilizada de un Consultorio de Kinesiología se encuentra en el Cap. 6, denominado Glosario, donde se explican los términos utilizados.

2.1 Interacción entre las Herramientas

Las Herramientas Elegidas para realizar el Proyecto fueron seleccionadas buscando la mayor integración posible entre las mismas en todo el desarrollo del Software.

Explicamos brevemente cómo se interrelacionan las herramientas utilizadas:

Utilizaremos **Jira** para tener un *espacio de trabajo colaborativo*, ordenando las tareas e integrando las demás herramientas. El mismo tiene integración con:

Bitbucket nos *permitirá gestionar el código* con un gestor GIT, facilitando el versionado de código y el control de cambios.

Docker nos permitirá realizar una *‘virtualización ligera’*, para poder empaquetar entornos y aplicaciones. Bitbucket tiene un sistema de Integración Continua basado en esta tecnología que nos permitió correr las pruebas de test automáticamente.

Symfony es un framework¹ diseñado para *optimizar el desarrollo de las aplicaciones web* basado en el patrón Modelo Vista Controlador. Separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web, se conecta a **MySQL** que es un sistema de gestión de *bases de datos relacional* y utilizaremos **NGINX** como *servidor web*. Symfony también tiene integración con **PHPUnit** para realizar las pruebas de unidad

¹Un Framework asiste el desarrollo del proyecto mediante una estructura definida (ver Glosario).

Con esto, en la terminal de desarrollo (computadora donde se programa y prueba el código), tenemos todo lo necesario para trabajar local e independientemente del servidor de producción, permitiendo integración continua y testing en Bitbucket.



Fig. 2.1-01: Ilustración de la integración de las herramientas

Podrá ver detalles sobre las Herramientas utilizadas en el Anexo 8.7 y 8.9. Para ver los detalles de los Lenguajes utilizados ver Anexo 8.8 y las configuraciones básicas en el Anexo 8.10

2.2 Diseño de Arquitectura

El diseño de la arquitectura propuesto para este desarrollo consiste en un modelo de tres capas. En este modelo la lógica de negocios, la capa de presentación y la capa de datos residen en un servidor o varios (servidor web) que brindan la interacción a través del navegador web de preferencia, logrando separar la interfaz de usuario de la lógica de la aplicación. De esta forma, es sencillo y mantenible crear diferentes interfaces sobre un mismo sistema sin requerir cambio alguno en la capa de datos o lógica. A continuación, describimos brevemente cada una de estas capas.

2.2.1 Capa de presentación

Esta es la capa que ve el usuario (también se la denomina capa de usuario), presenta el sistema al usuario, le comunica y captura la información de este en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). También es conocida como interfaz gráfica y debe tener la característica de ser “amigable” (entendible y fácil de usar). Esta capa se comunica únicamente con la capa de negocio.

2.2.2 Capa de negocio

Aquí es donde se implementan las principales funcionalidades del sistema y donde se encapsula toda la lógica de negocio. Incluye todos los servicios que responden a los requisitos funcionales de la aplicación.

Esta capa, se encarga de comunicarse con la capa de presentación para recibir las solicitudes y luego presentarle los resultados obtenidos, y con la capa de persistencia para recuperar y almacenar los datos utilizados por la aplicación.

2.2.3 Capa de persistencia

Es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos en una base de datos. Reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

2.2.4 Integración de Capas

Como se observa en la siguiente figura cada capa tiene su analogía en el diseño de la solución.

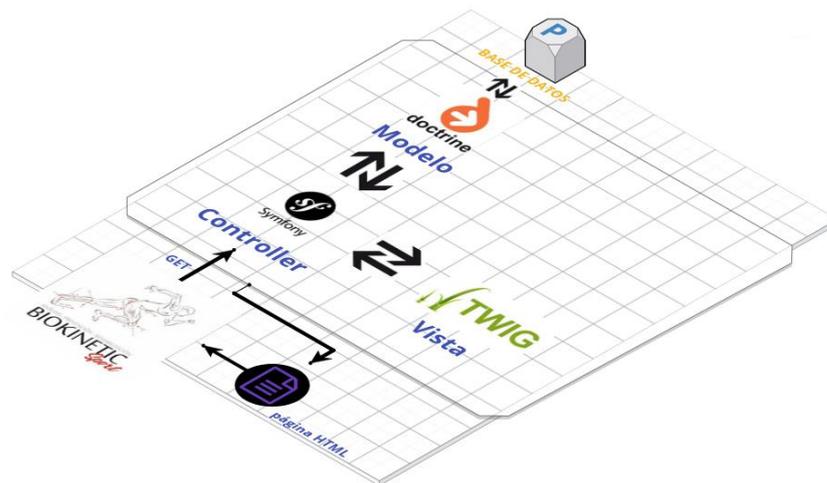


Fig. 2.2.4-01: Herramientas utilizadas y capas

El servidor NGINX será el Servidor Web el cual soportará el sistema manejando las conexiones procedentes del Cliente a Symfony y viceversa. El siguiente gráfico muestra en una simplificación cómo se conectará el usuario para interactuar con el sistema.



Fig. 2.2.4-02: Interacción entre usuario y sistema

3. Gestión de Proyecto

En esta sección daremos una breve explicación de la metodología utilizada para la realización del proyecto

3.1 Metodología Utilizada

3.1.1 Aspectos buscados en la Metodología a Utilizar ad hoc

Para llevar adelante el proyecto analizamos varias metodologías ágiles y seleccionamos los puntos fuertes de estas (ver 3.2 - Implementación de la Metodología) **generando una metodología ad hoc²**.

Buscamos un marco de trabajo flexible, que nos brindó la posibilidad de minimizar la documentación y permitió trabajar alternando roles en el equipo de desarrollo, de forma de lograr afianzar los conocimientos adquiridos a lo largo de la carrera.

En la primera etapa de desarrollo optamos por realizar reuniones frecuentes de desarrollo, en las cuales trabajamos en programación de a pares (*pair-programming*) mientras un tercero se encargó de las pruebas para el código que se estaba escribiendo (según el análisis inicial de habilidades - ver Anexo 8.1). Esto permitió alinearnos en cuestiones importantes desde el comienzo (simplificación, comentarios, estilo de código, legibilidad, etc.), permitiendo fijar un estándar de calidad para los siguientes sprints.

Luego de definir unos estándares iniciales pudimos trabajar cómodos y rápido, por cuestiones de trabajo, resultó beneficioso ordenar los tiempos dedicados en el desarrollo ya que al iniciar el proyecto el tiempo a dedicar y la capacidad horaria era holgada (coincidimos en el horario todos los miembros del equipo de desarrollo), lo que cambió sustancialmente luego de las primeras etapas de desarrollo.

Gracias a la capacidad de adaptación que nos permitía trabajar con metodologías **ad hoc** pudimos continuar el trabajo de forma eficiente adaptándonos al nuevo entorno que surgía.

²**Ad hoc.** Generalmente se refiere a una solución específicamente elaborada para un problema o fin preciso y, por tanto, no generalizable ni utilizable para otros propósitos.

Es importante tener en cuenta que la metodología también fue cambiando a lo largo del desarrollo del proyecto; a medida que avanzábamos, los roles y asignación de tareas no solo dependieron de las cualidades técnicas sino también de la disponibilidad horaria de los integrantes del grupo. Era de fundamental importancia que el grupo realice el trabajo de forma ordenada permitiendo potenciar las cualidades individuales de todos los integrantes del equipo y disminuir así la incertidumbre en el proceso de desarrollo.

La retroalimentación de *Biokinetic Sport* fue fundamental para cumplir con el objetivo de lograr un producto de calidad y tener entregas del producto funcionando a medida que avanzaba el proceso de desarrollo. Esto nos ayudó a evitar problemas de incumplimiento de agendas, desbordamiento de costes y funcionalidad deficiente.

3.1.2 Fundamentos en la elección de la Metodología

La metodología se basó en algunas prácticas provenientes de *Kanban*, *XP* y *Scrum*. Donde *se tomaron los aspectos principales sin compromiso de seguimiento*, ya que de no ser así estábamos atados a procedimientos (los cuales no dejan de ser buenas prácticas), pero requerían de un esfuerzo que no sumaría significativamente más de lo que elegimos como directivas para el proyecto. A modo de referencia, citamos una parte de Manifiesto Ágil firmado el 11 de febrero de 2001:

“Individuos e interacciones sobre procesos y herramientas.

Software funcionando sobre documentación extensiva.

Colaboración con el cliente sobre negociación contractual.

Respuesta ante el cambio sobre seguir un plan.

Esto es, aunque valoramos los elementos de la derecha, valoramos más los de la izquierda.”

A continuación, describiremos brevemente las metodologías de las cuales utilizamos algunos de sus principios y describiremos cuales tomamos solo en algunas etapas del desarrollo:

Kanban

Kanban se basa en el desarrollo incremental, dividiendo el trabajo en partes. Uno de los principales aportes, es que utiliza técnicas visuales para ver la situación de cada tarea. El

trabajo se divide en *partes* o *historias*, que deberían tener la estimación de la duración de la tarea.

El objetivo de esta visualización es que quede claro el trabajo a realizar, en qué está trabajando cada persona, que todo el mundo tenga algo que hacer y el tener clara la prioridad de las historias. Las fases del ciclo de producción o flujo de trabajo se definen según el proyecto.

Extreme Programming

La metodología *Extreme Programming (XP)* favorece que haya una comunicación constante con el cliente y dentro del equipo de trabajo, de esto dependerá que el desarrollo se lleve a cabo de una manera sencilla, entendible y que se entregue al cliente lo que necesita. Además, esta comunicación genera una retroalimentación constante.

Se espera que los desarrolladores³ sean valientes y eliminen y/o modifiquen código que ya no es útil o se puede mejorar, así como de funcionalidades que el cliente considera que ya no se requiere.

SCRUM

La metodología *SCRUM* es muy utilizada en la actualidad. Uno de sus puntos fundamentales es que después de cada sprint se ha completado un requerimiento concreto y el usuario ya puede empezar a utilizarlo sin necesidad de que el proyecto esté completado. A partir del mismo se podrán hacer entregas para mejorar la calidad del producto. Desde el final del sprint se tiene una funcionalidad desarrollada, aunque sea limitada.

Las prioridades se establecen en base a la funcionalidad. Lo más importante es que el producto pueda ponerse en uso, mientras que cuestiones accesorias o relativas al diseño ocuparían un segundo lugar. El proyecto se debe entender como una constante evolución desde el primer sprint.

3.2 Implementación de la Metodología ad hoc

Con la mayoría de los conceptos explicados detallamos la implementación de la metodología.

³ Parte del equipo de trabajo que se dedica a uno o más aspectos del proceso de desarrollo de software.

Para realizar el análisis inicial, nos basamos en la idea del Software como Producto Final a desarrollar. Esta visión al comienzo es difusa, pero durante el desarrollo del proyecto se va puliendo como resultado de reiteradas reuniones de aceptación y el feedback del cliente. A continuación, mencionamos los principales aspectos considerados a la hora de implementar la metodología ad hoc.

De la metodología de *Kanban*⁴ utilizaremos sus funcionalidades visuales para ver las tareas en curso y evitar cuellos de botella. También *Kanban* será la forma de seguimiento del trabajo para el equipo. A la par, esta metodología tiene un contexto más amplio que el desarrollo de software, por lo que permitirá al Cliente realizar un seguimiento diario de todo lo que ocurre en todo momento durante el desarrollo.

De la metodología *XP* utilizaremos en reiteradas ocasiones, principalmente al comienzo, la técnica de programación en parejas⁵. También las premisas de que todos son dueños del código, el código debe ser simple y entendible, programando sólo lo necesario y lo que se utilizará.

De la metodología *Scrum* implementaremos el desarrollo y entrega de funcionalidades completas⁶ en períodos cortos de tiempo, donde estas características se seleccionan por la importancia y el valor que aporta al cliente.

El control del proyecto se basó en sprints (similar a *SCRUM*), las cuales se planificaron en base a estimación de esfuerzo (todos los integrantes del grupo en forma conjunta) de las tareas necesarias para poder completar cada objetivo/requisito, creando así la lista de tareas del sprint (*Sprint Backlog*).

⁴ Permite ser complementado por otras metodologías y es muy recomendable su utilización cuando se tiene confianza plena en el equipo de desarrollo.

⁵ La programación en pareja es una técnica que sugiere que todo el código producido en un proyecto de desarrollo de software sea implementado por dos personas juntas, delante de la misma computadora. Las personas se turnan en el teclado. A la persona que está con las manos en el teclado se lo llama conductor y a la otra navegador.

⁶ Completas en la iteración, si en la retro se decide cambiar ya pasa a ser un requerimiento para los próximos *sprints*.

Para realizar una planificación, seguimiento y análisis de resultados, necesitábamos utilizar una medida de gestión (estimación de esfuerzo en cada tarea), para ello utilizamos *Story Points*.

3.2.1 Story Points

Los *Story Points* son una expresión del esfuerzo que requiere determinadas tareas para ser realizadas. Asignar puntos a todas las tareas a realizar en determinado sprint, tiene como finalidad facilitar la planificación y la evaluación de los resultados (medir para mejorar).

3.2.2 Implementación de Programación en parejas:

Trabajamos programando de a dos personas, sobre todo en los primeros sprints del desarrollo del proyecto, con un tercer integrante que participaba también activamente leyendo el código escrito por los otros y desarrollando las pruebas paralelamente.

Todo el código de producción lo escriben dos personas frente a una misma PC. Cada integrante de la pareja realiza un rol específico: uno codifica en la PC y hace su mayor esfuerzo para generar código de calidad, el otro piensa más estratégicamente, *¿Va a funcionar?, ¿Puede haber pruebas donde no funcione?, ¿Hay forma de simplificar el sistema global para que el problema desaparezca?*

El emparejamiento fue dinámico, rotando según la disponibilidad horaria de los integrantes y permitiendo, sobre todo en una primera instancia, adquirir las mejores prácticas de nuestros compañeros.

3.2.3 Documentación

Inicialmente, por un esfuerzo extra en análisis de requerimientos y madurez del proceso en el Cliente, resultó sencillo obtener una idea global del sistema a realizar y casi todas sus entidades y relaciones. De este análisis inicial, surgió un *modelo* que permitió ordenar el trabajo desde su inicio. A medida que avanzaba el desarrollo, este *modelo* se afinaba con mayor detalle, logrando el resultado final. Es conveniente destacar esto, ya que el proyecto no mutó demasiado de sus abstracciones originales quedando el cliente satisfecho (para ver estos cambios, ver Anexo).

Planteado lo anterior, consideramos que la documentación a realizar en la primera etapa fue modesta y en los últimos sprints mínimas.

Con el objetivo de lograr los entregables en los plazos planteados, evitamos generar documentación excesiva y registramos sólo las que consideramos importantes.

Al ser un proyecto que se apoya fuertemente en el Testing para garantizar su calidad, consideramos importante trabajar con documentación sobre los *casos de Pruebas y Aceptación* y comentando con detalle el código de las pruebas automatizadas.

Trabajamos con las reuniones típicas de los métodos ágiles como *Scrum*, pero de forma relajada y no tan apegados a los momentos en que debían realizarse (para ver los documentos de las reuniones, dirigirse al Anexo).

3.2.4 Reuniones de Planificación:

Realizamos una al inicio de cada sprint para planificar el trabajo encarado en las siguientes semanas. En la misma se plantean las historias que se deben construir y se aprovecha el espacio para evacuar las dudas que surjan, siempre respetando el tiempo luego de presentados los requerimientos.

Para ver un ejemplo de Documentos Generados en estas reuniones (ver anexos 8.4 y 8.5.) en la cual se analizan alternativas de implementación de diferentes requerimientos y propuestas al cliente.

3.2.5 Reuniones Diarias:

Para explicar brevemente la finalidad de las Reuniones diarias enunciamos las preguntas respondidas por cada integrante del equipo en ellas:

- *¿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 dificulte conseguir nuestro objetivo?*

Resultó complicado implementar todas las reuniones diarias, pero se hizo un esfuerzo por realizarlas a todas (en algunos casos se utilizó herramientas como *Zoom* y otras fueron presenciales).

La teoría dice que debe hacerse al comienzo del día para planificar el día de trabajo y refrescar conceptos que serán necesarios tener presentes al momento de desarrollar. No siempre se logró coordinar los horarios para comenzar a trabajar juntos, lo que postergó una hora la realización de estas como regla casi general.

Luego del cuarto sprint, la realizamos cada 3 días, ya que fue muy difícil coordinar horarios para trabajar juntos por motivos personales de los integrantes del grupo.

3.2.6 Reunión de Retrospectiva:

Se realizaron al final de cada *sprint* y en las mismas reflexionamos sobre lo que se había hecho en el sprint. Las mismas fueron muy beneficiosas porque la consideramos como una buena oportunidad para mejorar en los siguientes sprints.

3.2.7 Reunión de Revisión:

Al final de cada sprint, ya con el producto revisado y verificado, se realizaron las reuniones de revisión en las cuales nos planteamos las siguientes preguntas:

- ¿Cómo fue el sprint pasado?
- ¿Qué problemas tuvimos?
- ¿Qué hicimos bien?
- ¿Qué queremos mejorar?
- ¿Qué vamos a hacer para conseguirlo?

En sí, las reuniones fueron todas muy beneficiosas, ya que todos los integrantes del grupo estaban muy comprometidos en la calidad del producto, y la participación de las reuniones fue el esperado.

Una vez definidos los conceptos principales que se utilizaron en el desarrollo del proyecto, procedemos a comentar la evolución del mismo.

3.3 Implementación

3.3.1 Reunión Inicial

Una vez aprobado el Plan del Proyecto final de carrera, decidimos realizar la Reunión inicial del proyecto, en conjunto, el Equipo de trabajo junto con el Kinesiólogo de cabecera de Biokinetic nos dispusimos a definir/redefinir y negociar los requerimientos que darían inicio al Software Healmate. Los mismos habían sido analizados al momento de presentar el Plan, pero decidimos realizarla para darle formalidad.

Para la reunión preparamos el proyecto presentado en el plan y coordinamos como sería la forma de trabajo, la forma de utilización del software y planteamos cómo iría evolucionando el mismo a la largo del proyecto de desarrollo (explicación básica de la metodología ágil al cliente).

También se prepararon una serie de preguntas para conocer más sobre:

- El proceso e interrelación entre las Obras Sociales, el colegio de Kinesiólogos y el Consultorio
- El ingreso de las Órdenes Médicas y cómo esto generaba el proceso interno dentro del consultorio.
- Los Tratamientos que se interesaban volcar en el sistema y sus prioridades.

En la formalización de los Roles para llevar adelante el proyecto, utilizamos una asignación de Roles dinámica (existen otros roles dependientes de la etapa del proyecto, algunos estarán presentes temporalmente en el proceso y luego desaparecerán), en el cual sólo consideramos importante destacar cuatro, los cuales permanecerán mayormente fijos:

Cliente: **Mauro Milesi**, Kinesiólogo cabecera de Biokinetic.

Equipo de desarrollo: Estará compuesto por todos los integrantes del grupo del Proyecto. **Nelson Inalbon, Víctor Rosset y Matías Milesi**

Tracker: **Víctor Rosset** cumplirá este rol, por la experiencia en integración y el framework utilizado.

Encargado de Pruebas: **Matías Milesi**, por la cercanía con el cliente podrá asistir de forma personal en la realización de las pruebas.

Según la estimación inicial del desarrollo del proyecto, teníamos en claro sobre qué aspectos atacar en el desarrollo, pero con la finalidad de facilitar algunas decisiones, resolvimos realizar la documentación inicial.

3.3.2 Diagramas de Flujo

Los diagramas de Flujos son un mecanismo para describir un proceso. Son muy sencillos y se consideran una buena herramienta para comunicar procesos a personas que no están familiarizados con sistemas. Realizamos en forma conjunta algunos diagramas con el cliente, para facilitar el entendimiento del alcance de algunos aspectos (en la Reunión Inicial) como aproximación inicial.

Asignación de Turnos

Comenzamos el análisis con aspectos simples como la Asignación de Turnos. Si bien el proceso es predecible, la idea era que el cliente agregue algo de valor según su visión del proceso.

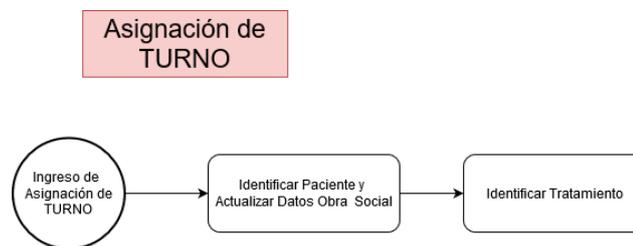


Fig. 3.2.1-01: Diagrama de Flujo de generación de turno

Recurso

Analizamos el proceso de asignar un Recurso, también para formalizar este tipo de Entidad que surgiría y presenta incertidumbre en su detalle esperado. Planteamos las siguientes consultas:

¿Hasta qué punto se gestionarían los recursos dentro de consultorio? ¿Sobre qué aspectos tendrían relación?

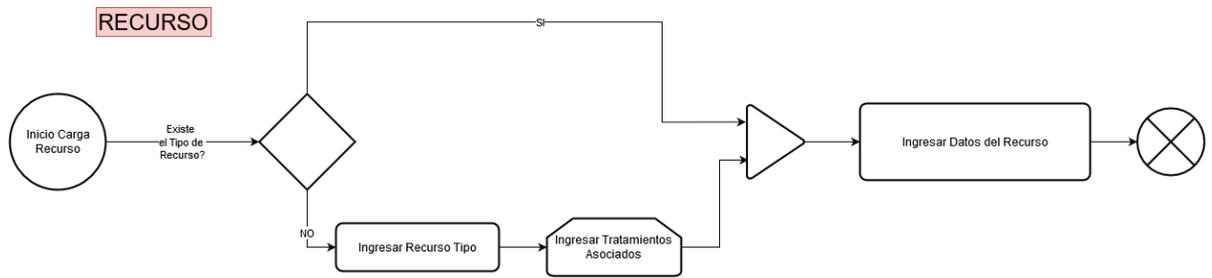


Fig. 3.2.1-02: Diagrama de Flujo de carga de nuevo recurso

En este gráfico, entendimos que los recursos a registrar, están asociados a los Tratamientos, los cuales cumplen su finalidad en una rehabilitación.

Tratamiento

Otra área no menor, era el análisis de los Tratamientos. ¿De qué forma se esperaba realizar una asociación entre una Patología y un Tratamiento?

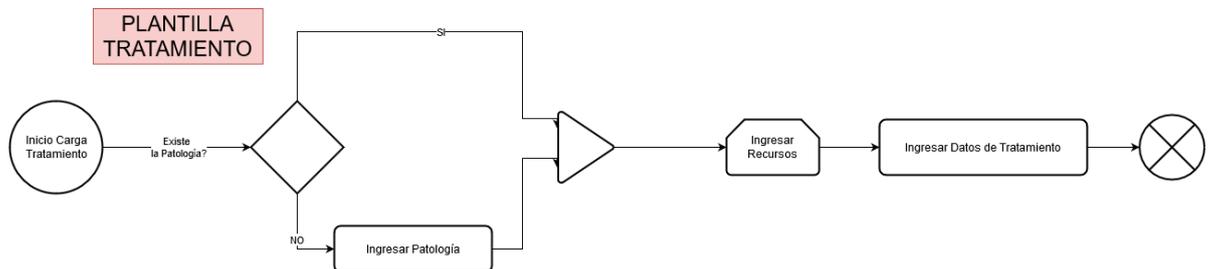


Fig. 3.2.1-03: Diagrama de Flujo de generación de un nuevo Tratamiento

También resultó un proceso muy simple, el cual la asociación era directa, pero entraban en juego los Recursos involucrados en estos Tratamientos. Originalmente, hablamos de asignar recursos sustitutos, los cuales podrían reemplazar a otros en algunos tratamientos especiales, pero la complejidad en su configuración era complicada de explicar de parte del cliente, por lo que se desistió de realizarlo en el proyecto, al menos en su primera versión final.

Tarifa de Servicio (Aranceles)

El Tarifado del Servicio o “Aranceles” también resultó un aspecto a considerar para lograr una buena planificación a lo largo del proyecto. ¿De qué forma se realizaría el

Tarifado? ¿Qué aspectos se deben tener en cuenta para lograr un buen tarifado acompañando a todo el proceso anterior del Consultorio?

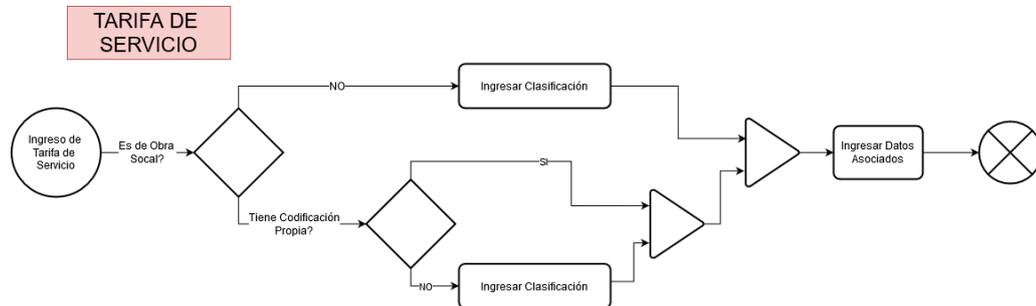


Fig. 3.2.1-04: Diagrama de Flujo de generación de nueva Tarifa

Aquí surgieron aspectos de Codificación y Clasificaciones desconocidas hasta el momento, pero presentaba un circuito no tan complejo.

Como se verá en el desarrollo del proyecto, esta “simplificación” de los procesos llegó a cambiar significativamente en su implementación, gracias a la retroalimentación del cliente al utilizar los entregables. Generamos un estimado de las Entidades y sus relaciones que tendría el proyecto.

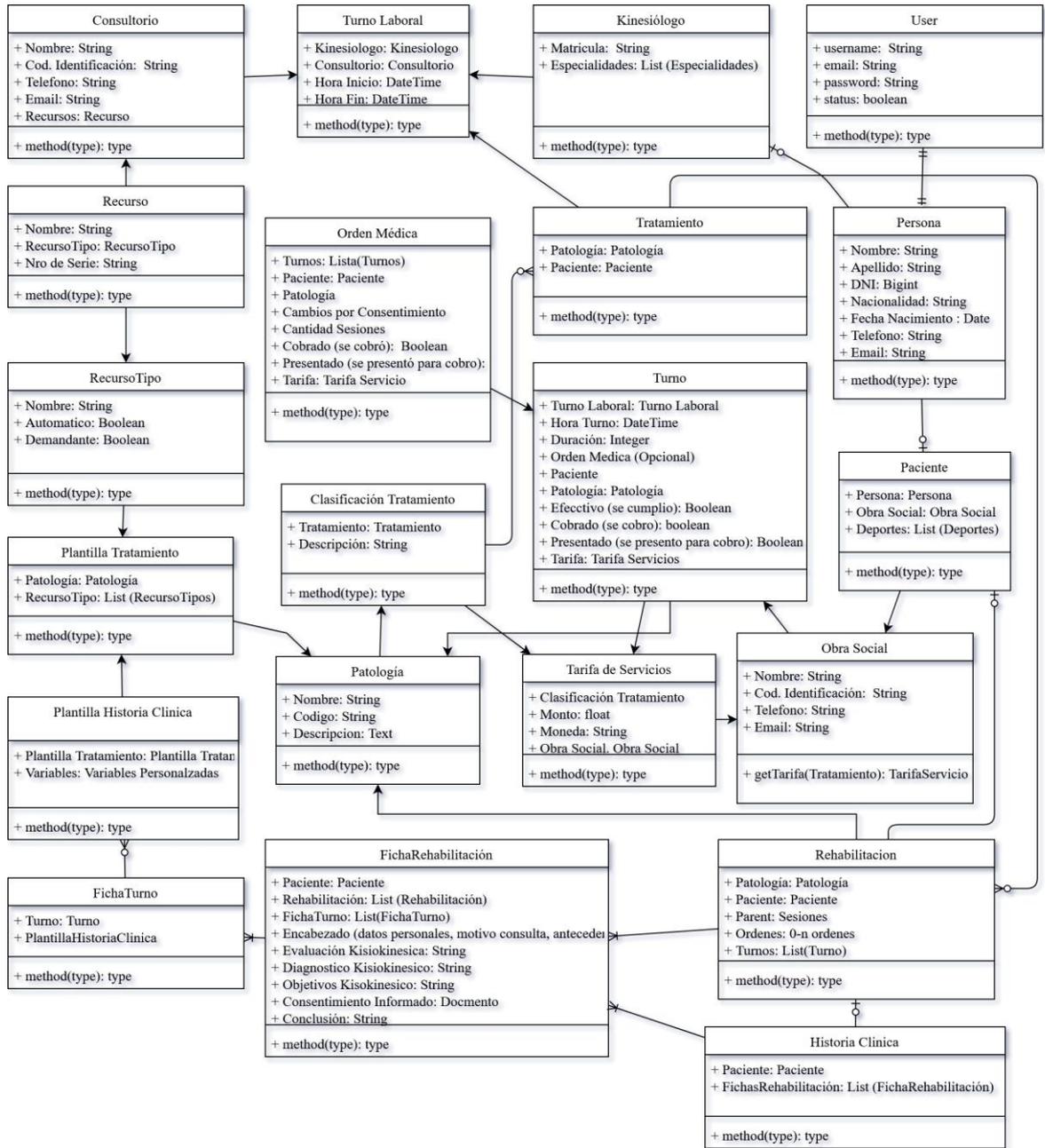


Fig. 3.2.1-05: Diagrama de Entidad – Modelo Inicial

Según este análisis, reacondicionamos el calendario, que difiere en algunos puntos sobre el propuesto en el plan del proyecto.

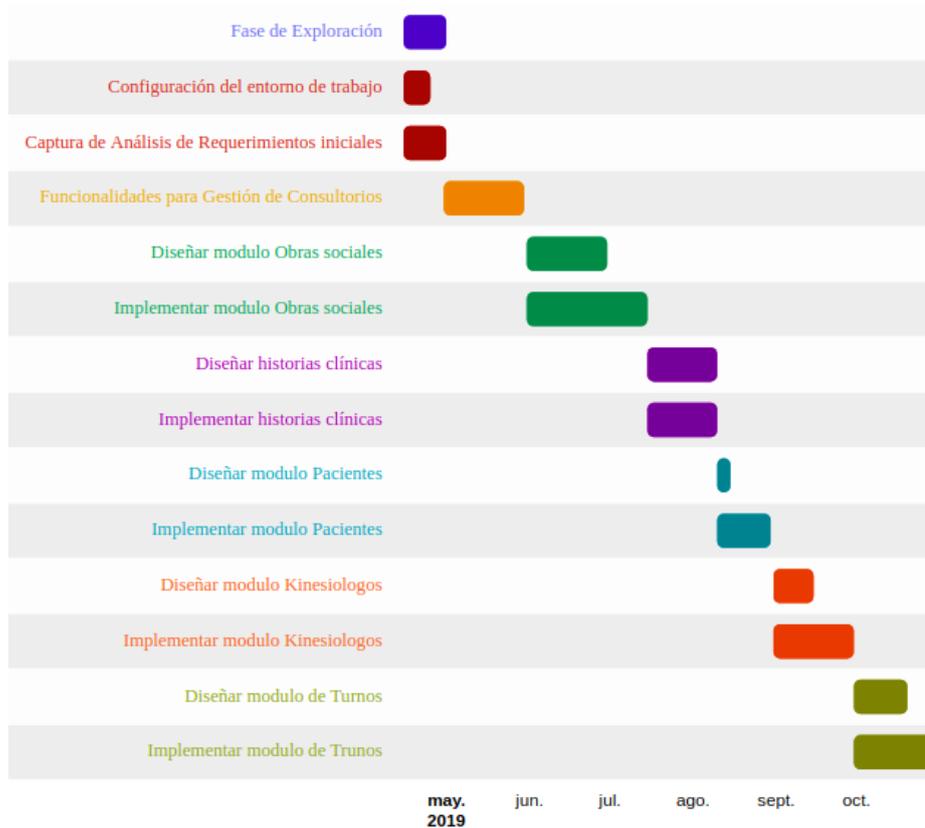


Fig. 3.2.1-06: Plan de desarrollo presentado en el Proyecto Inicial

Se debieron reprogramar las fechas en base a la fecha de comienzo de desarrollo real. Inicio: 10/08/2019 considerando las demás fechas como una sumatoria de días a partir de esa fecha.

3.3.3 Aclaraciones sobre las capturas de Análisis de Requerimientos Iniciales:

El cliente enfatizó fuertemente que el núcleo del sistema estaba en la asignación de los turnos, y que el éxito del proyecto radicaría en la facilidad de crear Turno y asociarlos a los demás módulos del sistema. Lo que provocó que lo planificado para el plan del proyecto cambie sustancialmente. Así que realizamos un primer sprint al sistema deseado para que el cliente pueda probar y solicitar cambios y mejoras desde un inicio, y de allí ir extendiendo el desarrollo. Esto nos marcó el orden de trabajo y cómo deberíamos ir realizando las Historias para lograr un producto de calidad para el cliente. Así también por la metodología elegida, establecimos que un requerimiento no estaría 100% completo con un solo sprint, y

que la retroalimentación y cambios serían constantes. Es decir, las Historias iniciales elegidas, cambiarían y evolucionarían con el uso diario del Software.

También fue importante luego del análisis de la primera reunión, el hecho de verificar que el modelo de trabajo planteado cumpliría las expectativas del cliente y del grupo de desarrollo.

Para ver un ejemplo de una minuta de Reunión, ver el Anexo 8.3 - “Ejemplo de Minuta de Reunión”.

3.4 Sprints

3.4.1 Primer sprint

Inicio: 10/08/2019 - Fin: 7/09/2019

En esta etapa se cumplieron los objetivos planteados. Las Historias sobre las cuales se trabajó en este sprint se detallan en la Tabla 3.4.1-01.

Sprint 1ro			
<i>Nº de Historia</i>	<i>STORYS</i>	<i>PRIORIDAD</i>	<i>Story Points</i>
1	Generación de Entorno de trabajo	ALTA	2
2	Generación de Entidades	ALTA	2
3	Generación de Base de Datos	ALTA	2
4	Montaje Servidor de Producción	MEDIA	4
TOTAL			12

Tabla 3.4.1-01: Historias de Usuario, Primer sprint

Se trabajó fuertemente en investigar los puntos de las herramientas desconocidas, que podrían darnos ventaja a la hora de automatizar procesos en el desarrollo, para facilitar el trabajo y realizar un buen sprint inicial de tiempo con esta etapa inicial.

Las fases de Exploración y Configuración del entorno de trabajo, acontecieron sin grandes inconvenientes según el Gráfico 2.3.1. Podrá ver los detalles sobre las herramientas usadas y su configuración en el Anexo 8.7

Se definieron las Entidades que se estimaron necesarias para el desarrollo del proyecto, pensando él mismo como algo que cambiaría con el paso del tiempo. Para ello, generamos

las entidades según el Diagrama, para poder comenzar con un desarrollo integral del proyecto desde el primer sprint.

Se generó la base de datos inicial acorde a las entidades elegidas, también se realizaron los fixtures iniciales⁷.

Se montó un servidor de Producción donde se instaló la plataforma completa y fue puesta a disposición del equipo de trabajo. También fue habilitado el acceso para dar servicio al cliente, progresivamente, a medida que se fueran entregando los incrementos del proyecto.

No teníamos del todo claro aún la forma de trabajo al finalizar esta etapa, nos conocíamos en nuestras debilidades y fortalezas individuales, era un riesgo a la hora de utilizar una metodología Ad hoc pero sabíamos que funciona con trabajo en equipo y colaboración, dedicando un poco más del tiempo estimado en esta primera etapa.

Con un pequeño sobreesfuerzo, logramos completar la etapa y llegar a tiempo para mostrar el esqueleto de lo realizado. De esto también aprendimos que el tiempo productivo real no era como el que habíamos planificado, ya que se pedía media hora de trabajo neto.

Se trabajó principalmente reunidos en un espacio físico común, para coordinar y aprender de las habilidades de todo el grupo de desarrollo, de forma tener un seguimiento diario de todos los problemas que surgieran y trabajamos fuertemente en equipo, intercambiando PC, revisando código, y criticando objetivamente el desarrollo de nuestros compañeros, para establecer la calidad deseada desde un comienzo.

Backlog del sprint:

STORY	TAREA	Importancia	Task Points	Story Points
Generación de Entorno de trabajo				2
	Generación entorno de desarrollo	ALTA	8	
	Generación entorno de testing	MEDIA	7	
	Configuración de recursos	ALTA	6	
Generación de Entidades				2
	Identificación de entidades	MEDIA	4	
	Definición de atributos	ALTA	7	

⁷Las fixtures se utilizan para cargar una serie de datos (fake/o reales) en nuestra aplicación web para que estén disponibles cuando generemos un nuevo entorno

	Definición de relaciones	BAJA	7	
Generación de Base de Datos				2
	Definición de recursos	ALTA	7	
	Configuración de base de datos	ALTA	8	
Montaje del Servidor de Producción				4
	Instalación de software	ALTA	8	
	Configuración de parámetros	MEDIA	4	
	Configuración de red	MEDIA	4	
	Integración con entorno de trabajo	ALTA	7	
TOTALES			77	10

Tabla 3.4.1-02: Backlog del primer sprint

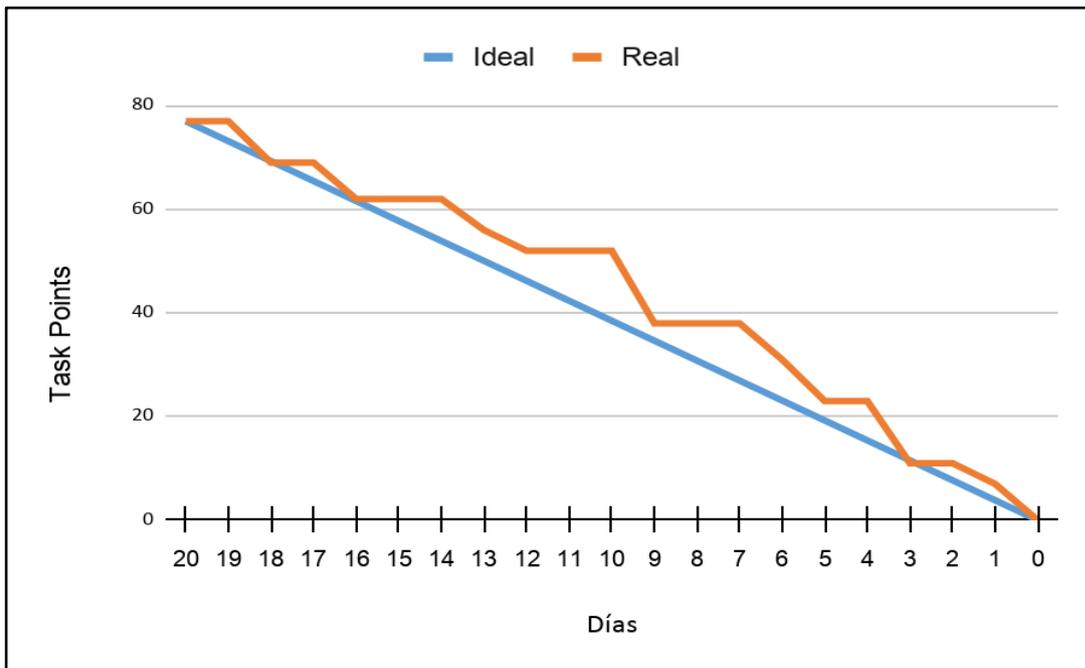


Gráfico 3.4.1-04: Burndown del primer sprint

3.4.2 Segundo sprint

Inicio: 07/09/2019 - Fin: 05/10/2019

En esta etapa se cumplieron los objetivos planteados. Las Historias sobre las cuales se trabajó en este sprint se detallan en la Tabla 3.4.2-01.

Sprint 2do

<i>Nº de Historia</i>	<i>STORYS</i>	<i>PRIORIDAD</i>	<i>Story Points</i>
5	Modelo Básico de Interfaz	ALTA	4
6	ABM de Consultorios	ALTA	3
7	ABM de Usuarios	MEDIA	4
TOTAL			11

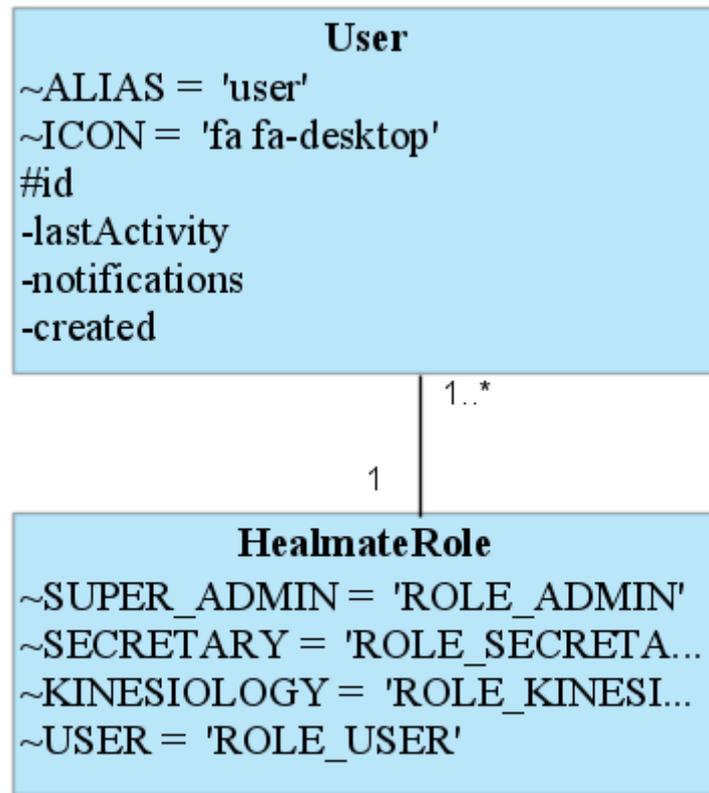
Tabla 3.4.2-01: Historias de Usuario, Segundo sprint

Gracias a la primera sprint, pudimos realizar un estimativo mejorado para este segundo sprint, ya que los horarios productivos de desarrollos eran menores a lo que habíamos estimado, disminuyendo el tiempo disponible por semana.

Se trabajó en el Diseño de la Interfaz de Usuario, para presentar los datos de forma ordenada y elegante, estableciendo y configurando la Barra Lateral y la Barra Superior, donde muestran las notificaciones y el usuario logeado.

Se definió aquí la iconografía y aspectos de diseños que consideramos adecuados para el proyecto para que el cliente pueda verificar y criticar.

Como todas las Entidades definidas en el proyecto, las mismas cambiarán a medida que transcurra el desarrollo del proyecto. El mejor ejemplo es la entidad Usuario, el cual en ese sprint cambió sustancialmente al modelo inicial y luego volverá a cambiar en los sprints futuros cuando se le asocien las Notificaciones (en el siguiente sprint).



Img.3.4.2-01 Entidad de Usuario

La cantidad de Roles y permisos no fue numerosa. Los casos necesarios fueron: Admin, Usuario, Kinesiólogo y Secretaria.

Como regla general, utilizamos para todas las Entidades generadas un Alias y un Icono, los cuales serán tomados en la capa de presentación para lograr uniformidad en las vistas de los usuarios.

Ejemplo: *const ALIAS = 'user'; const ICON = 'fa fa-desktop';*

Trabajamos luego, con reuniones donde el Cliente estaba presente para implementar la Interfaz de Usuario, entre muchas de los parámetros definidos, luego de varias propuestas y feedback.

Seguidamente, generamos los archivos de traducción, los cuales tienen como objetivo traducir las palabras “marcadas” al idioma que detecte el Navegador. Como método inicial, se tradujo todo lo estándar al español.

En esta instancia comenzamos también la generación del Manual de Usuario. El mismo sería actualizado en todos los sprints (manual de usuario Adjunto como documento extra).

Otro puntapié inicial fue la generación de las pruebas automatizadas de todo lo necesario para el control continuo del desarrollo del proyecto. Para ello generamos un modelo de Casos de Pruebas. Luego de resumir los sprints se explicará en detalle las pruebas realizadas y automatizadas, así como las pruebas de aceptación de cada sprint. Las Pruebas, se enumeran al final de este capítulo y sus detalles están en el Anexo.

Conjuntamente con el Cliente, diseñamos la primera interfaz de ABM del sistema que sería la base para todas las demás. Luego trabajamos con el ABM de Usuario y el ABM de Consultorio.

En esta parte del proyecto, fue cuando dedicamos un esfuerzo extra al orden y la forma de trabajo, porque definimos las primeras APIs de búsqueda y eventos. Es importante definir una metodología consistente para facilitar el trabajo en equipo y el mantenimiento del sistema.

El trabajo en un lugar concurrente, con métodos de programación en equipos y alta colaboración inicial, permitió que el equipo de trabajo facilite el entendimiento en épocas en las cuales no podíamos reunirnos por disponibilidad horaria.

Backlog del sprint:

<i>STORY</i>	<i>TAREA</i>	<i>Importancia</i>	<i>Task Points</i>	<i>Story Points</i>
Modelo Básico de Interfaz				4
	Loguin de usuario	ALTA	8	
	Configuración Layout	MEDIA	4	
	Configuración de Elementos de Interfaz de Usuario	ALTA	7	
	Configuración de Formularios	ALTA	7	
	Configuración de Notificaciones	MEDIA	4	
	Configuración de Calendario	MEDIA	5	
	Configuración de Tablas	BAJA	3	
ABM de Consultorios				3
	Alta de Consultorios	ALTA	7	
	Baja de Consultorios	BAJA	2	
	Modificación de Consultorios	MEDIA	4	

	Mostrar Consultorio	MEDIA	4	
	Búsqueda de Consultorios	MEDIA	6	
	Interfaz de Consultorios	MEDIA	5	
	Testing del Módulo de Consultorios	ALTA	7	
ABM de Usuarios				4
	Alta de Usuarios	ALTA	8	
	Modificación de Usuarios	MEDIA	5	
	Baja de Usuarios	BAJA	3	
	Búsqueda de Usuarios	MEDIA	6	
	Mostrar Usuario	MEDIA	5	
	Diseño de Interfaz	MEDIA	5	
	Testing del Módulo de Usuarios	ALTA	7	
TOTALES			112	11

Tabla 3.4.2-02: Backlog del segundo sprint

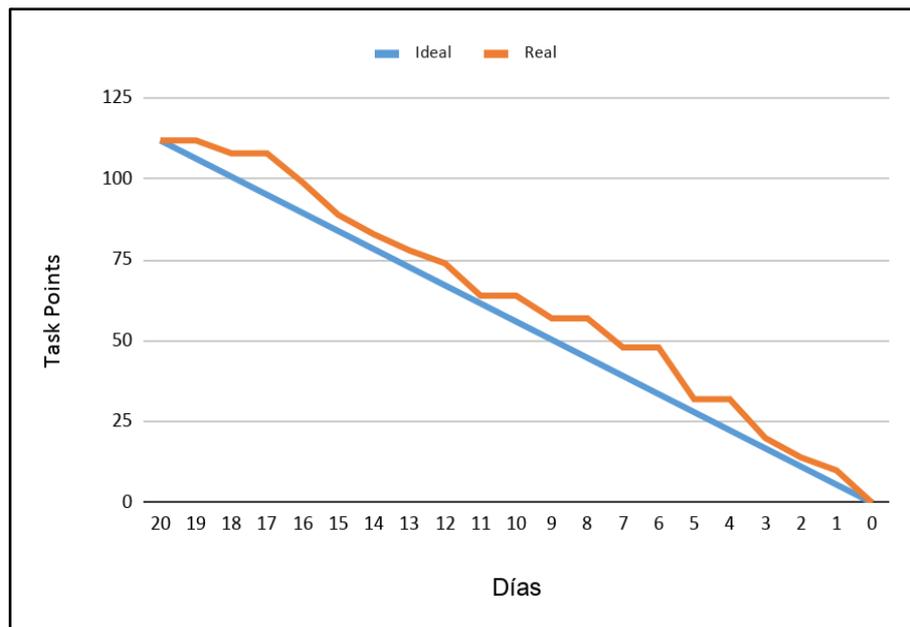


Gráfico 3.4.2-01: Burndown del segundo sprint

3.4.3 Tercer sprint:

Inicio: 05/10/2019 - Fin: 02/11/2019

En esta etapa cumplimos con los objetivos planteados, teniendo en cuenta que el Segundo sprint la estimación había sido certera, consideramos los mismos criterios a la hora de evaluar el alcance de esta etapa. Las Historias sobre las cuales se trabajó en este sprint se detallan en la Tabla 3.4.3-01.

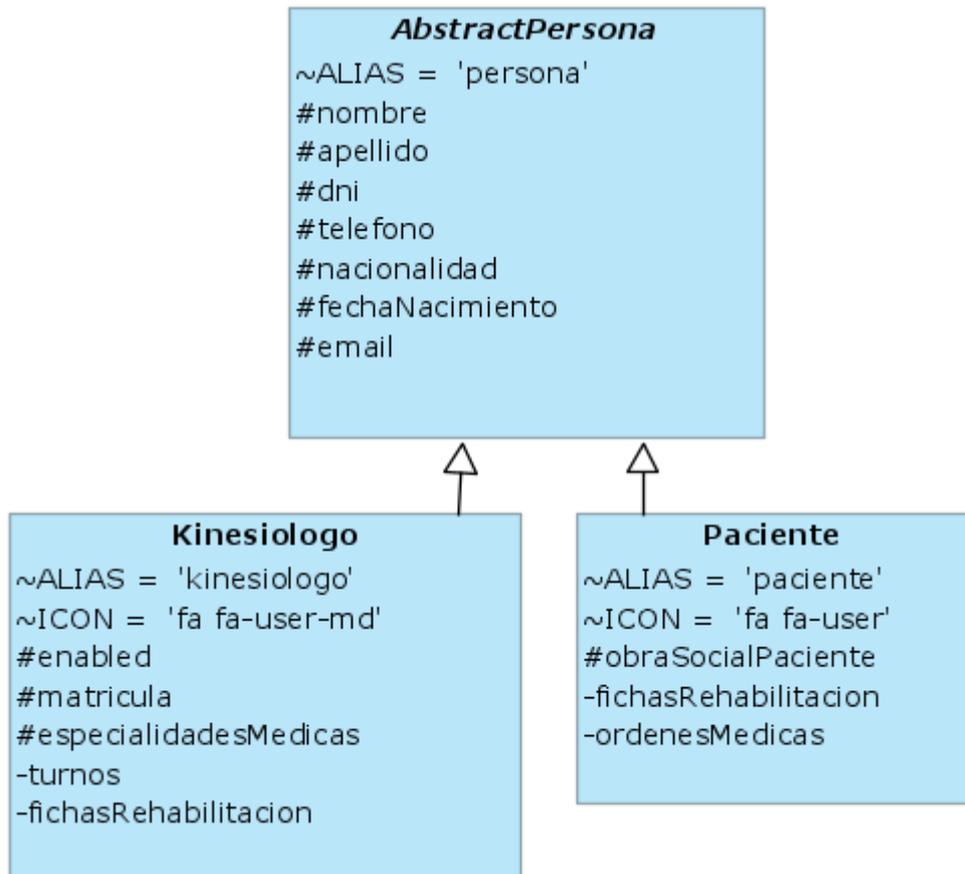
Sprint 3ro			
<i>Nº de Historia</i>	<i>STORYS</i>	<i>PRIORIDAD</i>	<i>Story Points</i>
8	ABM de Pacientes	MEDIA	2
9	Gestión de Turnos Rápidos	ALTA	4
10	ABM de Especialidades Médicas	MEDIA	2
11	ABM de Kinesiólogos	MEDIA	2
12	ABM de Patología	MEDIA	2
TOTAL			11

Tabla 3.4.3-01: Historias de Usuario, Tercera Sprint

Según las reuniones iniciales y las necesidades del cliente, al llegar a esta etapa consideramos importante lograr un incremento donde se brinde la posibilidad de generar turnos en el sistema aportando una muy valiosa funcionalidad en cuanto a lo administrativo de la entidad. Esto fue acompañado de un aumento del esfuerzo del equipo que debió trabajar concurrentemente y realizando reuniones diarias, en labor conjunta, para alcanzar la calidad necesaria en tan importante módulo.

La funcionalidad de Turnos Rápidos fue complementada con las de ABM de Pacientes, Kinesiólogos y Patologías sin imprevistos en el camino. El proceso de creación del turno fue detallado en el Diagrama de Flujo de la Reunión Inicial.

Según el diseño inicial, actualizamos las entidades de los pacientes y kinesiólogos. El siguiente diagrama muestra las diferentes definiciones de las Entidades, algunos atributos fueron agregados en los siguientes sprints:



Img.3.4.3-01 Herencia de Persona – Kinesiólogo y Paciente

Backlog del sprint:

<i>STORY</i>	<i>TAREA</i>	<i>Importancia</i>	<i>Task Points</i>	<i>Story Points</i>
ABM de Pacientes				2
	Alta de Pacientes	ALTA	7	
	Baja de Pacientes	BAJA	2	
	Modificación de Pacientes	MEDIA	4	
	Mostrar Paciente	MEDIA	4	
	Búsqueda de Pacientes	MEDIA	6	
	Interfaz de Pacientes	MEDIA	5	
	Testing del Módulo de Pacientes	ALTA	7	
Gestión de Turnos Rápidos				3
	Alta de Turnos rápidos	ALTA	6	
	Búsqueda de Turnos	BAJA	3	
	Interfaz de Turnos Rápidos	MEDIA	4	
ABM de Especialidades Médicas				2
	Alta de Especialidades Médicas.	MEDIA	5	

	Modificación de Especialidades Médicas.	MEDIA	4	
	Baja de Especialidades Médicas	BAJA	2	
	Búsqueda de Especialidades Médicas.	BAJA	2	
	Mostrar Especialidad Médicas.	BAJA	2	
	Interfaz de Especialidad Médicas.	MEDIA	4	
ABM de Kinesiólogos				2
	Alta de Kinesiólogos	ALTA	7	
	Baja de Kinesiólogos	BAJA	2	
	Modificación de Kinesiólogos	MEDIA	4	
	Mostrar Kinesiólogo	MEDIA	4	
	Búsqueda de Kinesiólogos	MEDIA	6	
	Interfaz de Kinesiólogo	MEDIA	5	
	Testing del Módulo de Kinesiólogo	ALTA	7	
ABM de Patología				2
	Alta de Patologías	MEDIA	5	
	Baja de Patologías	BAJA	2	
	Modificación de Patologías	MEDIA	4	
	Mostrar Patología	MEDIA	4	
	Búsqueda de Patologías	BAJA	3	
	Interfaz de Patologías	MEDIA	4	
	Testing del Módulo de Patologías	MEDIA	6	
TOTALES			130	11

Tabla 3.4.3-02: Backlog del tercer sprint

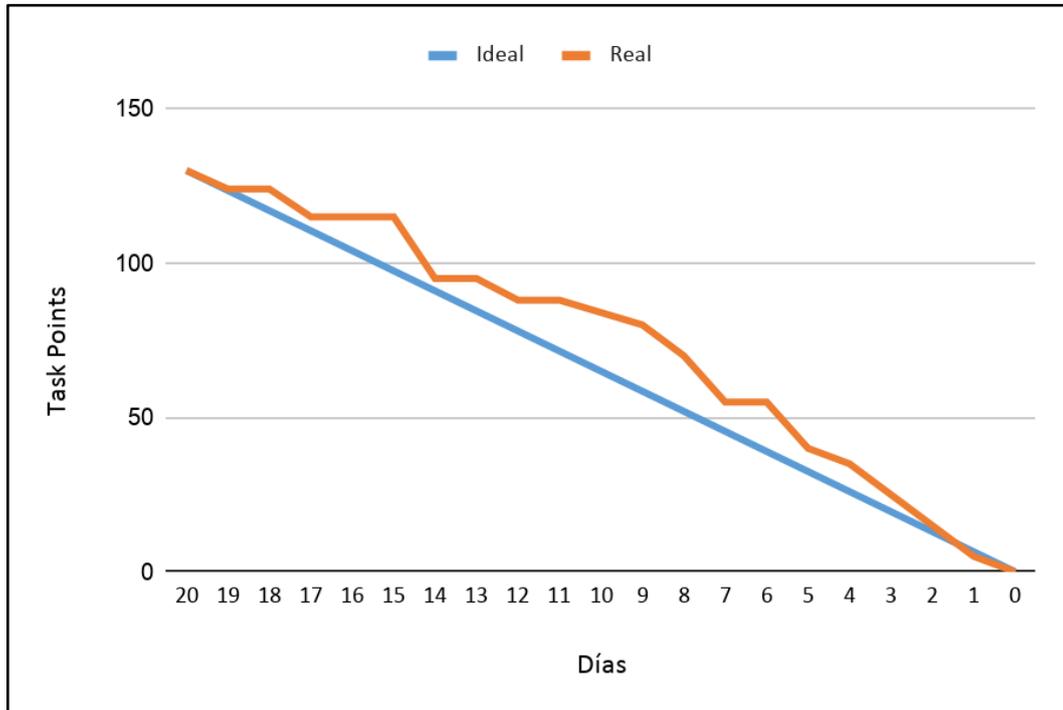


Gráfico 3.4.3-01: Burndown del tercer sprint

3.4.4 Cuarto sprint

Inicio: 2/11/2019 - Fin: 07/01/2020

En esta etapa se cumplieron las historias acordadas, pero no según los tiempos, ya que a dos de los integrantes del grupo le disminuyó drásticamente la capacidad horaria para dedicar al proyecto. Aquí teníamos dos opciones, prolongar el fin del sprint o dividir la entrega en la mitad para mantener la continuidad de las entregas. Luego de consensuar con el Clientes, decidimos prolongar el sprint hasta poder cumplir las horas planificadas en el desarrollo de forma de verificar si las horas estimadas cumplían con las reales.

Las Historias sobre las cuales se trabajó en este sprint se detallan en la Tabla 3.4.4-01.

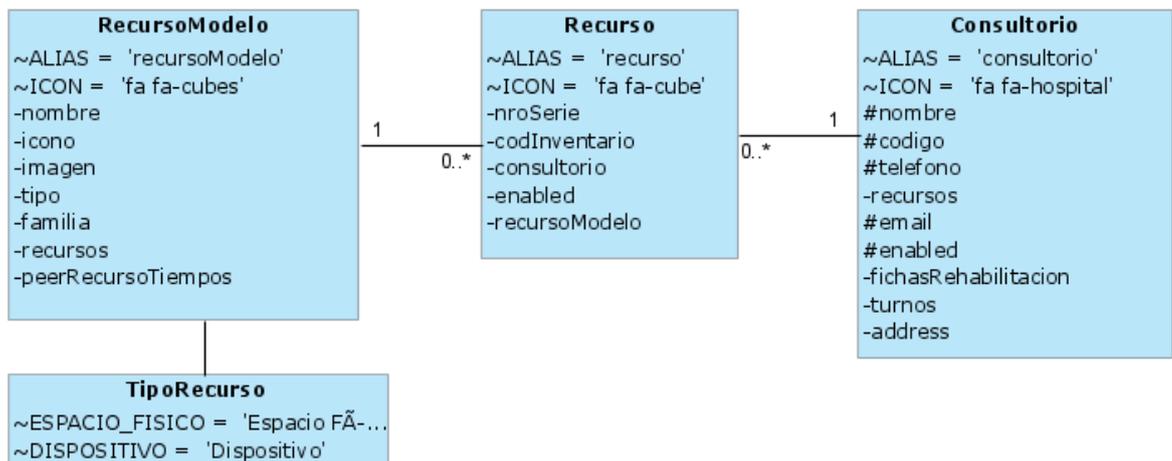
Sprint 4to			
<i>Nº de Historia</i>	<i>STORYS</i>	<i>PRIORIDAD</i>	<i>Story Points</i>
13	ABM Tipo de Recursos	MEDIA	2
14	ABM de Recursos	MEDIA	2
15	ABM de Obras Sociales	ALTA	4
16	ABM de Turnos	ALTA	4

TOTAL	12
--------------	-----------

Tabla 3.4.4-01: Historias de Usuario, Cuarto sprint

Teniendo en cuenta que en el tercer sprint la estimación había sido certera, consideramos los mismos criterios a la hora de evaluar el alcance de esta etapa, teniendo en cuenta los plazos en horas dedicadas. Al comienzo del *sprint* había incertidumbre en relación a cuánto tiempo disponíamos como grupo para avanzar con el proyecto, por lo que decidimos trabajar con una cantidad de horas mínimas semanales para garantizar el cumplimiento del plazo.

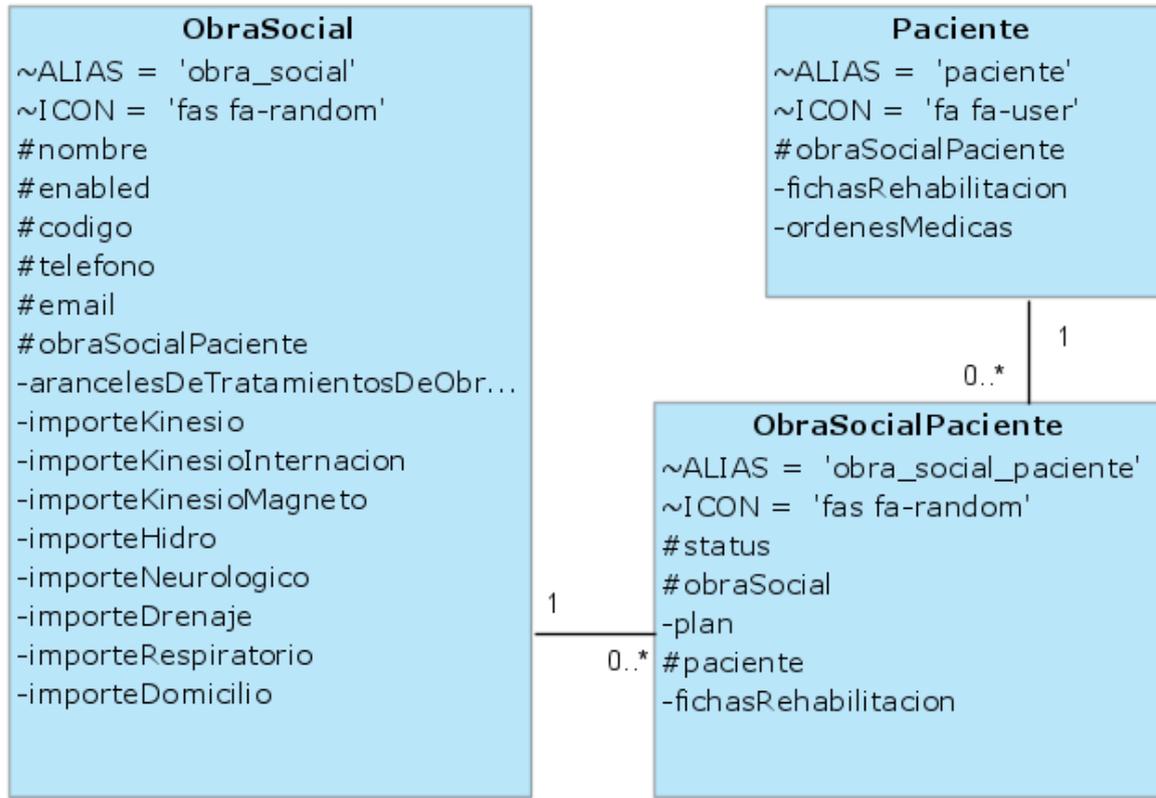
Según el Diagrama de Flujo de Recurso de la Reunión Inicial, la asociación entre un Tipo de Recurso, Recurso y Consultorio se explican en el siguiente Diagrama:



Img. 3.4.4-01: Relación entre Recurso y Consultorio

En esta etapa se trabajó sobre ABM de Tipos de Recursos, ABM de Recurso, ABM de Obras Sociales y se cerró el ABM de Turnos, que se había planteado en dos etapas. El ABM de Turnos sería un punto que sufriría muchas modificaciones en todo el proyecto, y sería un punto fuerte al finalizar el mismo, ya que se trabajó en él no sólo en este sprint y en la anterior, sino en todas las posteriores, mejorándolo gracias al feedback del cliente (a pesar de haber pasado las pruebas de Aceptación, las mejoras surgieron con el uso).

La asociación entre los Pacientes y las Obras Sociales también fue desarrollada. Los pacientes podrían tener muchas Obras Sociales y los mismos estarían asociados en el sistema a través de un Plan. El resultado fue el siguiente:



Img. 3.4.4-02: Relación entre Paciente y Obra Social

Burndown del sprint:

STORY	TAREA	Importancia	Task Points	Story Points
ABM Tipo de Recursos				2
	Alta de Tipo de Recursos	MEDIA	4	
	Baja de Tipo de Recursos	BAJA	2	
	Modificación de Tipo de Recursos	MEDIA	4	
	Búsqueda de Tipo de Recursos	MEDIA	4	
	Mostrar Tipo de Recurso	BAJA	2	
	Interfaz de Tipo de Recursos	MEDIA	4	
ABM de Recursos				2
	Alta de Recursos	MEDIA	6	
	Baja de Recursos	BAJA	2	
	Modificación de Recursos	MEDIA	4	
	Búsqueda de Recurso	MEDIA	4	
	Mostrar Recurso	BAJA	2	
	Interfaz de Recurso	MEDIA	4	
	Testing de Módulo Recursos	MEDIA	6	
ABM de Obras Sociales				4

	Alta de Obras Sociales.	ALTA	7	
	Baja de Obras Sociales.	BAJA	2	
	Modificación de Obras Sociales.	MEDIA	4	
	Mostrar Obras Sociales.	MEDIA	4	
	Búsqueda de Obras Sociales.	MEDIA	6	
	Interfaz de Obras Sociales.	MEDIA	5	
	Testing del Módulo de Obras Sociales	ALTA	7	
	Asociar Obra Social Paciente	MEDIA	6	
ABM de Turnos				4
	Alta de Turnos	ALTA	7	
	Baja de Turnos	BAJA	2	
	Modificación de Turnos	MEDIA	5	
	Mostrar Turno	MEDIA	4	
	Búsqueda de Turnos	MEDIA	6	
	Adecuar la interfaz de turnos	MEDIA	6	
	Testing del Módulo de Turnos	ALTA	7	
	Disponibilidad de Recursos	MEDIA	6	
TOTALES			132	12

Tabla 3.4.4-02: Backlog del Cuarto sprint

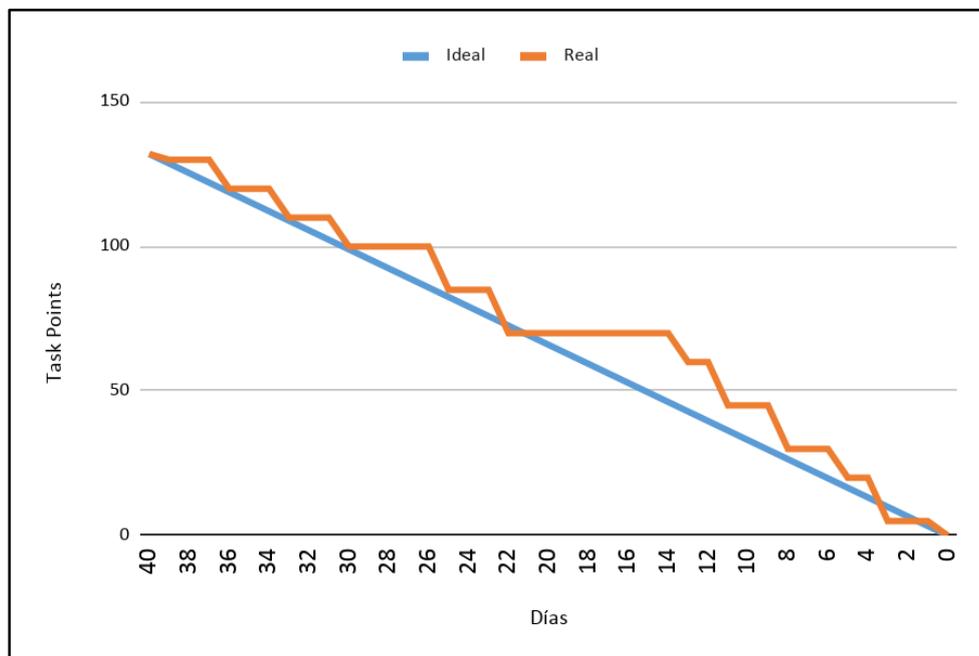


Gráfico 3.4.4-01: Burndown del cuarto sprint

3.4.5 Quinto sprint:

Inicio: 07/01/2020 - Fin: 08/02/2020

En esta etapa cumplimos con los objetivos planteados, ya que la disponibilidad horaria del equipo de desarrollo se había normalizado. Las Historias sobre las cuales se trabajó en este sprint se detallan en la Tabla 3.4.5-01.

Sprint 5to			
Nº de Historia	STORYS	PRIORIDAD	Story Points
17	ABM de Ordenes Médicas	ALTA	3
18	ABM Tratamientos	ALTA	3
19	Gestión por Calendario	MEDIA	2
20	Notificaciones y Mensajes	BAJA	2
TOTAL			10

Tabla 3.4.5-01: Historias de Usuario, Quinto sprint

La estimación resultó correcta.

Al ser procesos muy importantes para el cliente, la mayor energía debió centrarse en entender la gestión de la Orden Médica, dejando el Envío y Cobranzas a realizar en los sprints posteriores.

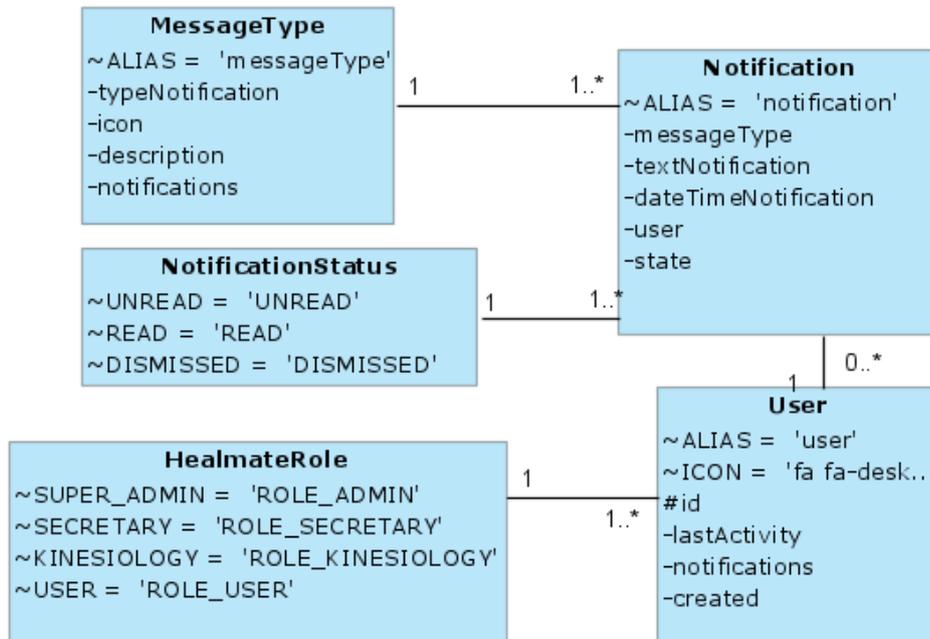
Un punto significativo fue la vista del Calendario, ya que sería una pieza importante en el software porque sería utilizado frecuentemente en la Interfaz. Se encontraron dos aspectos necesarios en cuanto a la vista del Calendario, tanto para la organización del Kinesiólogo como para la administración de todos los turnos del consultorio.

Éste fue uno de los sprints más satisfactorios, ya que se logró cumplir el objetivo con una interfaz y funcionamiento que fue del agrado del cliente.

También planteamos nuevamente el funcionamiento deseado para el análisis de los Recursos, mejorando el proceso para lograr más transparencia.

El modelo planteado para las Notificaciones y Mensajes fue muy simple. Las Notificaciones tienen Tipos de Mensajes, los cuales tienen su iconografía y colores

personalizados. Las Notificaciones tienen 3 estados permitidos. A continuación, se detalla el diagrama de Entidad.



Img. 3.4.5-01: Tipo de Mensaje y Notificación

Backlog del sprint:

STORY	TAREA	Importancia	Task Points	Story Points
ABM de Ordenes Médicas				3
	Alta de Ordenes Médicas.	ALTA	7	
	Baja de Ordenes Médicas.	BAJA	2	
	Modificación de Ordenes Médicas.	MEDIA	4	
	Mostrar Ordenes Médicas.	MEDIA	4	
	Búsqueda de Ordenes Médicas.	MEDIA	6	
	Interfaz de Ordenes Médicas.	MEDIA	5	
	Testing del Módulo de Ordenes Médicas.	ALTA	7	
ABM Tratamientos				4
	Alta de Tratamientos	ALTA	8	
	Baja de Tratamientos	ALTA	7	
	Modificación de Tratamientos	MEDIA	4	
	Mostrar Tratamiento	MEDIA	5	
	Buscar Tratamientos	MEDIA	6	
	Interfaz de Tratamientos	MEDIA	5	
	Testing del módulo de Tratamientos	ALTA	7	

Gestión por Calendarios				2
	Interfaz del Calendario	MEDIA	4	
	Mostrar Turnos reservados	MEDIA	4	
	Selección de fecha de turnos por calendario	ALTA	7	
Notificaciones y Mensajes				2
	Alta de Notificación	MEDIA	4	
	Baja de Notificación	BAJA	2	
	Modificación de Notificación	MEDIA	4	
	Ver Notificación	MEDIA	3	
	Diseño de Interfaz	MEDIA	4	
	Notificaciones generadas automáticamente	ALTA	7	
	Testing del Módulo de Notificaciones	ALTA	6	
TOTALES			122	11

Tabla 3.4.5-02: Backlog del quinto sprint

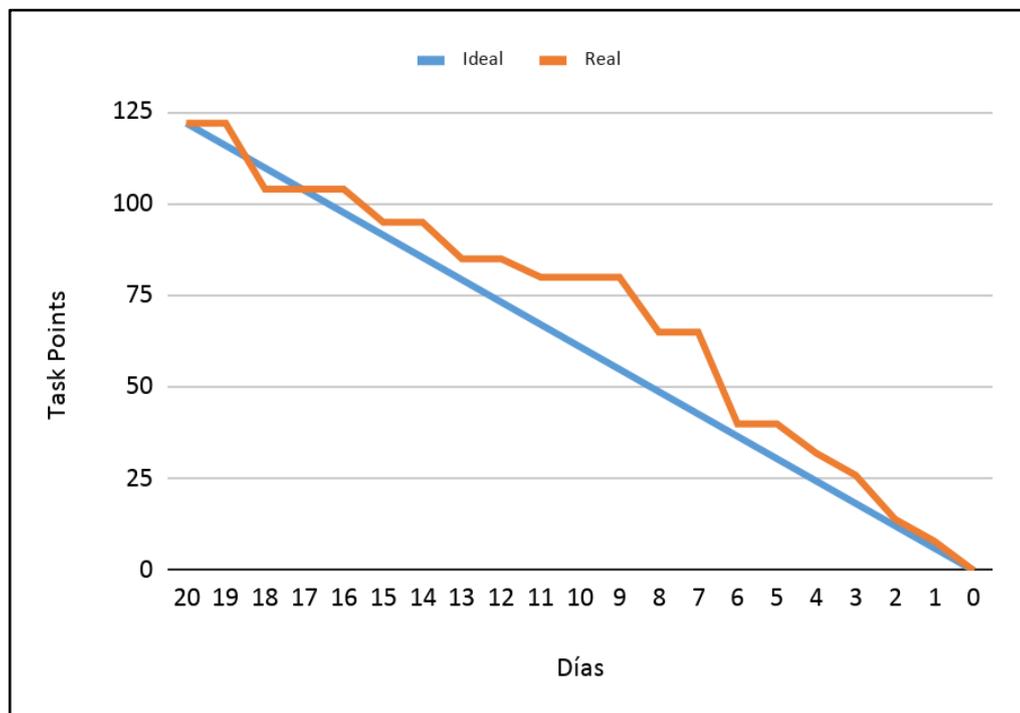


Gráfico 3.4.5-01: Burndown del quinto sprint

3.4.6 Sexto sprint:

Inicio: 08/02/2020 - Fin: 07/03/2020

En esta etapa cumplimos con los objetivos planteados. Las Historias sobre las cuales se trabajó en este sprint se detallan en la Tabla 3.2-11.

Sprint 6to			
<i>Nº de Historia</i>	<i>STORYS</i>	<i>PRIORIDAD</i>	<i>Story Points</i>
21	Gestión de Aranceles	BAJA	1
22	ABM Sesiones de Tratamientos	ALTA	3
23	ABM Rehabilitaciones	ALTA	3
24	Procesamiento de O.M.	ALTA	3
TOTAL			10

Tabla 3.4.6-01: Historias de Usuario, Sexto sprint

La estimación resultó certera, siendo el esfuerzo de esta etapa moderado/alto.

A medida que trabajamos con el Arancelado de los Tratamientos por Obra Social, desarrollamos varias alternativas las cuales, luego de reiteradas reuniones y feedback del cliente, eligió *la más práctica a la hora de aplicar*. Cuando definimos el arancelado lo hicimos inicialmente por Plan de Obra Social para cada uno de los tratamientos definidos (lo que hacía poco práctico y tedioso). Acto seguido, el cliente pidió modificarlo a un Genérico por Obra Social con todas las alternativas, permitiendo una rápida elección a la hora del envío del Cobro. Es por eso que esta Gestión resultó costosa para el equipo de desarrollo, obligando a extender las horas diarias dedicadas.

Los procesos destacados en este sprint son el envío de la Orden Médica al Círculo y el Cobro propiamente dicho. Una orden médica debe ser enviada para luego ser cobrada.

El paso de estado a Cobrado es simplemente marcar la orden médica como cobrada, estableciendo la Fecha de cobro para generar así la caja. El control deseado por el cliente es a través de los informes. Sólo guardando los movimientos de cobro y sus fechas será suficiente para el control requerido.

Las Rehabilitaciones son las encargadas de encuadrar una serie de turnos, pudiendo contener varias Órdenes Médicas y sobre estas órdenes se realizan los análisis con las Variables.

Backlog del sprint:

<i>STORY</i>	<i>TAREA</i>	<i>Importancia</i>	<i>Task Points</i>	<i>Story Points</i>
Gestión de Aranceles				2
	Alta de Aranceles de Obras Sociales.	MEDIA	5	
	Modificación de Aranceles de Obras Sociales.	BAJA	3	
	Interfaz de Aranceles de Obras Sociales.	MEDIA	5	
ABM Sesiones de Tratamientos				3
	Alta de Sesión de Tratamientos	ALTA	8	
	Copia de Sesión de Tratamiento	MEDIA	4	
	Baja de Sesión de Tratamientos	BAJA	3	
	Modificación de Sesión de Tratamientos	MEDIA	6	
	Asociación de par Recurso-Tiempo	MEDIA	5	
	Asociar Tratamiento	ALTA	8	
ABM Rehabilitación				3
	Alta de Rehabilitaciones	ALTA	7	
	Baja de Rehabilitaciones	MEDIA	4	
	Modificación de Rehabilitaciones	MEDIA	5	
	Mostrar Rehabilitación	MEDIA	5	
	Buscar Rehabilitaciones	MEDIA	4	
	Diseño de Interfaz	MEDIA	5	
	Testing del Módulo de Historia Clínica	ALTA	7	
Procesamiento de O.M.				2
	Interfaz: Gestión de Proceso de envío y cobro de O.M.	MEDIA	6	
	Gestión de estados de O.M.	ALTA	7	
	Informes de Estados de O.M.	MEDIA	6	
	Testing completo del Módulo de O.M.	MEDIA	4	
TOTALES			107	10

Tabla 3.4.6-02: Backlog del sexto sprint

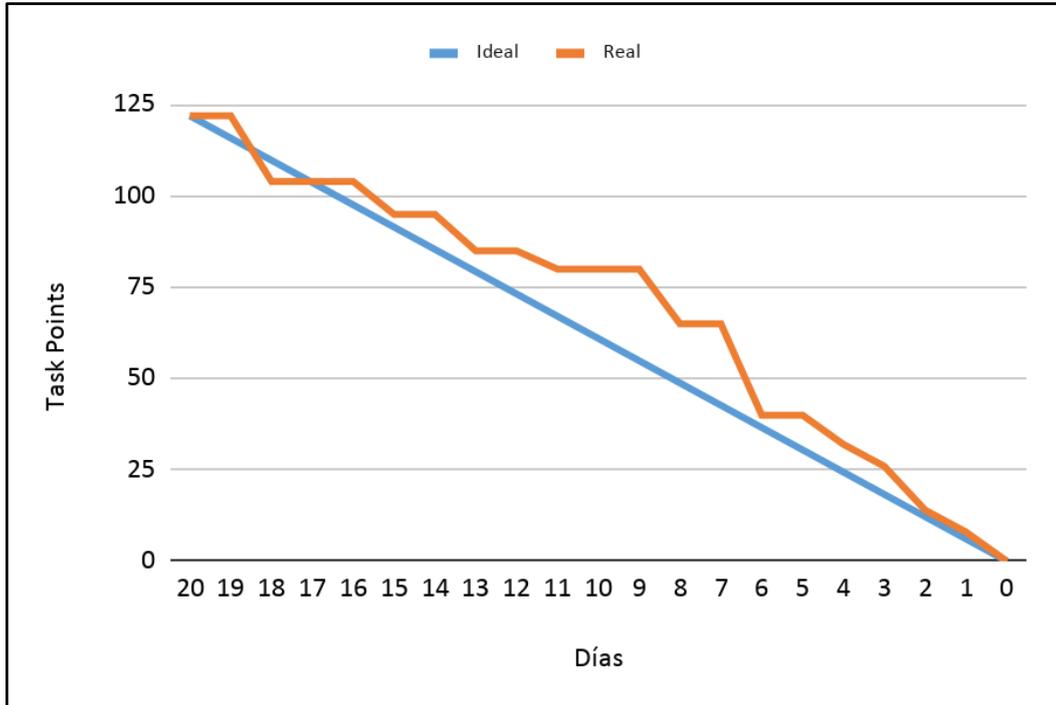


Gráfico 3.4.6-01: Burndown del sexto sprint

3.4.7 Séptimo sprint:

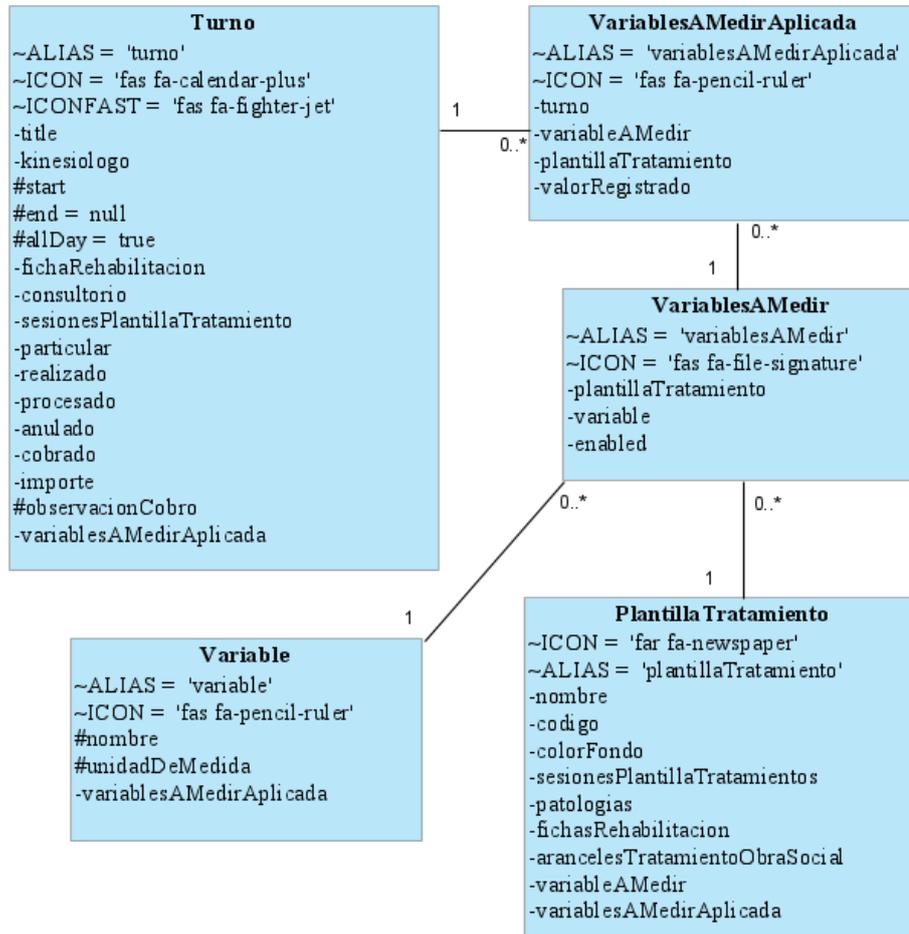
Inicio: 07/03/2020 - Fin: 4/04/2020

En esta etapa cumplimos con los objetivos planteados. Las Historias sobre las cuales se trabajó en este sprint se detallan en la Tabla 3.2-13.

Sprint 7mo			
Nº de Historia	STORYS	PRIORIDAD	Story Points
25	ABM Variables de Seguimiento	ALTA	4
26	Informes: Variables de S. y Turnos	BAJA	2
27	ABM Lugar de Guardado	MEDIA	2
TOTAL			8

Tabla 3.4.7-01: Historias de Usuario, Séptimo sprint

El proceso de registro y generación de Variables de Seguimiento fue muy aproximado al diseño inicial. El diseño final fue simple.



Img. 3.4.7-01: Relación entre Variables y Turnos

Como se observa en el gráfico, las variables están asociadas al Tratamiento. Las mismas se registran en los turnos asociados a dichos Tratamientos. El total de las Variables a Medir Aplicadas permite realizar los diferentes análisis de la evolución de las Rehabilitaciones.

El módulo de informes no tuvo una estimación muy certera (se subestimó), pero con un sobre esfuerzo durante una semana logramos cumplir el tiempo pactado.

Backlog del sprint:

STORY	TAREA	Importancia	Task Points	Story Points
ABM Variables de Seguimiento				4
	Alta de Variables de Seguimiento.	MEDIA	4	
	Baja de Variables de Seguimiento.	BAJA	2	
	Modificación de Variables de Seguimiento.	MEDIA	4	

	Mostrar Variable de Seguimiento.	BAJA	3	
	Buscar Variables de Seguimiento.	MEDIA	4	
	Tratamientos: Asociar Variables de Seguimiento.	MEDIA	4	
	Turnos: carga información de Variables de Seguimiento.	ALTA	7	
	Interfaz de Variables de Seguimiento.	MEDIA	5	
	Testing del Módulo de Variables de Seguimiento	ALTA	7	
Informes: Variables de S.				2
	Generar Informe según Variables de Seguimiento.	ALTA	7	
	Exportar Informes	ALTA	6	
	Testing Módulo informes Variables de Seguimiento.	MEDIA	6	
ABM Lugar de Guardado				2
	Alta de Lugar de Guardado.	MEDIA	5	
	Baja de Lugar de Guardado.	MEDIA	4	
	Modificación de Lugar de Guardado.	MEDIA	4	
	Mostrar Lugar de Guardado.	MEDIA	4	
	Búsqueda de Lugar de Guardado.	MEDIA	5	
	Interfaz de Lugar de Guardado.	MEDIA	5	
	Asociar Orden Médica.	BAJA	2	
TOTALES			88	8

Tabla 3.4.7-02 Backlog del Séptimo sprint

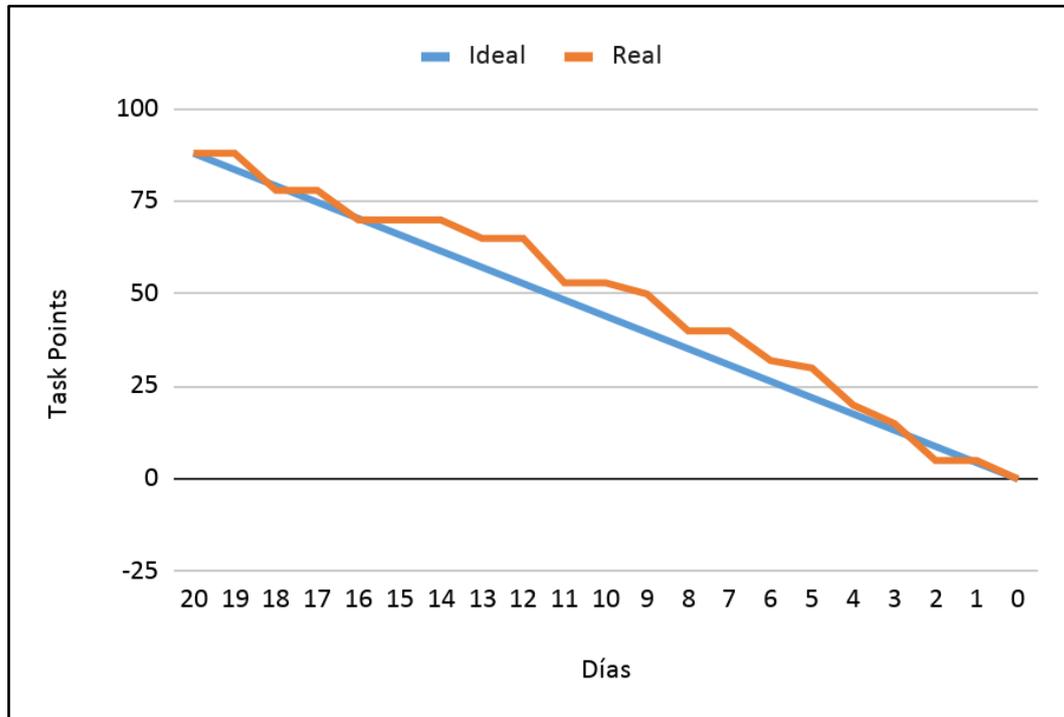


Gráfico 3.4.7-01: Burndown del séptimo sprint

3.4.8 Octavo sprint:

Inicio: 04/04/2020 - Fin: 22/05/2020

En esta etapa cumplimos con los objetivos planteados. Las Historias sobre las cuales se trabajó en este sprint se detallan en la Tabla 3.4.8-01.

Sprint 8va			
Nº de Historia	STORYS	PRIORIDAD	Story Points
28	Informes de Turnos	ALTA	4
29	Diseño de Dashboard	BAJA	3
30	Correcciones y Mejoras	BAJA	3
TOTAL			10

Tabla 3.4.8-01: Historias de Usuario, Octavo sprint

En este sprint realizamos el cierre del proyecto, trabajando en los informes de Turnos requeridos por el Cliente y el Tablero que le brinda información disponible en la página inicial. *También atendimos mejoras significativas sugeridas por el usuario, sobre*

módulos en los cuales trabajamos los sprints anteriores. Dimos prioridad a estas sugerencias que fortalecieron las funcionalidades en base a la experiencia del usuario.

Logramos cumplir los objetivos del sprint y agregamos más valor aún al software, trabajando sobre las mejoras

Decidimos hacer un corte por falta de disponibilidad horaria de todos los integrantes del grupo, dejando lo no realizado para trabajos futuros.

Backlog del sprint:

<i>STORY</i>	<i>TAREA</i>	<i>Importancia</i>	<i>Task Points</i>	<i>Story Points</i>
Informes de Turnos				4
	Generar informe según los Turnos de un Kinesiólogo	ALTA	7	
	Exportar Informes	MEDIA	4	
	Testing del Módulo de Informes	ALTA	5	
Diseño de Dashboard				3
	Generar acceso a funcionalidades cotidianas	ALTA	8	
	Edición de distribución de accesos a funcionalidades	MEDIA	7	
	Interfaz del tablero	MEDIA	5	
Correcciones y Mejoras				3
	Adaptaciones finales	MEDIA	50	
TOTALES			86	10

Tabla 3.4.8-02: Backlog del octavo sprint

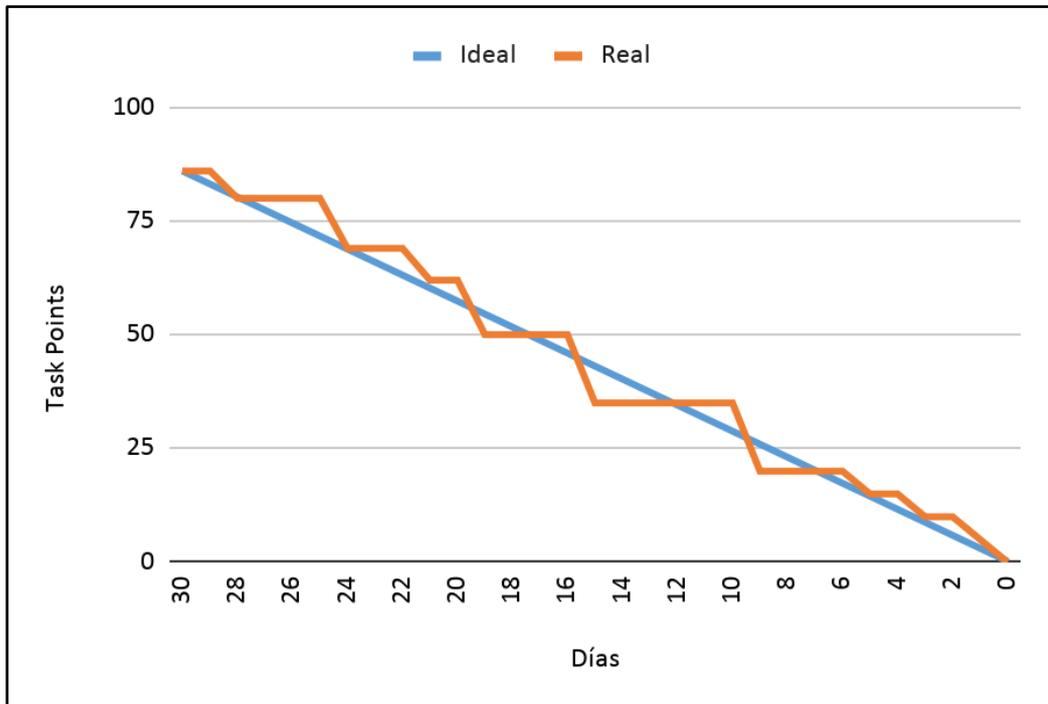
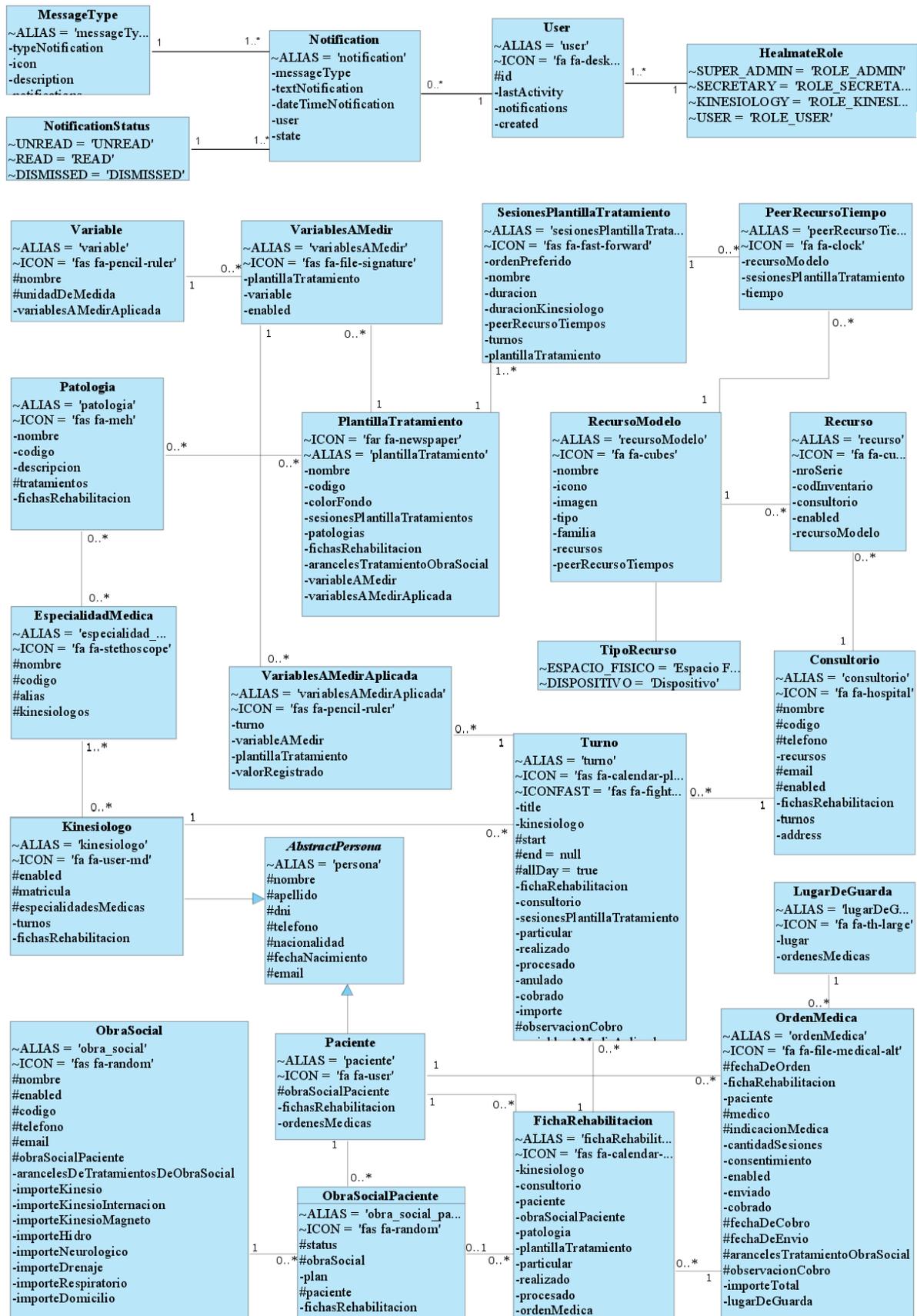


Gráfico 3.4.8-01: Burndown del octavo sprint

El Diagrama final del modelo quedó conformado de la siguiente manera:



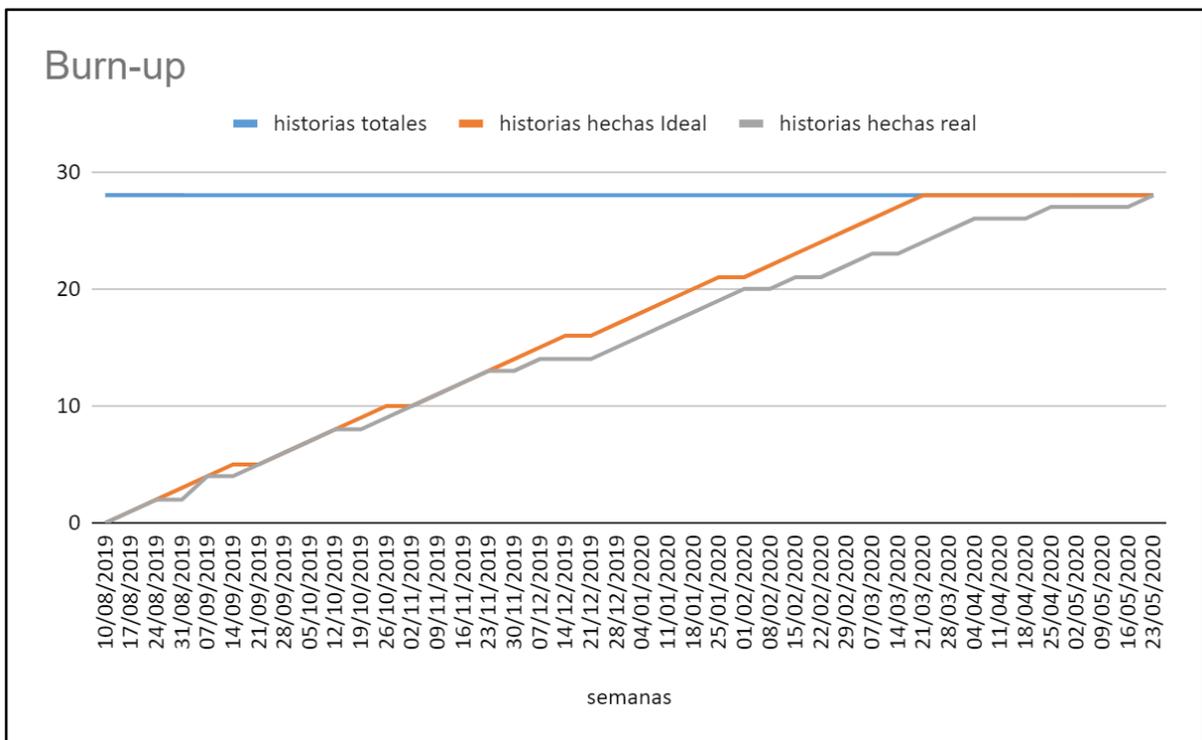
Img. 3.4.8-01 Diagrama de Entidad Final

Logramos un modelo sencillo pero altamente productivo para el cliente.

3.5 Burn-up Chart del Proyecto

El gráfico de burn-up muestra el trabajo acumulado que ya fue realizado. Esto permite tener un seguimiento del progreso del proyecto desde el inicio y ver cuántas Historias se terminaron.

Este gráfico se compone de tres líneas. La primera y horizontal indica la cantidad de historias totales que componen el proyecto. El ideal de avance es indicado por la línea color naranja y por último en color gris vemos el avance real alcanzado.



Img. 3.5-01 Burn-up del Proyecto

3.6 Testing

El equipo de trabajo consideró al Testing como una parte fundamental en todo el proceso de desarrollo, ya que ayuda a garantizar la calidad del producto final. Realizamos testing para detectar errores a tiempo y garantizar que el software cumple los requerimientos. Las pruebas se plantearon desde la primera reunión de requerimientos hasta finalizado el desarrollo.

Las pruebas que se programen servirían a lo largo de todo el proyecto y se ejecutarían luego de cada commit, verificando la calidad del código subido haciendo posible una detección temprana de fallas.

A continuación, realizaremos una breve descripción de las estrategias de Testing utilizadas y luego realizaremos una breve descripción de sus implementaciones. Las pruebas de unidad y funcionales fueron realizadas con PHPUnit, a pesar de ser poco amigable, entendimos que su potencialidad nos permitiría realizar las pruebas en todos los entornos que necesitemos. El esfuerzo se vería recompensado a medida que se avanzó con el proyecto.

Teniendo en cuenta que nos basamos en pruebas para garantizar la calidad del desarrollo, seleccionamos tres tipos de pruebas a desarrollar.

Los tipos de pruebas seleccionados fueron:

- Pruebas Unitarias.
- Pruebas Funcionales.
- Pruebas de Aceptación.

3.6.1 Pruebas de Unidad

Las pruebas unitarias aseguran que la porción de código analizada (llámese módulo o función) produce los resultados esperados al momento de ejecutarse, dado un “input” (entrada) determinado. Es decir, prueban el funcionamiento de los módulos en sus diferentes entradas. Por ejemplo, dada la función “última palabra” devuelve como resultado la última palabra de un texto. Podrían plantearse muchas entradas diferentes y los resultados esperados serían predecibles. Las pruebas unitarias funcionan con un único caso a la vez, lo que implica plantear una variedad que cubra todos los casos posibles para aumentar la calidad de la prueba.

Aquí un ejemplo de una prueba de Unidad sobre el módulo de Kinesiólogos.

```
/**
 * @dataProvider provideKinesiologos
 */
public function testKinesiologo(string $nombre, string $apellido, string $mail, string $dni,
string $nacionalidad, string $fecha, string $especialidad, string $matricula)
{
    $espe= new EspecialidadMedica();
```

```
$espe->setNombre($especialidad);
$country = new Country($nacionalidad, 'AR');
$kinesiologo= new Kinesiologo();
$nacimiento= new \DateTime($fecha);
//--VERIFICACION DE LA CLASE -
//Verifica con los Assert de cada variable de la entidad cuando se llama al constructor con parámetros
$kinesiologo->setNombre($nombre);
$kinesiologo->setApellido($apellido);
$kinesiologo->setDni($dni);
$kinesiologo->setEmail($mail);
$kinesiologo->setNacionalidad($country);
$kinesiologo->setFechaNacimiento($nacimiento);
$kinesiologo->setEspecialidadesMedicas(array($especialidad));
$kinesiologo->setMatricula($matricula);
$kine= new ArrayCollection();
$kine->add($kinesiologo);
$espe->setKinesiologos($kine);
//Verifica que el correo electrónico tenga el formato válido.
$this->assertTrue((strpos($kinesiologo->getEmail(), ".") &&strpos($kinesiologo->getEmail(), "@"),"mail
invalido");
$this->assertEquals($mail,$kinesiologo->getEmail());
$this->assertEquals($nombre, $kinesiologo->getNombre());
$this->assertEquals($apellido, $kinesiologo->getApellido());
$this->assertEquals($dni, $kinesiologo->getDni());
$aux= $kinesiologo->getEspecialidadesMedicas();
$this->assertEquals($especialidad, $aux[0]);
$this->assertEquals($matricula, $kinesiologo->getMatricula());
$this->assertEquals($country, $kinesiologo->getNacionalidad());
$this->assertEquals($nacimiento, $kinesiologo->getFechaNacimiento());
}

publicfunctionprovideKinesiologos()
{
return[
'arreglo 1' => ['juan','perez','juanperez@mail.com','12346578','Argentino','1984-12-01','Kinesiología y Fisiatría','1234'],
'arreglo 2' => ['ernesto','martinez','emartinez@mail.com','7984678','Argentino','1950-12-01','Kinesiología y Fisiatría','1235'],
'arreglo 3' => ['martin','lopez','mlopez@mail.com','15975365','Argentino','1975-12-01','Kinesiología y Fisiatría','1236'],
'arreglo 4' => ['soledad','silva','ssilva@mail.com','14578555','Argentino','1981-12-01','Kinesiología y Fisiatría','1237'],
];
}
```

Código 3.6.1-01 – ejemplo de Prueba de Unidad de Kinesiólogo

Las pruebas de unidad fueron generadas a medida que codificamos el proyecto. Las mismas son ejecutadas cada nuevo cambio o commit de código con la intención de detectar si la misma dejó de funcionar para detectar un problema.

3.6.2 Pruebas Funcionales

Las pruebas funcionales comprueban si las diferentes capas de una aplicación están integradas de forma correcta. Programarlas no es algo muy diferente a las Pruebas de Unidad pero siempre respetando el siguiente proceso:

- Realiza un request (pedido/petición)
- Comprueba la respuesta
- Hace clic en un link o rellena un formulario
- Comprueba la respuesta
- Refrescar y repetir

Aquí un ejemplo de una prueba que realiza una verificación de datos no válidos:

```

/** Solamente mostramos una prueba de la parte de los E-MAILS */
/**
 * Description of KinesiologoControllerDatosNoValidosTest
 *
 * @author matias
 */
class KinesiologoControllerDatosNoValidosTest extends WebTestCase
{
    private $client = null;

    public function setUp()
    {
        $this->client = static::createClient();
    }

    private function login()
    {
        $session = $this->client->getContainer()->get('session');
        // .. // para ver el código completo, ver anexo
        case 0:
            $this->assertContains('no puede contener números', $this->client->getResponse()->getContent());
            break;
        case 1:
            // .. // para ver el código completo, ver anexo
    }
}

/*
 * casos:
 0 - no puede contener numeros (nombre apellido)
 1 - este valor es demasiado corto (nombre apellido)
 2 - este valor no es válido (dni, matricula, mail, fecha)
 3 - este valor no debería estar vacío (nombre, apellido, dni, matricula)
 4 - NotNullConstraintViolationException
 5 - Guardado - para probar dni repetido
 6 - Excepcion dni repetido

 */
$aux = (string)rand(0, 99999999);
return [
    //no validos
    'array 1' => ['l', 'Rubio', 'lmilesi@hotmail.com', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 1],
    'array 2' => ['jose', 'm', 'lmilesi@hotmail.com', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 1],
    'array 3' => ['martin', 'tres', 'lmilesi@hotmail.com', 'tres', 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 2],
    'array 5' => ['pedro', '56cinco', 'rlmilesi@hotmail.com', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 0],
    'array 6' => ['ped66ro', 'ci66nco', 'rlmilesi@hotmail.com', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 0],
    'array 7' => ['45juan', 'cinco', 'rlmilesi@hotmail.com', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 0],
    'array 8' => ['pe44dro', 'mainardi', 'rlmilesi@hotmail.com', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 0],
    'array 9' => ['pedro', 'cincodo', 'rlmilesi@hotmail.com', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 2],
    'array 10' => ['pedro', 'cincoun', 'rlmilesi@hotmail.com', (string)rand(0, 99999999), 'Argentino', '1984-12-01', 'lala', 'Kinesiología y Fisiatría', 2],
    'array 11' => ['pedro', 'cincone', 'rlmilesi', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 2],
    'array 12' => ['pedro', 'cincoloco', '@hotmail.com', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 2],
    'array 13' => ['pedro', 'cincooro', 'rlmilesimail.com', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 2],
    'array 14' => ['pedro', 'cincotres', 'rlmilesi@hotmail', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 2],
    'array 15' => ['pedro', 'cincosiete', 'rlmilesi@hotmail.c', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 2],
    'array 16' => ['pedro', 'cincose', 'rlmilesi@hotmail.', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 2],
    //vacios
    'array 17' => ['', 'Rubio', 'lmilesi@hotmail.com', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 3],
    'array 18' => ['rafa', '', 'lmilesi@hotmail.com', (string)rand(0, 99999999), 'Argentino', '1984-12-01', (string)rand(0, 99999999), 'Kinesiología y Fisiatría', 3],
    'array 19' => ['mabel', 'Rubio', 'mabel@hotmail.com', '', 'Argentino', '1984-12-01', (string)rand(0,

```

```
999999),'Kinesiología y Fisiatría', 3],  
'array 20' => ['mabel', 'Rubio', 'rubio@hotmail.com', (string)rand (0, 99999999), 'Argentino', '1984-12-  
01', '', 'Kinesiología y Fisiatría', 3],  
];  
}  
}
```

Código 3.6.2-01 – ejemplo de validación de Datos

3.6.3 Pruebas de Aceptación

Son las pruebas de las necesidades del usuario o historias de usuario, realizadas para determinar si un sistema satisface los criterios de aceptación que permitan que el usuario pueda determinar si acepta o no el incremento presentado.

Existen dos tipos de pruebas de Aceptación:

1. Pruebas de Aceptación Formal que definen explícitamente los pasos a seguir para desarrollar la evaluación de las funcionalidades del sistema.
2. Pruebas de Aceptación Informal que no tienen la rigurosidad de las pruebas Formales. Se exploran y documentan las actividades testeadas y no tienen un plan de acción definido y estudiado al nivel de implementación que demarca el otro tipo de pruebas.

Decidimos hacer Pruebas de Aceptación Formales, las que se caracterizan por ser diseñadas y gestionadas de manera tal de comprobar fehacientemente que las funcionalidades solicitadas por el usuario fueron implementadas con la calidad que el cliente requirió.

Estas pruebas tienen un conjunto de ventajas que decidimos aprovechar:

- Conocer de antemano las funcionalidades a probar y los parámetros a tener en cuenta.
- Los criterios de aceptación se definen al diseñar la prueba.
- El progreso de las pruebas se supervisa junto con el cliente.

Las pruebas de aceptación fueron realizadas por el Cliente, asistidos por el equipo de desarrollo. Intentamos variar el Rol en el proceso con la finalidad de obtener experiencia.

3.6.4 Detalle de Pruebas

Como conclusión, vamos a mencionar que se planteó desde un comienzo realizar en detalle todas las pruebas, documentando y llevando una notación secuencial de todos los casos de pruebas unitarias, funcionales y de aceptación. El seguimiento que debía hacerse para tener esta secuencialidad y formalidad de las pruebas requería mucho tiempo limitando el progreso del desarrollo de las actividades de cada sprint. Reducimos la formalidad de las pruebas de aceptación dejando sólo el código debidamente comentado para explicar allí el objetivo del test (considerando su automatización).

Las pruebas realizadas fueron las siguientes. Las pruebas CPA son pruebas de aceptación y para su realización siempre contábamos con el complemento del Manual de Usuario. Las Pruebas CP son Funcionales o de Unidad. El detalle de todas las pruebas está en el Anexo de Pruebas.

Nombre	Título de la prueba	Descripción	Sprint	Fecha de diseño	Fecha de ejecución de la prueba
CPA2-01	Prueba de inicio de sesión	Verificar el correcto inicio de sesión con un usuario y contraseña válidos	2	28-sep-19	05-oct-19
CPA2-02	Prueba de inicio de sesión con un usuario y/o contraseña no válidos	Verificar la negativa de inicio de sesión con un usuario y/o contraseña no válidos.	2	28-sep-19	05-oct-19
CP2-03	Prueba de Creación de nuevo usuario	Verifique que puede dar de alta un nuevo usuario en el sistema	2	28-sep-19	05-oct-19
CP2-04	Prueba de edición de un usuario	Verifique que puede editar un usuario en el sistema	2	28-sep-19	05-oct-19
CP2-05	Prueba de asignación de un Rol a un usuario	Probar que se pueden asignar Roles a un Usuario	2	28-sep-19	05-oct-19
CP2-06	Prueba de edición de Roles de usuarios	Probar que se pueden editar los Roles a un Usuario	2	01-oct-19	05-oct-19
CP2-07	Prueba de Creación de Consultorio	Verificar la carga correcta de un Consultorio.	2	04-oct-19	05-oct-19
CP2-08	Prueba de Edición de un Consultorio	Verificar la correcta edición de los datos de un Consultorio.	2	04-oct-19	05-oct-19
CP2-09	Prueba de Eliminación de un Consultorio	Verifica la eliminación de un Consultorio existente en base de datos	2	04-oct-19	05-oct-19
CPA2-10	Prueba de Creación de nuevo usuario	Verifique que puede dar de alta un nuevo usuario en el sistema	2	04-oct-19	05-oct-19
CPA2-11	Prueba de edición de un	Verifique que puede editar un usuario	2	04-oct-19	05-oct-19

	usuario	en el sistema			
CP2-12	Prueba de las rutas iniciales del sistema	Probar si la web responde correctamente al requerimiento de las URL implementadas en la primera etapa. (con código nulo)	2	04-oct-19	05-oct-19
CP3-01	Prueba de creación de un Paciente	Verificación del alta de un paciente en el sistema	3	15-oct-19	02-nov-19
CP3-02	Prueba de edición de un Paciente	Verificar la correcta edición de los datos de un paciente.	3	15-oct-19	02-nov-19
CP3-03	Prueba de eliminación de un Paciente	Verifica la eliminación de un Consultorio existente en base de datos	3	15-oct-19	02-nov-19
CP3-04	Prueba de creación de Kinesiólogo	Prueba automatizada de creación de Kinesiólogos	3	26-oct-19	02-nov-19
CP3-05	Prueba de edición de un Kinesiólogo	Prueba automatizada de edición de un Kinesiólogos	3	26-oct-19	02-nov-19
CP3-06	Prueba de Eliminación de un Kinesiólogo	Probar si se puede eliminar un Kinesiólogo previamente creado	3	26-oct-19	02-nov-19
CP3-07	Prueba de edición de una patología	Probar que se pueden editar las Patologías.	3	26-oct-19	02-nov-19
CP3-14	Prueba de Creación de una Patología	Probar que se puede crear una Patología	3	26-oct-19	02-nov-19
CPA3-08	Prueba de creación de un Kinesiólogo	Verifique que puede dar de alta un Kinesiólogo.	3	25-oct-19	02-nov-19
CPA3-09	Prueba de edición de un Kinesiólogo	Verifique que puede editar un Kinesiólogo.	3	25-oct-19	02-nov-19
CPA3-10	Prueba de creación de una Patología	Verifique que puede dar de alta una Patología	3	25-oct-19	02-nov-19
CPA3-11	Probar asociar un Paciente a una Obra Social	Probar asociar un paciente a una Obra social estableciendo un Plan en esa asociación.	3	25-oct-19	02-nov-19
CPA3-12	Prueba de creación de un Turno Rápido	Verifique que puede dar de alta un Turno Rápido.	3	25-oct-19	02-nov-19
CPA3-13	Prueba de edición de una Patología	Verifique que puede editar una Patología	3	25-oct-19	02-nov-19
CP4-01	Prueba de creación de un Turno	Verifique que puede dar de alta un Turno.	4	01-ene-20	07-ene-20
CP4-02	Prueba de edición de un Turno	Verifique que puede editar un Turno.	4	01-ene-20	07-ene-20
CPA4-03	Prueba de eliminación de un Turno	Verificar que el sistema permite eliminar un turno determinado	4	01-ene-20	07-ene-20

CPA4-04	Probar el análisis de disponibilidad de Turno según la disponibilidad	Se deberá probar la disponibilidad de los recursos para agregar un turno, según la disponibilidad de recursos en ese horario.	4	01-ene-20	07-ene-20
CPA4-05	Prueba de creación de un Turno	Verifique que puede dar de alta un Turno.	4	01-ene-20	07-ene-20
CPA4-06	Prueba de edición de un Turno	Verifique que puede editar un Turno.	4	01-ene-20	07-ene-20
CPA4-07	Probar asignar un turno a pesar de no tener recursos disponibles	Probar asignar un turno a pesar de no tener recursos disponibles, el resultado debería ser exitoso	4	01-ene-20	07-ene-20
CPA4-08	Prueba de creación de un Recurso Modelo	Probar que se puede crear un Recurso Modelo	4	02-ene-20	07-ene-20
CPA4-09	Prueba de creación de un Recurso	Probar que se puede crear un Recurso y Asignarlo a un Recurso Modelo	4	02-ene-20	07-ene-20
CPA4-10	Prueba de Edición de un Recurso Modelo	Probar que se puede Editar un Recurso Modelo	4	02-ene-20	07-ene-20
CPA4-11	Prueba edición de un Recurso	Probar que se puede editar un Recurso	4	02-ene-20	07-ene-20
CPA4-12	Prueba de Eliminación de un Recurso Modelo	Verificar que se puede eliminar un Recurso Modelo	4	02-ene-20	07-ene-20
CPA4-13	Prueba de Eliminación de un Recurso	Verificar que se puede Eliminar un Recurso	4	02-ene-20	07-ene-20
CPA4-14	Prueba de creación de un Arancel de una Obra Social	Probar que se pueden crear un Arancel de un Tratamiento en una Obra Social	4	05-ene-20	07-ene-20
CPA4-15	Prueba de Edición de un Arancel de Obra Social	Probar que se puede editar un Arancel de la Obra Social.	4	05-ene-20	07-ene-20
CPA5-01	Prueba de Creación de una Orden Médica	Verificar el correcto funcionamiento de la Creación de una Orden Médica	5	06-feb-20	08-feb-20
CPA5-02	Prueba de edición de Orden Médica	Probar que se puede editar una Orden Médica	5	06-feb-20	08-feb-20
CPA5-03	Prueba de Eliminación de Orden Médica	Probar que se puede eliminar una Orden Médica	5	06-feb-20	08-feb-20
CPA5-04	Prueba de mostrado de turnos en calendario	Probar que los datos del Turno se muestran correctamente en el Calendario.	5	01-feb-20	08-feb-20
CPA5-05	Probar acceder a una Turno mediante el calendario	Probar la funcionalidad de acceder a un Turno mediante un clic en el calendario	5	01-feb-20	08-feb-20
CPA5-06	Prueba de Creación de un Tratamiento	Probar que se puede Crear un Tratamiento	5	06-feb-20	08-feb-20

CPA5-07	Prueba de Edición de un Tratamiento	Probar que se puede editar un Tratamiento	5	06-feb-20	08-feb-20
CPA5-08	Probar el análisis de disponibilidad de Turno según la disponibilidad	Se deberá probar la disponibilidad de los recursos para agregar un turno, según la disponibilidad de recursos en ese horario.	5	06-feb-20	08-feb-20
CPA5-09	Probar asignar un turno a pesar de no tener recursos disponibles	Probar asignar un turno a pesar de no tener recursos disponibles, el resultado debería ser exitoso	5	06-feb-20	08-feb-20
CPA6-01	Prueba de marcado como cobrado una Orden Médica a cobrar	Probar que se puede marcar como cobrada una Orden Médica	6	05-mar-20	07-mar-20
CPA6-02	Prueba de envío de Orden Médica a cobrar	Probar que se puede enviar a cobrar una Orden Médica	6	05-mar-20	07-mar-20
CPA6-03	Prueba de Creación de Sesiones de Tratamientos	Probar que se pueden crear nuevas Sesiones en un Tratamiento asignando Recursos	6	05-mar-20	07-mar-20
CPA6-04	Prueba de Edición de Sesiones de Tratamientos	Probar que se pueden modificar las Sesiones de un Tratamiento y sus Recursos necesarios.	6	05-mar-20	07-mar-20
CPA6-05	Prueba de asignación de turnos en calendario mediante nueva Rehabilitación	Probar que se pueden asignar turnos mediante la opción del calendario en la generación de una nueva Rehabilitación.	5	06-mar-20	07-mar-20
CPA6-06	Prueba de asignación de turnos mediante horarios cuando generamos una nueva Rehabilitación	Probar que se pueden asignar turnos mediante la opción de Horarios en la generación de una nueva Rehabilitación.	5	06-mar-20	07-mar-20
CPA7-01	Prueba de Creación de Variables de Seguimiento	Probar que se puede generar una nueva Variable de Seguimiento.	7	01-abr-20	04-abr-20
CPA7-02	Prueba de Edición de Variables de Seguimiento	Probar que se pueden Editar Variables de Seguimiento	7	01-abr-20	04-abr-20
CPA7-03	Prueba de Eliminación de Variables de Seguimiento	Probar que se puede eliminar una Variable de Seguimiento	7	01-abr-20	04-abr-20
CPA7-04	Prueba de Gráficos de Variables de Tratamientos	Probar que el sistema Grafica el resultado de las Rehabilitaciones según las mediciones ingresadas.	7	01-abr-20	04-abr-20
CPA8-01	Pruebas de generación de Informe Diario de Turnos	Probar que se puede generar un PDF con todos los turnos del día en Curso.	8	17-may-20	22-may-20
CPA8-02	Pruebas de generación de Informe Diario de Turnos de un día específico.	Probar que se puede generar un PDF con todos los turnos de un día en específico.	8	17-may-20	22-may-20
CPA8-03	Pruebas de generación de Informes de Cobros	Probar la generación de los Cobros	8	17-may-20	22-may-20

	Pendientes	Pendientes			
CPA8-04	Pruebas de generación de Informes de Envíos Pendientes	Probar la generación de los Envíos Pendientes	8	17-may-20	22-may-20
CPA8-05	Pruebas de generación de Informes de Cobros del Último Mes	Probar la generación de los Cobros del Último mes	8	17-may-20	22-may-20

Tabla 3.6.4 -01 – Detalle de las Pruebas

Aquí mostramos el código de una de las pruebas CP. Para ver el documento que permitió codificarla, diríjase al Anexo de Pruebas.

CP2-12	Prueba de las rutas iniciales del sistema	Probar si la web responde correctamente al requerimiento de las URL implementadas en la primera etapa. (con código nulo)	2	04-oct-19	05-oct-19
--------	---	--	---	-----------	-----------

Esta prueba nos permitió detectar rápidamente los errores generados al momento de actualizar los nombres de las Rutas por otras más amigables. Luego de una serie de repeticiones y actualización de la Prueba logramos solucionar todos los inconvenientes en menos de una hora de trabajo.

```

<?php
namespace AppBundle\Tests\Route;

use Symfony\Bundle\FrameworkBundle\Test\WebTestCase;
use Symfony\Component\BrowserKit\Cookie;
use Symfony\Component\HttpFoundation\Response;
use Symfony\Component\Security\Core\Authentication\Token\UsernamePasswordToken;

/*
 * CP02-12:
 * Prueba de las rutas iniciales del sistema
 * Probar si la web responde correctamente al requerimiento de las URL implementadas en la primer etapa. (con código nulo)
 * Condiciones:
 * El Sistema debe tener cargado Turnos, Órdenes Médicas y Rehabilitaciones.
 */
class CP02_12Test extends WebTestCase
{
    private $client = null;

    public function setUp()
    {
        $this->client = static::createClient();
    }

    private function login()
    {
        $session = $this->client->getContainer()->get('session');
        $firewallName = 'main';
        $firewallContext = 'main';
        $person = self::$kernel->getContainer()->get('doctrine')->getRepository('AppBundle:User\User')->findOneByUsername('admin');
    }
}

```

```
$token = new UsernamePasswordToken($person, null, $firewallName, $person->getRoles());
$session->set('_security_'.$firewallContext, serialize($token));
$session->save();

$cookie = new Cookie($session->getName(), $session->getId());
$this->client->getCookieJar()->set($cookie);
$this->client->catchExceptions(false);
}

/**
 * @dataProvider urlProvider
 */
publicfunctiontestPagelsSuccessful($url)
{
    $this->login();
    //$client = self::createClient();
    $this->client->request('GET', $url);
    $this->assertSame(Response::HTTP_OK, $this->client->getResponse()->getStatusCode());
}

publicfunctionurlProvider()
{
    returnarray(
        //rutas de ficha de las ordenes médicas: nueva, edición, show y general
        array('/ordenes-medicas'),
        array('/ordenes-medicas/1'),
        array('/ordenes-medicas/1/edit'),

        //rutas de ficha del dashboard
        array('/dashboard'),

        //rutas de ficha de turnos: nueva, edición, show y general
        array('/turnos'),
        array('/turnos/1'),
        array('/turnos/1/edit'),
        array('/turnos/new'),
        array('/turnos/new_rapido'),

        //rutas de ficha de rehabilitación: nueva, edición, show y general
        array('/fichas-rehabilitaciones'),
        array('/fichas-rehabilitaciones/new'),
        array('/fichas-rehabilitaciones/40'),
        array('/fichas-rehabilitaciones/40/edit'),

        //ruta de pacientes
        array('/pacientes'),
        array('/pacientes/30'),
        array('/pacientes/edit/30'),
        array('/pacientes/new'),

        //ruta de kinesiologos
        array('/kinesiologos/view/15'),
        array('/kinesiologos/edit/15'),

        //ruta de patologías
        array('/patologias'),
        array('/patologias/new'),
        array('/patologias/28'),
        array('/patologias/28/edit'),

        //ruta de tratamientos
        array('/tratamientos'),
        array('/tratamientos/5'),
```

```
array('/tratamientos/5/edit'),

//rutas de variables y recursos
array('/variables/new'),
array('/recursos'),
array('/recursos/1'),
array('/recursos/1/edit'),
array('/recursomodelo'),
array('/recursomodelo/32'),
array('/recursomodelo/32/edit'),

//ruta de especialidades medicas y obras sociales
array('/especialidades-medicas'),
array('/obras-sociales'),

//rutas de consultorios y notificaciones
array('/consultorios'),
array('/notifications'),

);
}
}
```

Código 3.6.4 -01 – CP02-12 – Prueba de Rutas

3.7 Integración Continua

Bitbucket ofrece una herramienta de Testeo llamado Pipelines, el cual evalúa el código al momento de realizar Commit. Para hacer esto toma el archivo de configuración de Docker y genera un espacio virtual donde levanta todos los contenedores especificados, iguales que en el ambiente de Desarrollo.

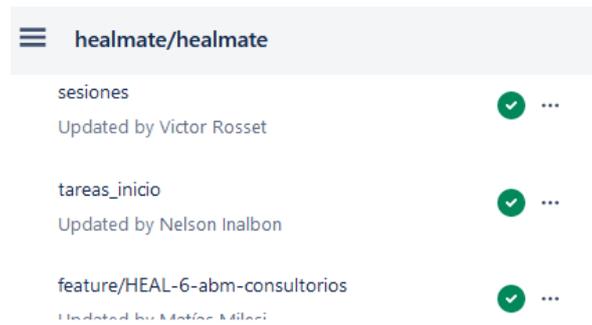
Aquí fue clave la correcta generación de Fixtures (serie de definiciones iniciales de valores para que el sistema funciones, ejemplo: usuarios estándares, patologías estándares, etc.) para tener los modelos iniciales de datos al momento de simular un nuevo entorno de pruebas.

La ventaja de la herramienta de Pipelines es su herramienta de Integración Continua, que hace un Deployment⁸ sobre el Server si todas las pruebas fueron correctas.

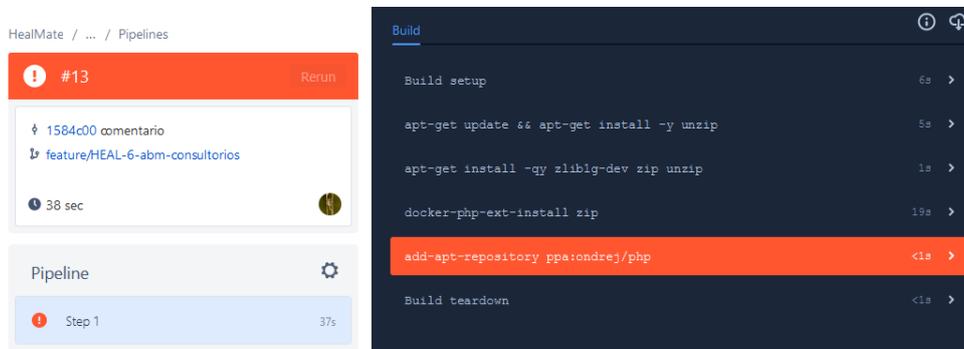
Aquí algunos ejemplos de la evaluación y sus resultados:

En caso de que todas las pruebas hayan tenido éxito mostrará una tilde verde

⁸Deployment: son todas las actividades que hacen que un sistema de software esté disponible para su uso



En caso de que no hayan tenido éxito y las pruebas se cancelaron en el proceso, mostrará un signo de exclamación.



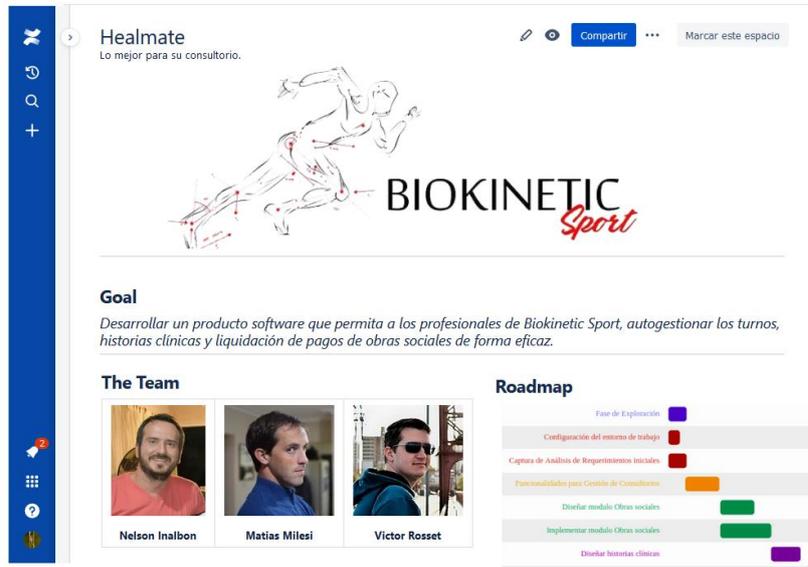
Img. 3.7-01: Prueba cancelada por error en test

3.8 Gestión del Código

Para mantener el control de las versiones de nuestro código de forma eficiente y confiable utilizamos Git, el cual lleva control y registro sobre los cambios en los archivos y permite coordinar el trabajo sobre los archivos compartidos. Utilizamos un servidor de Git de Bitbucket que está asociada a herramientas como Confluence y Jira.

Jira permite crear una hoja de Ruta del proyecto y hacer un seguimiento sobre el mismo, cuenta también con tablero de Kanban para gestionar el trabajo cotidiano.

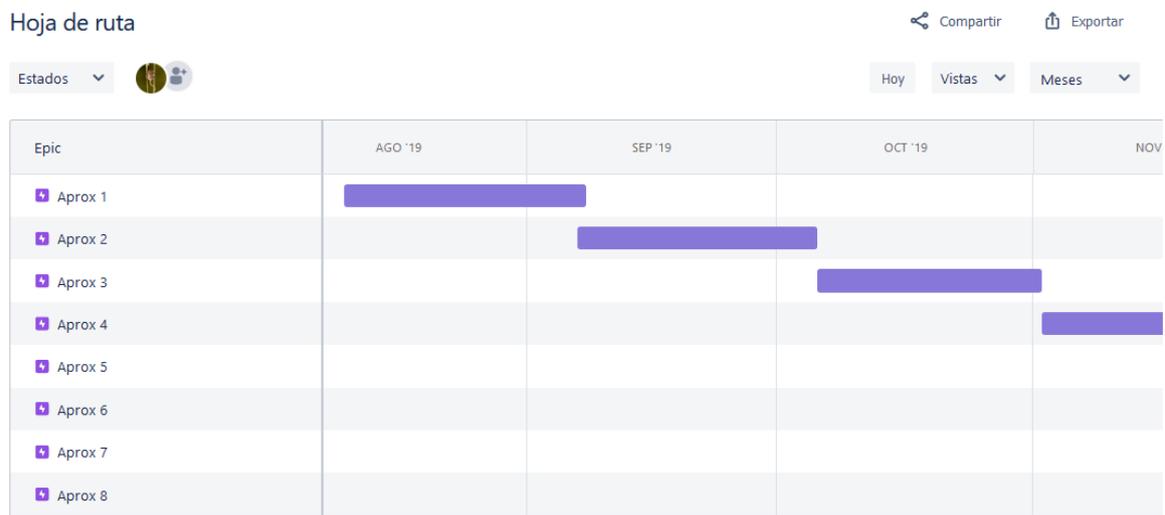
Confluence es un espacio de trabajo con una completa variedad de plantilla de todo tipo de documentos, lo cual nos permitió mantener toda la documentación de Requerimientos, Avances, Reuniones, Análisis de casos particulares, Análisis de riesgos, etc. Aquí la página principal de Confluence.



Img. 3.8-01: Página principal de Confluence

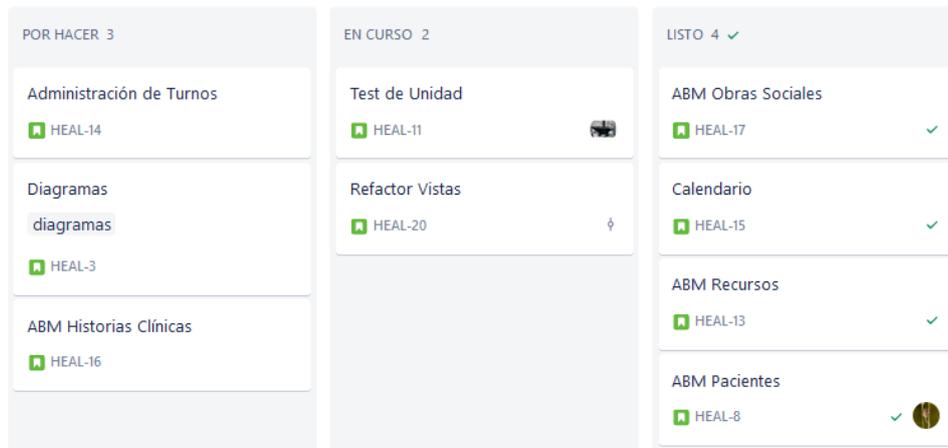
Al comenzar definimos la hoja de Ruta inicial, en la cual sabíamos habría muchos cambios, dejando todo previsto para permitir los cambios a medida que avance el proyecto.

Aquí mostramos la generación inicial del proyecto:



Img. 3.8-02: Ejemplo Diagrama de Gantt

Y aquí el tablero Kanban en un momento determinado (a modo de ejemplo) ya que la herramienta no es amigable para ver fotografías de estados en determinada fecha.

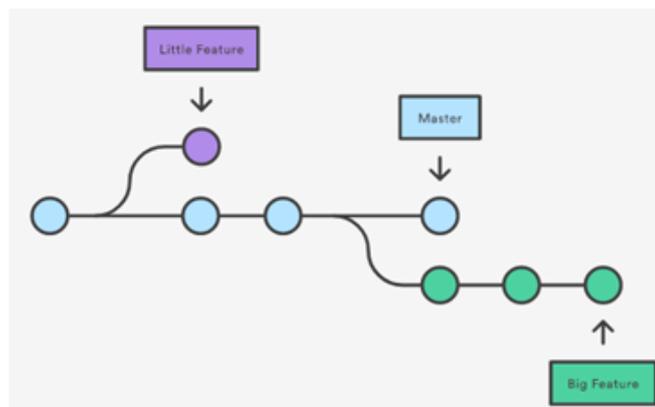


Img. 3.8-03: Ejemplo de Tablero Kanban

3.8.1 Branching o manejo de ramas

A continuación, realizaremos una breve explicación de la utilidad del manejo de Ramas y luego explicaremos nuestra implementación.

“Las ramas son parte del proceso de desarrollo diario. Las ramas de Git son un puntero eficaz a las instantáneas de tus cambios. Cuando quieres añadir una nueva función o solucionar un error (sin importar su tamaño) generas una nueva rama para alojar estos cambios. Esto hace que resulte más complicado que el código se fusione con el código base principal y te da la oportunidad de limpiar tu historial futuro antes de fusionarlo con la rama principal.”



Img. 3.8.1-01 : Ejemplo de trabajo en Ramas

El diagrama anterior representa un repositorio con dos líneas de desarrollo aisladas, una para una función pequeña y otra para una función más extensa. Al desarrollarlas en ramas, no solo es posible trabajar con las dos de forma paralela, sino que también se evita que el código dudoso se fusione con la rama maestra.

La implementación que subyace a las ramas de Git es mucho más sencilla que la de otros modelos de sistemas de control de versiones. En lugar de copiar archivos entre directorios, Git almacena una rama como referencia a una confirmación. En este sentido, una rama representa el extremo de una serie de confirmaciones, es decir, no es un contenedor de confirmaciones. El historial de una rama se extrapola de las relaciones de confirmación. (Learn Branching with Bitbucket, 2020, enlace: <https://www.atlassian.com/es/git/tutorials/using-branches>).

3.8.2 Implementación de Branching

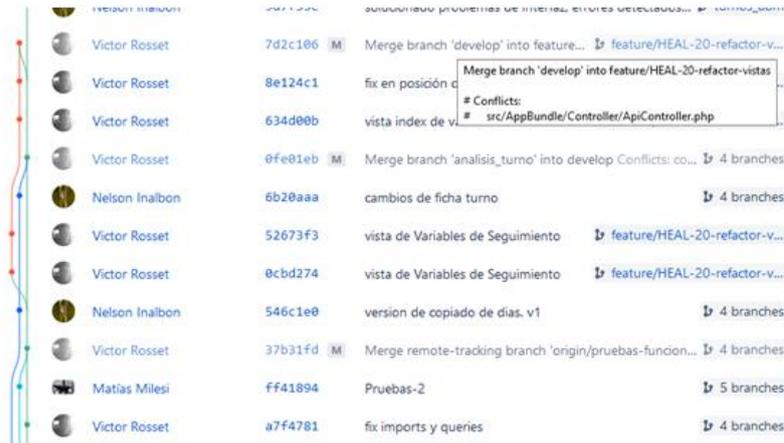
Es una forma de trabajo para organizar el desarrollo de software cuando se quiere agregar alguna funcionalidad de forma organizada y controlada sin cambiar el desarrollo en producción hasta verificar todo lo necesario antes de implementarlo. Sirve para gestionar y ordenar el código así como sus versiones en el desarrollo.

Nosotros elegimos trabajar con una rama principal llamada Master y otra Develop como guías principales (Baseline) sobre las que se sentarán los pilares de nuestro de desarrollo. Al generar una nueva rama y trabajar en ella significa que, si modificamos, borramos o añadimos archivos, esta incluirá esos cambios pero no así la rama Master o Develop.

La Rama Master (Principal) es la rama que tendrá el código que está en producción, el cual será subido al servidor una vez verificado todo lo necesario y realizadas las pruebas de Integración.

La Rama Develop (Desarrollo) es la rama que tendrá el código en Desarrollo. Siempre se verá en el árbol que las ramas vuelven a develop y salen de develop, ya que es la rama principal de control.

El merge o la fusión sobre las dos Ramas, tanto en Develop como en Master es un poco tedioso pero tiene muchos más beneficios que costos. Al realizar una doble verificación en el Merge de las Ramas, se obtiene un código muy estable y testeado.

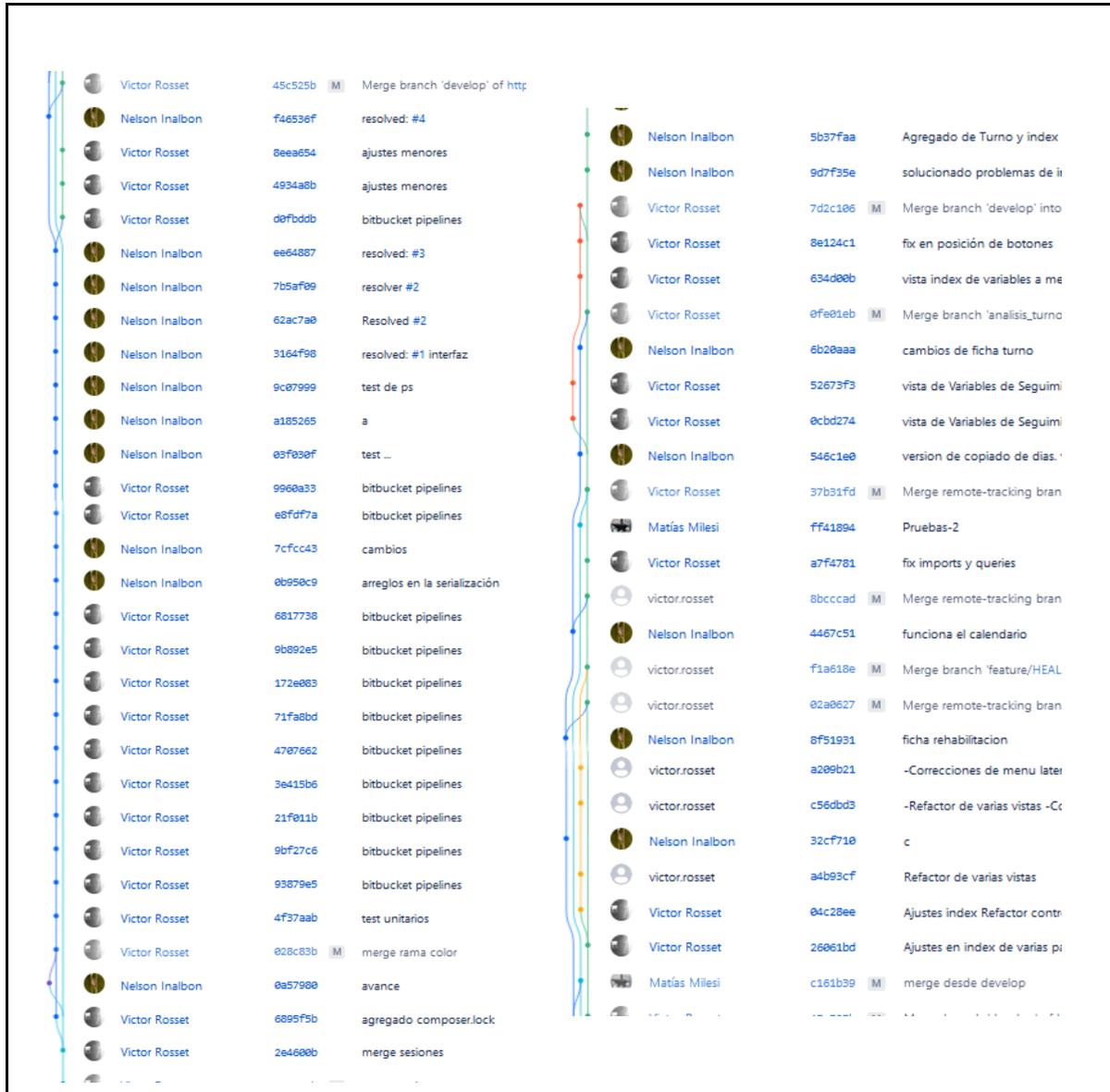


Img. 3.8.2-01 : Ejemplo de vista de las Ramas

Según el gráfico, la rama Roja es la Master, la Azul una rama de Gestión de Turnos especial y la Verde la rama Develop. Se puede observar cómo en un punto determinado del desarrollo se optó por volcar el contenido Develop a Master para realizar una actualización en Producción.

La potencialidad de esta forma de trabajo es la practicidad de acoplarse al trabajo y continuar desarrollando sin desperdiciar tiempo valioso. Afortunadamente para el grupo, adoptamos esta forma de trabajo, ya que en el proceso de desarrollo migramos varias veces de Terminales de Desarrollo, al menos tres veces por Integrante. Para facilitar estas migraciones generamos un documento inicial README, el cual tiene toda la información de cómo acoplarse al trabajo y los comandos para actualizar el entorno de trabajo. Éste debió ser actualizado también cuando se avanzó en el desarrollo.

Aquí mostramos una breve parte del árbol generado:



Recuadro 3.8.2-02: Muestra de Ramas en el desarrollo

4 Marco de Trabajo

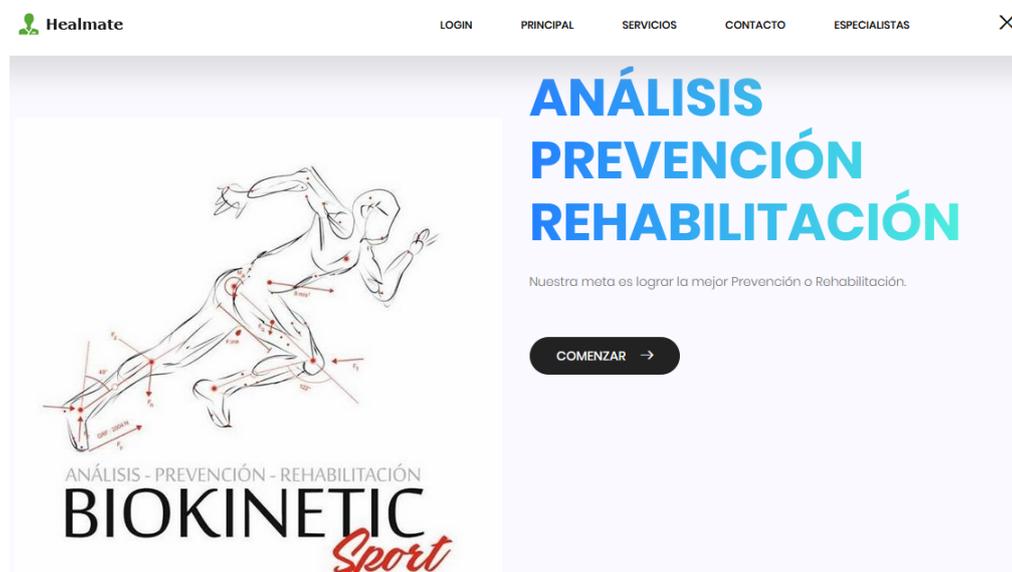
Un marco de trabajo define en términos generales un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia para enfrentar y resolver nuevos problemas de índole similar.

4.1 Implementación de la interfaz de usuario

Las premisas planteadas en los requerimientos del proyecto obligaron a hacer un párrafo aparte para la interfaz de usuario y la capacidad de accesibilidad ofrecida, tomando como obligatorio la capacidad del usuario de acceder a cualquier parte del sistema con no más tres clics y una integridad en el diseño que permita diferenciar rápidamente en el sector del software en el cual estamos y detectar también situaciones anormales.

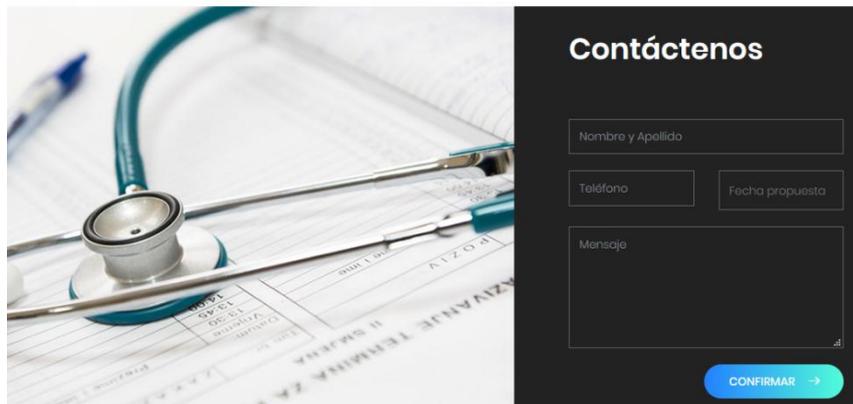
4.1.1 Página de Bienvenida

En todo sitio Web es importante la página de bienvenida, en la cual se intenta mostrar información de contacto y una breve reseña de lo que hace el consultorio y sus especialistas. Esto se logró con un diseño simple y funcional.



Como se observa en el gráfico, en la barra superior están disponibles las Opciones de Logín, Principal, Servicios, Contacto y Especialistas.

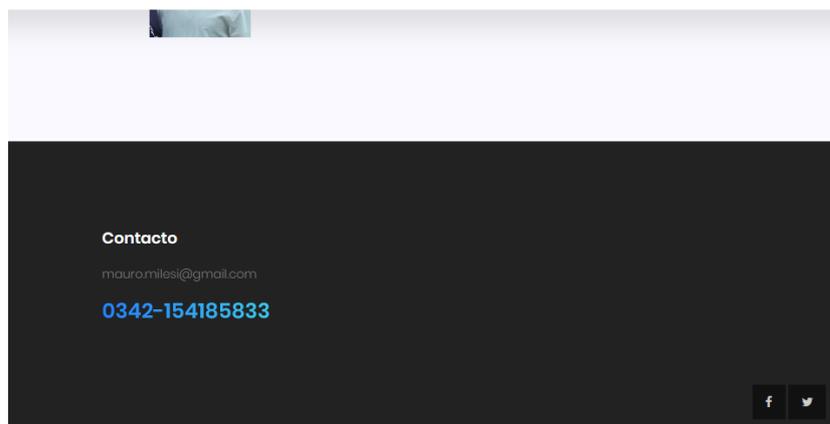
Contactos:



Especialistas:



Datos de Pie de página:



4.1.2 Inicio de Sesión

Se utilizó el diseño clásico para el inicio de sesión, con diseño acorde al sistema.

Healmate

Por favor, inicie sesión

nombre de usuario

Contraseña

Recordarme?

INICIAR SESIÓN

[Olvidé mi contraseña](#)
[Registrar una cuenta nueva](#)

4.1.3 Barras de MENU

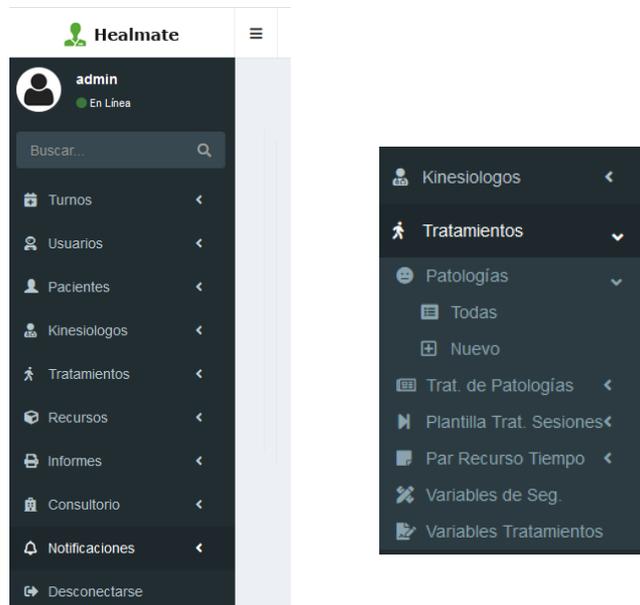
4.1.3.a Barra Lateral

La barra lateral es accesible y ocultable por el botón situado a su derecha.

Según el Rol de Usuario, las opciones serán variadas. Aquí vemos el ROL de Administrador, el cual tiene todas las opciones habilitadas.

Cada Opción tiene un árbol de opciones para acceder a las diferentes funcionalidades del sistema.

Aquí un ejemplo de las opciones internas



4.1.3.b Barra Superior

La barra Superior contiene las opciones de la Sesión, en ellos se ven las notificaciones y el perfil del usuario.



-Notificaciones y Mensajes:

Las Notificaciones se muestran en el lugar tradicional de las aplicaciones Web.



Informa la cantidad de notificaciones sin leer con un número en amarillo.

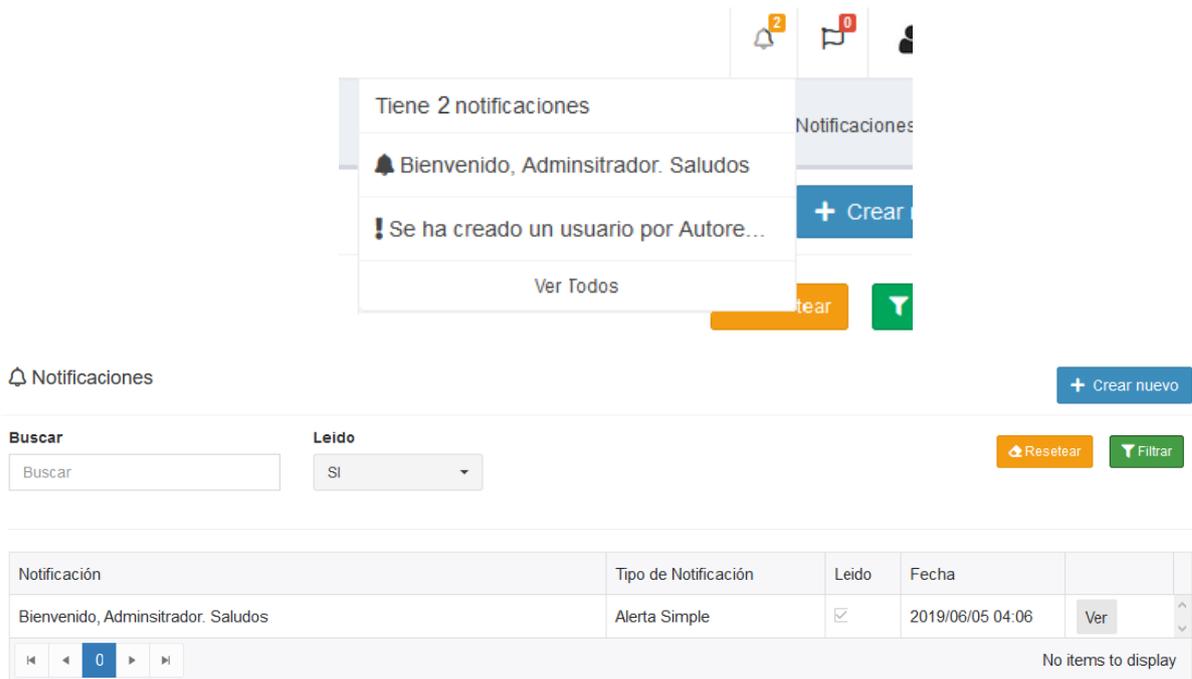
Presionando sobre el ícono se despliegan las notificaciones.

Podrá acceder a cualquiera de las notificaciones presionando sobre ellas y así ver el detalle.

El tipo de notificación (Mensaje) es personalizable, el usuario podrá cargar el tipo de mensaje y detalle el ícono que desea utilizar. Para comenzar con el proyecto sólo elegimos

dos tipos de notificaciones: alertas e información. El usuario estuvo de acuerdo y consideró que un futuro generaría nuevas, a medida que utilice más el sistema.

También incorporamos búsquedas por Texto de notificación y por Estado (leído o no leído)



Notificaciones

Buscar

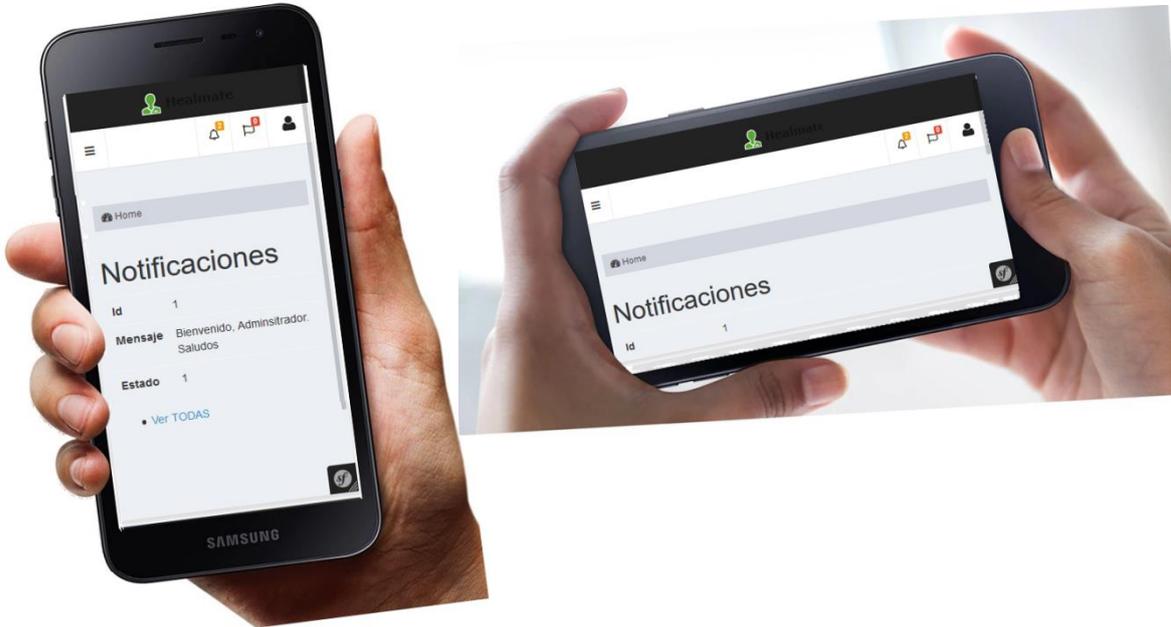
Leido

Notificación	Tipo de Notificación	Leido	Fecha
Bienvenido, Adminsitrador. Saludos	Alerta Simple	<input checked="" type="checkbox"/>	2019/06/05 04:06

No items to display

4.1.4 Funcionamiento en Dispositivos Móviles

Para el funcionamiento en dispositivos móviles se optó también por un diseño compacto. Sólo se muestra información importante para la página que se esté mostrando, es por eso que la barra lateral aparece oculta (para facilitar la navegabilidad) y la barra superior se mantiene con la opción de ampliar el menú lateral y mostrar las notificaciones.



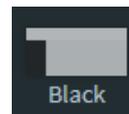
4.1.5 Paletas de Colores

Se escogieron colores de contraste medio.

Menú lateral: Color gris oscuro

Menú superior: Color gris claro.

Ventanas internas: Color gris.



Botones: Se trabajó con Bootstrap, en sus versiones Primary, Success, Danger, Info y Warning.

Switches: También basado en Bootstrap, con sus equivalentes colores para lograr un gran contraste.

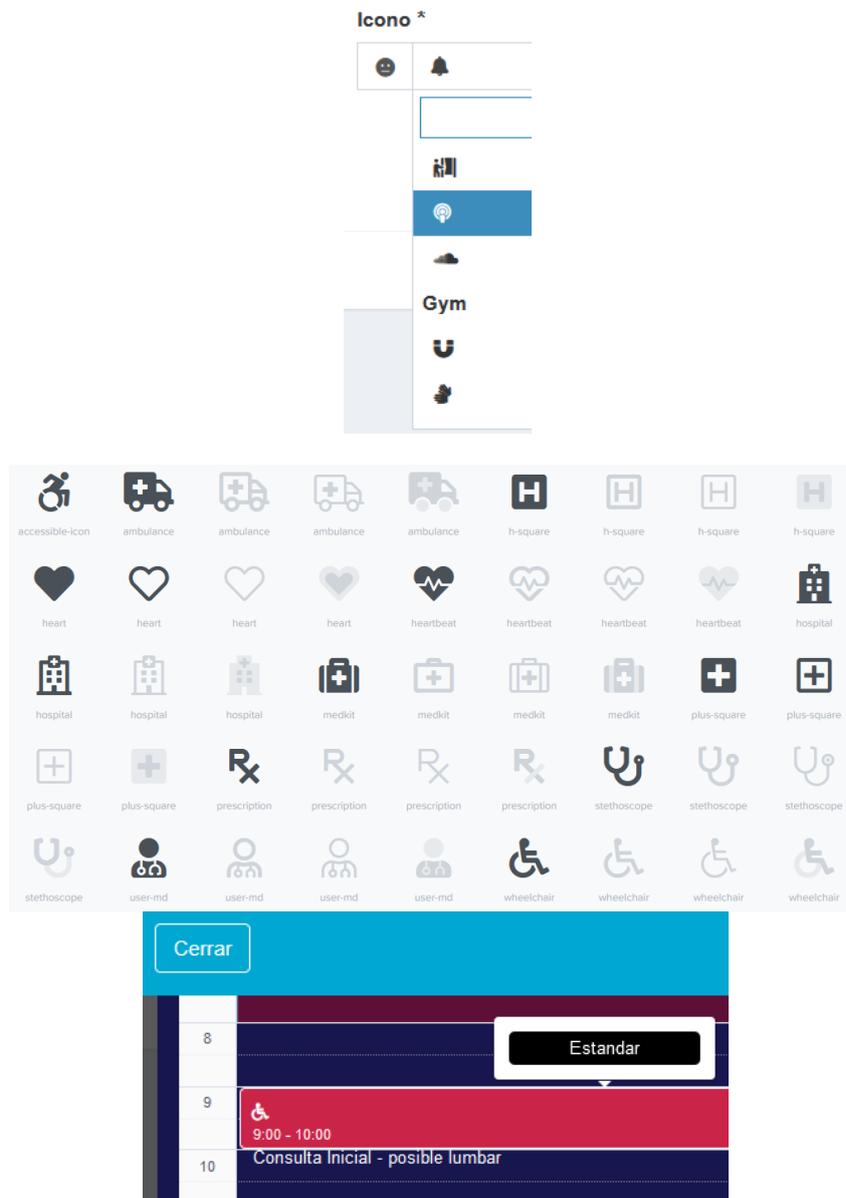


4.1.6 Iconografía

Consideramos primordial una buena iconografía para poder asociar los recursos del Consultorio a ellos, simplificando la presentación de datos en el Calendario y los formularios de consulta.

Se integró el paquete de íconos de Font Awesome de © Fonticons, Inc que tiene integrado variados íconos relacionados a salud, advertencias e íconos estándares, el cliente consideró suficiente.

Aquí una muestra de algunos de los íconos para elegir.



4.1.7 Tipografía

Por motivos estéticos la tipografía cumple un rol importante en mejorar el diseño y profesionalidad de nuestro sitio web. No sólo con la intención de hacerla más atractiva visualmente sino por cuestiones que afectan directamente al usuario:

Legibilidad y profesionalidad como premisa: Elegir una buena tipografía supondrá que el usuario leerá más cómodamente el contenido de nuestro software. La tipografía es acorde a la profesionalidad del Consultorio, es un aspecto de coherencia empresarial exigida por el cliente.

Utilizamos una fuente estándar para los sistemas web, de corte simple, clara y funcional para los calendarios, informes y formularios. Perteneciente a Google Fonts: **Source Sans Pro.**

Extra-Light
Extra-Light Italic
Light
Light Italic
Regular
Regular Italic
Semi-Bold
Semi-Bold Italic
Bold
Bold Italic
Black
Black Italic

4.1.8 Mensajes y Alertas

Para los elementos emergentes y mensajería utilizamos Toastr.

La herramienta integrada en el sistema cuenta con los 4 tipos de mensajes más utilizados que son suficientes para las necesidades de comunicación de mensajes al usuario. Ellos son Éxito, Información, Peligro y Error.

(Copyright © 2012-2014 [Hans Fjällemark](#), [JohnPapa](#) & [Tim Ferrell](#).)



4.1.9 Calendario

Al ser un sistema que estaría integrado fuertemente con los turnos de los pacientes fue crucial buscar un calendario (para no desarrollar uno nosotros) que sea funcional en mostrar información de turnos, íconos, colores y personalizaciones de horarios laborales y eventos. Analizamos varias alternativas y optamos por la opción de Full-Calendar que presentaba un Bundle integrable con Symfony en la versión que fue elegida para el desarrollo del sistema.

Consideramos los siguientes aspectos para elegirlos:

Responsive

El Bundle elegido está adaptado para trabajar en dispositivos móviles y browsers de PC. Presenta muy buenas interfaces para dispositivos móviles conservando todos los aspectos configurados de diseño, permitiendo una navegabilidad concordante en todos los dispositivos.



Personalización

Los elementos que se permiten personalizar son muy variados y totalmente funcionales para las necesidades del cliente: Se pueden mencionar: las vistas por Mes, Semana y Día; Resizing dinámico, botones de navegación de días, semana y meses, opción de ir a fecha actual; eventos de clic en Browsers y dispositivos móviles (pantallas touch); husos horarios y zonas de trabajo.

Popups

Con la premisa de disminuir la cantidad de clics necesarios en la utilización del sistema, utilizamos los Popups para mostrar más información del turno al hacer clic sobre él.

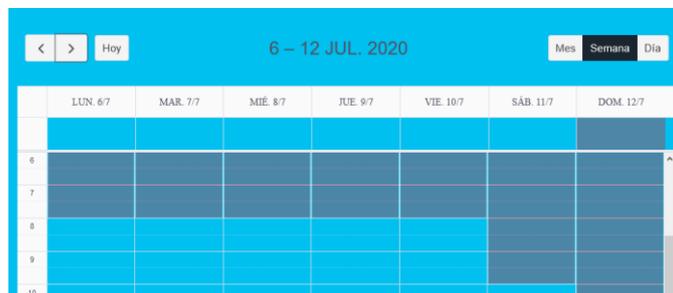


Personalización de los Turnos

Los tipos de turnos pueden personalizarse al punto de elegir colores de fondo, tipografías, iconografía, textos y programación de comportamientos dependiendo del mismo de evento capturado.

Horarios Laborales

El Bundle permite configurar los horarios laborales para así lograr una mejor legibilidad en el manejo y asignación de turnos lo que el cliente considera de suma importancia.



Captura de Acciones dentro del calendario

Fue de suma importancia para lograr navegabilidad y accesibilidad en el sistema la posibilidad de ir a un turno con tal solo un clic en calendario o poder asignar un turno viendo el calendario y su información de carga. Se permite así una muy buena funcionalidad visual y potenciando el sistema para su practicidad.

4.2 Presentación del Sistema

En esta sección presentamos los principales módulos y sus funcionalidades, enumerándolas y describiendo brevemente las capacidades de cada módulo.

4.2.1 Usuarios

La solución elegida es una plataforma web. Se definen usuarios y contraseñas para cada uno para dar acceso seguro y permitir la definición de distintos roles que limitarán las funcionalidades de estos en el sistema.

Los perfiles apuntan a dar o no acceso a los módulos de gestión, permitiendo proteger la información o configuración de un Consultorio, ayudando a tener un sistema consistente. Este es un punto fundamental ya que de la correcta carga de datos del negocio es necesaria

para dar una óptima solución a uno de los principales requerimientos del presente desarrollo.

Se cuenta con cuatro perfiles:

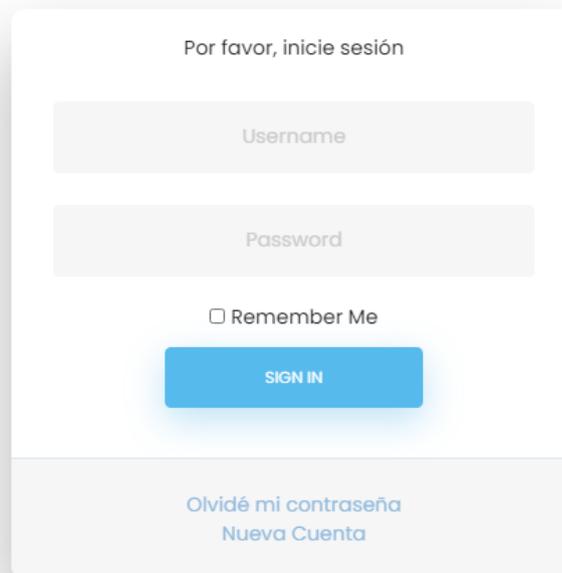
Administrador: Tiene permisos globales en el sistema.

Secretario: Tiene permiso para el manejo de la agenda de turnos de un consultorio, gestión de obras sociales, etc.

Kinesiólogo: Rol que permite el manejo de los turnos, gestión de tratamientos, variables de seguimiento.

Usuario: Tiene permisos de lectura para utilizar el sistema, no puede editar la información dentro del sistema.

Healmate



Por favor, inicie sesión

Username

Password

Remember Me

SIGN IN

[Olvidé mi contraseña](#)
[Nueva Cuenta](#)

Editar Usuario Habilitado

Nombre de Usuario *

Email *

Roles *

ROLE_ADMIN

ROLE_SECRETARY

ROLE_KINESIOLOGY

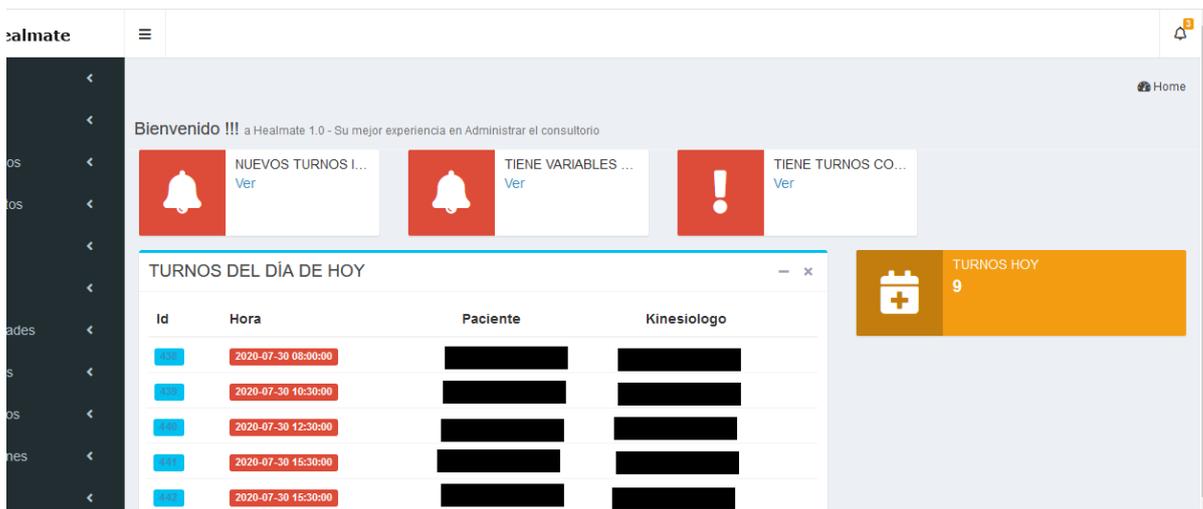
ROLE_USER

Eliminar
Cancelar
Guardar

4.2.2 Ingreso al sistema

Luego de autenticarse con las credenciales de usuarios mencionadas anteriormente, el sistema brinda una vista resumida del calendario del Consultorio en el día de la fecha. A la izquierda cuenta con una columna en la que se agrupan todos los módulos disponibles para el usuario autenticado.

En vista inicial (Dashboard) se pueden ver los turnos del día actual y las notificaciones, pudiendo acceder desde el identificador de cada turno para ver la información completa que contiene cada uno de ellos.



The dashboard shows a welcome message: "Bienvenido !!! a Healmate 1.0 - Su mejor experiencia en Administrar el consultorio". Below this are three notification cards: "NUEVOS TURNOS I...", "TIENE VARIABLES ...", and "TIENE TURNOS CO...". The main section is titled "TURNOS DEL DÍA DE HOY" and contains a table with the following data:

Id	Hora	Paciente	Kinesiólogo
1001	2020-07-30 08:00:00	[Redacted]	[Redacted]
1002	2020-07-30 10:30:00	[Redacted]	[Redacted]
1003	2020-07-30 12:30:00	[Redacted]	[Redacted]
1004	2020-07-30 15:30:00	[Redacted]	[Redacted]
1005	2020-07-30 15:30:00	[Redacted]	[Redacted]

On the right side, there is a "TURNOS HOY" widget showing a calendar icon and the number "9".

4.2.3 Consultorio, Kinesiólogos y Especialidades

Estas tres entidades se encargan de definir el entorno donde se aplica el sistema, siendo por ello las más importantes. El consultorio es la institución donde entablan su tarea laboral los Kinesiólogos, los que tienen asociadas una o varias especialidades profesionales.

Estas tres unidades básicas tienen sus respectivos módulos donde pueden ser creadas, editadas o eliminadas del sistema.

De aquí se inicia la configuración del software de gestión del consultorio. Primer paso al inicio es tener cargada esta información ya que son necesarias para completar los datos indispensables del resto de la aplicación.

En las siguientes figuras se muestran los diferentes datos que deben proporcionarse relativos a estos componentes del sistema.

Home > Consultorios > Biokinetic

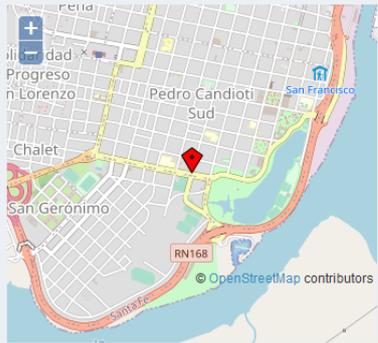
Consultorio Habilitado

Nombre	Biokinetic	Código	BIO
Teléfono	34534534	Email	biokinetic@yopmail.com

Dirección

Calle	4 de Enero	Número	1899
Localidad	Santa Fe	Código Postal	3000
Provincia	Santa Fe	País	Argentina

Volver Editar



Home > Kinesiólogos > Milesi, Marcos

Kinesiólogo Habilitado

Nombre	Marcos	Apellido	Milesi
DNI	12345678	Matricula	ABC123
Especialidades Médicas	Traumatología Ortopedia	Nacionalidad	Argentina
Fecha de Nacimiento	05/09/2019	Email	marcosmilesi@yopmail.com

Volver Editar

Home > Especialidades > Todas

Especialidades Médicas + Crear nuevo

Buscar

Q Buscar Reselear Filtar

Exportar a PDF Exportar a Excel

#	Nombre	Código	Alias	Acciones
2	Kinesiología y Fisiatría	01	FK	Ver Editar
3	Reeducación Postural Global	02	RPG	Ver Editar

Elementos mostrados 1 - 2 de 2

Home > Especialidades > Traumatología

Especialidad Médica

Nombre	Código	Alias
Traumatología	TR	TRA

Delete
Volver Editar

4.2.4 Recursos

Los recursos son los diferentes elementos de un Consultorio que serán ocupados en la atención de los Pacientes. Un tratamiento puede ser definido mediante un proceso que tiene un determinado tiempo de duración, el cual es definido por el profesional encargado de llevar adelante este tratamiento y que utiliza una cantidad dada de estos Recursos pertenecientes al Consultorio.

El sistema permite la definición de Tipos de Recursos que dependen de la organización de cada institución. En este caso podemos enumerar los siguientes que fueron requeridos por el propietario: Box, Gimnasio, Camilla, Magneto, Ultrasonido, etc.

Los distintos elementos de cada uno de estos tipos de recursos se deben cargar uno a uno para poder generar el total de recursos disponibles. Ellos limitarán y servirán para automatizar la presentación de espacios de tiempo para otorgar turnos a los pacientes, junto con otros parámetros a tener en cuenta respecto del tratamiento en cuestión.

Un aspecto a tener en cuenta aquí es que el kinesiólogo es tomado como un recurso a la hora de realizar el análisis de disponibilidades en el proceso de automatización de los turnos disponibles.

Home > Recursos > Todos

Recursos

+ Crear nuevo

Buscar: Consultorio: Todos Tipo: Todos Operativo: Todos

Reseteo Filtros

Exportar a PDF Exportar a Excel

#	Nro de Serie	Tipo	Cod Inventario	Consultorio	Oper...	Acciones
1	14921115296	Magneto	CEC-MAG3 1	Biokinetic Sport	Si	Ver Editar
2	30340616365	Magneto	CEC-MAG3 2	Biokinetic Sport	Si	Ver Editar
3	568008120340	Magneto	MEDITEA-MANETHERP330 1	Biokinetic Sport	Si	Ver Editar
4	569909110347	Ultrasonido	MEDITEA-SONOTHERP330 1	Biokinetic Sport	Si	Ver Editar
5	63100816400	Ultrasonido	CEC-USM1 1	Biokinetic Sport	Si	Ver Editar
6	569006170326	Onda Corta	MEDITEA-ONDA CORTA PULSATIL 1	Biokinetic Sport	Si	Ver Editar
7	58140616374	Onda Corta	CEC-M8 1	Biokinetic Sport	Si	Ver Editar
8	58140616374	Onda Corta	CEC-M8 1	Biokinetic Sport	Si	Ver Editar
9	68560716358	Tens - Ondas Rusas	CEC-Ondas Rusas T 1	Biokinetic Sport	Si	Ver Editar
10	99310716385	Tens - Ondas Rusas	CEC-Ondas Rusas T 2	Biokinetic Sport	Si	Ver Editar

Elementos mostrados 1 - 10 de 19

Home > Recursos > Recursos > Magneto - MEDITEA-MANETHERP330 1

Detalle Recurso

Habilitado

Número de Serie 568008120340	Modelo Magneto
Código Inventario MEDITEA-MANETHERP330 1	Consultorio Biokinetic Sport

Volver Editar

Tipo de Recurso

Nombre Magneto	Icono 	Sin imagen
Tipo Dispositivo	Familia Dispositivo Eléctrico	

Edita

Home > Recursos > Tipos de Recursos > Todas

Tipo de Recurso + Crear nuevo

Buscar Reseteo Filtros

Exportar a PDF Exportar a Excel

#	Nombre	Tipo	Familia	Icono	Imagen	Acciones
32	Magneto	Dispositivo	Dispositivo Eléctrico			Ver Editar
33	Ultrasonido	Dispositivo	Dispositivo Eléctrico			Ver Editar
34	Onda Corta	Dispositivo	Dispositivo Eléctrico			Ver Editar
35	Tens - Ondas Rusas	Dispositivo	Dispositivo Eléctrico			Ver Editar
36	Lámpara Infrarroja	Dispositivo	Dispositivo Eléctrico			Ver Editar
37	Gimnasio	Espacio Físico	Habitación			Ver Editar
38	Box	Espacio Físico	Habitación			Ver Editar
39	Camilla	Dispositivo	Mueble			Ver Editar

Elementos mostrados 1 - 8 de 8

4.2.5 Tratamientos

Una Patología es la definición del diagnóstico detectado por un profesional y en el consultorio se apunta a la rehabilitación del paciente. Es por ello que el sistema permite la administración de las patologías en su base de datos. Es una entidad simple que sólo tiene un nombre que la identifica y una descripción que acompaña al diagnóstico.

Un tratamiento es la forma mediante la cual se trabaja para tratar una patología. No existe una sola forma de trabajar con una patología en particular por lo que puede existir más de un tratamiento asociado con una patología. Esto es así para darle la flexibilidad al sistema de adaptarse a las diferentes lesiones o diagnósticos de un paciente.

Por último, dentro de este módulo, está a la definición de las Variables de Seguimiento, que están compuestas por un nombre y una unidad de medida establecida por el profesional que dependerá de las prácticas que realice y la forma de analizar a sus pacientes.

Estas variables pueden asociarse a los tratamientos, lo que brinda la posibilidad de realizar un análisis mediante la generación de informes o el seguimiento en la rehabilitación del paciente.

Home > Tratamientos > Tratamientos > Todos

Plantillas de Tratamientos + Crear nuevo

Search Reseteo Filtros

Exportar a PDF Exportar a Excel

#	Nombre	Sesiones	Código	Patologías	Color	Acciones
1	Tratamiento de Rodillas	8	ROD	Tendinitis Fracturas	#770892	Ver Editar
2	Masaje muscular	0	MASA	Tendinitis Esguinces	#002F0F	Ver Editar
3	Gimnasio	0	GIM	Tendinitis Hernias discales	#FF0001	Ver Editar
4	Estandar	1	1		#00FF00	Ver Editar
5	Recuperación de desgarros	3	ASKK	Tendinitis	#002FFF	Ver Editar
6	hhhh	1	hhh	Tendinitis	#000000	Ver Editar

Elementos mostrados 1 - 6 de 6

4.2.6 Obras Sociales

La gestión de las Obras sociales está directamente relacionada con la información de aranceles por los tratamientos realizados a los pacientes. Cuando se da el alta de un paciente se permite seleccionar la obra social y el plan al que corresponde en el caso que sea necesario. Además, cuenta con los datos de contacto de la O.S.

Home > O. Sociales > Jerárquicos Salud

Obra Social Habilitado

Nombre	Código
Jerárquicos Salud	1221
Teléfono	Email
4553417	jerarquicos@yopmail.com

Volver Editar

ÓRDENES NO PRESENTADAS

0

No hay Órdenes sin presentar

PENDIENTE COBRAR

888,99 HACER!!!

Aranceles de Tratamientos

#	Tratamiento	Válido desde	Nombre	Remuneración por Sesión	Acciones
4	Estandar		Sin Aranceles cargados		Cargar
3	Gimnasio		Sin Aranceles cargados		Cargar
6	hhhh		Sin Aranceles cargados		Cargar
2	Masaje muscular		Sin Aranceles cargados		Cargar
5	Recuperación de desgarros		Sin Aranceles cargados		Cargar
1	Tratamiento de Rodillas		Sin Aranceles cargados		Cargar

Órdenes Médicas Pendientes Presentar

No hay Órdenes pendientes de presentar

4.2.7 Pacientes

La gestión de los pacientes permite darles de alta, modificar datos y eliminarlos del sistema. Estos se asocian directamente con todo lo descrito anteriormente.

Estos pacientes pueden tener una o varias Obras sociales asociadas o atenderse mediante el pago de las sesiones particularmente.

Los mismos pueden tener asociada una o varias fichas de rehabilitación, dentro de las cuales se detallan los tratamientos, órdenes médicas presentadas, la obra social con la que se encuentra asociado y un historial de las que tuvo en el transcurso de su atención en Biokinetic.

Desde la ficha de cada paciente se puede ver toda su Historia clínica, incluso desde allí dirigirse al módulo correspondiente para la edición o carga de nueva información sobre los tratamientos, obras sociales, etc.

Paciente	
Nombre	Apellido
Victor	Rosset
Fecha de Nacimiento	DNI
24/04/1984	30587235
Nacionalidad	Email
Argentina	victorrosset@gmail.com

[Volver](#) [Editar](#)

\$	CUENTA CORRIENTE 0,00 El Paciente no adeuda nada
+	TURNOS PENDIENTES 0 El Paciente no posee turnos pendi...
+	OBRA SOCIAL IAPOS - pmi 1000 No hay Órdenes sin Presentar

4.2.8 Turnos

Una de las principales funcionalidades apuntadas en esta solución son los turnos, la automatización y facilidad de asignar turnos y ver los blancos disponibles en el calendario para poder operar tanto rápida como intuitivamente.

El sistema da la posibilidad de trabajar de dos formas diferentes:

1. Turnos Rápidos: por pedido del cliente se desarrolló un procedimiento que permite la rápida generación de un turno en el calendario que sólo reserva el día y horario. Esto se hace sin la necesidad de seleccionar un paciente, sólo agregando una descripción, que podría ser el nombre del paciente o cualquier dato relacionado que permita usarse luego para la posterior carga de la información restante. Esto surge por la necesidad del cliente de una rápida atención del paciente a la hora de solicitar el primer turno. Poca información se tiene del diagnóstico o la necesidad del mismo lo que requiere que la primera visita se dé y allí obtener la información necesaria para generar la reserva de los turnos, recursos necesarios y la rehabilitación adecuada.
2. Turnos: En la parte de turnos convencionales se puede realizar la carga de turnos individuales seleccionando el Tratamiento que debe realizar el paciente, un día y hora de inicio y fin. Estos se muestran en la vista principal de los turnos, en una tabla con los datos principales. Desde el listado puede accederse a ver los datos completos de un turno, editar o eliminar turnos.

4.2.9 Rehabilitaciones

La ficha de Rehabilitación de un Paciente cuenta con los datos acerca de su Obra social, Patología, Tratamiento a recibir, Kinesiólogo que lo atenderá y las sesiones

indicadas. También se puede realizar la carga de los turnos en forma masiva, es decir, un Paciente que tenga que realizar un Tratamiento que conste de 5 Sesiones podrá reservar de una sola vez los 5 turnos que corresponden, con la facilidad de seleccionar los días hábiles de la semana en los que distribuirá los turnos, ej. lunes, miércoles y viernes. Al finalizar la carga de datos, el sistema automatiza la generación de cada Turno en el Calendario y reserva de Recursos de los días y horarios elegidos. Desde la vista principal de Rehabilitaciones pueden verse todas las instancias de rehabilitación que se han generado, con la posibilidad de aplicar diferentes filtros y acceder según la elección del usuario al listado necesario. Desde cada una de estas instancias se puede ver el contenido de la Rehabilitación, que además de los datos que se cargan, muestra las asociaciones del tratamiento con una Orden médica, si es que la hubiera, con los datos del médico, indicaciones, etc., y un listado de los turnos separados por día con el detalle de haber sido completado o no. Esta información está disponible para editarse o eliminarse desde aquí.

4.2.10 Orden Médica

Las Rehabilitaciones en un gran porcentaje de casos vienen acompañadas de una derivación de un profesional especialista, el cual diagnostica e indica al Kinesiólogo un Tratamiento y las sesiones recomendadas. Esta información viene documentada en la Orden Médica. Como mencionamos anteriormente, en la carga de Turnos Rápidos, el paciente llega con una orden a la primera visita. Una vez revisado el Paciente es cuando se puede completar la Ficha de Rehabilitación donde el profesional carga la información completa del Tratamiento a realizar.

Luego de la creación de la instancia de la rehabilitación se puede generar la Orden Médica en el sistema, desde la vista de Rehabilitaciones.

Órdenes Médica

Buscar Obra Social

Consentimiento Habilitada Enviada Cobrada Reseteo Filtrar

#	Paciente	Indicación	Sesiones	Fecha Orden	Consentimiento	Habilitada	Enviada	Cobrada	Acciones
1	[Redacted]	[Redacted]	[Redacted]	31/12/1969	Si	Si	No	Si	[Icons]
2	[Redacted]	[Redacted]	1	31/12/1969	Si	No	Si	No	[Icons]
3	[Redacted]	[Redacted]	1	31/12/1969	Si	Si	No	No	[Icons]
4	[Redacted]	[Redacted]	[Redacted]	31/12/1969	No	Si	Si	No	[Icons]

En la vista de Órdenes Médicas podrá gestionar las mismas. Ya sea enviar a cobrar, asentar el cobro, editarlas o verlas en detalle.

Para enviar a cobrar un orden podrá utilizar el nuevo método de aranceles por obra social generalizado en caso de ser necesario, mediante botón  o bien, en caso de necesitar especificidad, mediante el arancel por tratamiento específico.

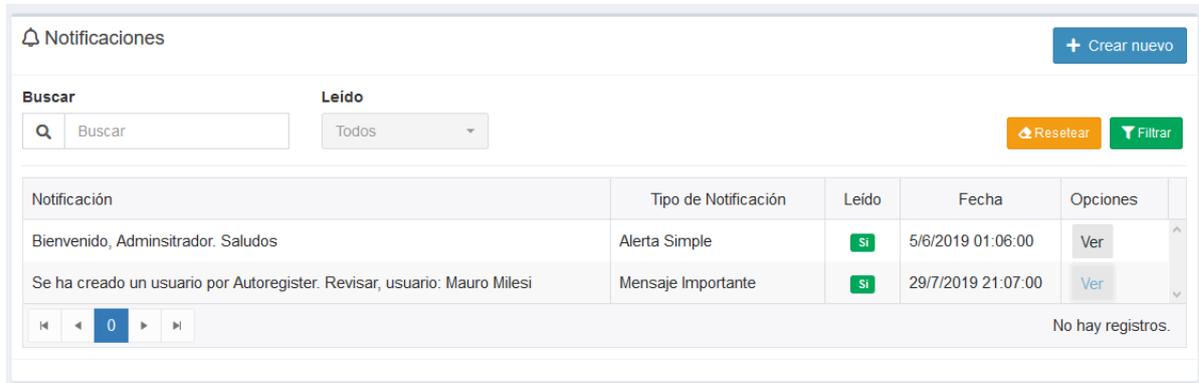
A modo de ejemplo:

Se ha recuperado la siguiente información

N° Ses.	<input type="text" value="123"/>	ORDEN N°	<input type="text" value="[Redacted]"/>
Fisio Kinesio	<input type="text" value="22.3"/>	<input type="button" value="Fisio/Kinesio"/>	
Fisio Kinesio / Internac.	<input type="text" value="22"/>	<input type="button" value="Fisio/Kinesio con Internación"/>	
Fisio Kinesio c/ Magneto	<input type="text" value="45"/>	<input type="button" value="Fisio/Kinesio con Magneto"/>	
Hidroterapia	<input type="text" value="87"/>	<input type="button" value="Hidroterapia"/>	
Neurológicos	<input type="text" value="99"/>	<input type="button" value="Neurológico"/>	
Drenaje Linfático	<input type="text" value="987"/>	<input type="button" value="Drenaje Linfático"/>	
Modulo Respiratorio	<input type="text" value="33"/>	<input type="button" value="Respiratorio"/>	
Domicilio?	<input type="text" value="66"/>	<input type="checkbox"/>	
TOTAL		<input type="text"/>	<input type="button" value="Enviar a Cobrar"/>

4.2.11 Notificaciones

Las notificaciones del sistema se pueden ver y ordenar según diferentes criterios.



The screenshot shows a notification management interface. At the top, there is a search bar labeled 'Buscar' and a dropdown menu for 'Leído' set to 'Todos'. There are buttons for '+ Crear nuevo', 'Resetear', and 'Filtrar'. Below is a table with columns: Notificación, Tipo de Notificación, Leído, Fecha, and Opciones. Two notifications are listed: 'Bienvenido, Adminsitrador. Saludos' (Alerta Simple, Leído: Si, Fecha: 5/6/2019 01:06:00) and 'Se ha creado un usuario por Autoregister. Revisar, usuario: Mauro Milesi' (Mensaje Importante, Leído: Si, Fecha: 29/7/2019 21:07:00). At the bottom, there is a pagination control showing '0' and a message 'No hay registros.'

4.2.12 Informes

Listado de Turnos diarios.



Resumen Diario para el Kinesiólogo Mauro Milesi en la Fecha del 2020-08-06

Paciente	Inicio	Fin	P. Trat.	Patología	Recursos
Ernesto Fernandez	08:30			Contractura Muscular Torax	
Ernesto Fernandez	09:00			Contractura Muscular Torax	MagnetoTens
Joaquin Sereno	09:00			Desgarro de Muscular anterior del Muslo	RusasCamilla
Ariel Barboti	09:00			Contractura Muscular Torax	BoxUltrasonidoMagnetoTens - Ondas RusasCamilla
Marcos Milesi	09:00			Desgarro de Muscular anterior del Muslo	
Mariano Dimagio	12:30			Contractura Muscular Extremidades Inferiores	

Informes de cobranzas

En la lista de las Órdenes Médica podrá acceder a los siguientes informes

 Cobros Último Mes

 Cobros Pend.

 Envíos Pend.

Listado de Órdenes Médicas Cobradas



Listado de Ordenes Médicas Cobradas

El Total es: \$16250

Paciente	Kinesiologo	Patologia	Montos	Fecha Orden	Fecha Enviada	Obra Social
Marcos Milesi	Mauro Milesi	Desgarro de Muscular anterior del Muslo	\$ 5250	2020-07-20	2020-08-05	OSDE BINARIO - Pl. 210
Ernesto Fernandez	Mauro Milesi	Contractura Muscular Torax	\$ 2000	2020-08-04	2020-08-04	IAPOS - DPTO LA CAPITAL
Joaquin Sereno	Mauro Milesi	Desgarro de Muscular anterior del Muslo	\$ 5250	2020-07-04	2020-08-05	OSDE BINARIO - Pl. 210
Ana Bracamonte	Mauro Milesi	Contractura Muscular Cervical	\$ 2000	2020-07-13	2020-08-05	IAPOS - DPTO LA CAPITAL
Mariano Dimaglio	Mauro Milesi	Contractura Muscular Extremidades Inferiores	\$ 1750	2020-08-05	2020-08-06	OSDE BINARIO - Pl. 210

Listado de Órdenes Médicas por Cobrar



Listado de Ordenes Médicas a Cobrar

El Total es: \$10520

Paciente	Kinesiologo	Patologia	Montos	Fecha Orden	Fecha Enviada	Obra Social
Martin Arancio	Mauro Milesi	Contractura Muscular Extremidades Inferiores	\$ 2000	2020-07-20	2020-08-06	IAPOS - DPTO LA CAPITAL
Ariel Barboti	asdf asdf	Contractura Muscular Torax	\$ 2380	2020-07-06	2020-08-06	IAPOS - DPTO LA CAPITAL
Micaela Jordan	Mauro Milesi	Contractura Muscular Extremidades Inferiores	\$ 2010	2020-08-07	2020-08-06	DASUTeN
Manuel Curatolo	Mauro Milesi	Contractura Muscular Torax	\$ 1750	2020-08-06	2020-08-06	OSDE BINARIO - Pl. 210
Federico Rodriguez	Mauro Milesi	Contractura Muscular Cervical	\$ 2380	2020-05-05	2020-08-06	JERARQUICOS SALUD

Listado de Órdenes Médicas por Enviar



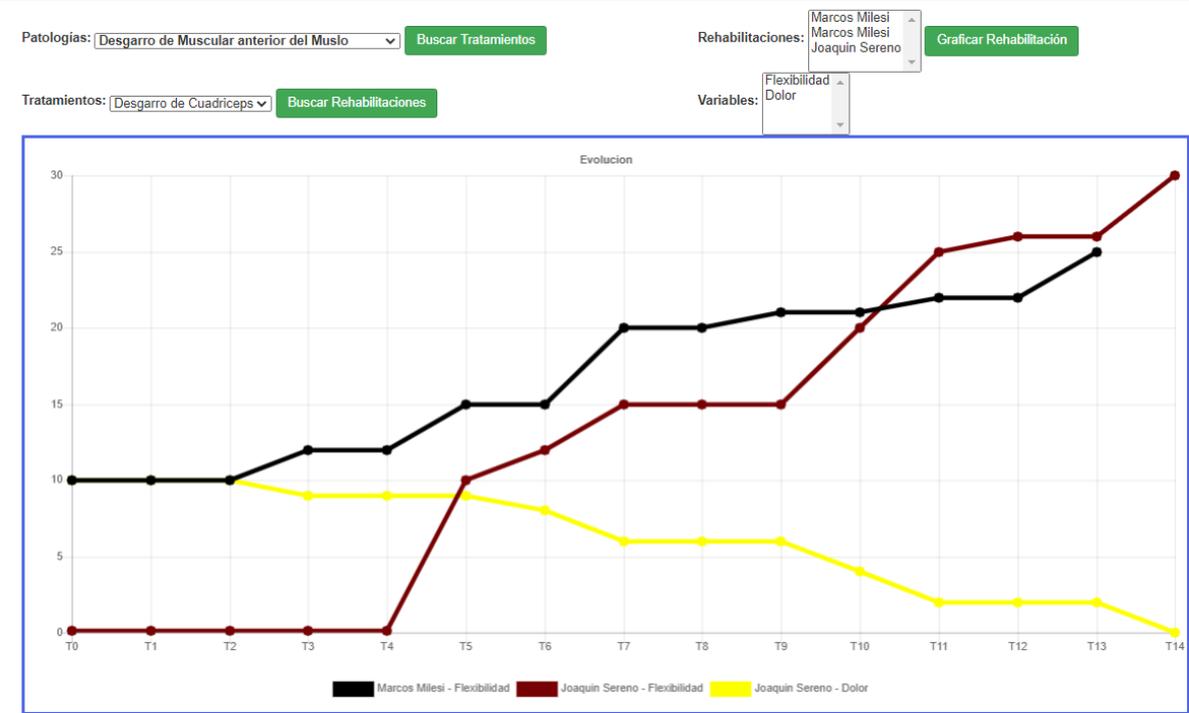
Listado de Ordenes Médicas a Enviar

El Total es: \$NaN

Paciente	Kinesiologo	Patologia	Montos	Fecha Orden	Fecha Enviada	Obra Social
Juan Amador	Mauro Milesi	Contractura Muscular Extremidades Inferiores	\$ null	2020-08-06		IAPOS - DPTO LA CAPITAL
Miguel Juarez	Mauro Milesi	Contractura Muscular Torax	\$ null	2020-08-06		DASUTeN
Gabriel Gonzalez	Mauro Milesi	Contractura Muscular Cervical	\$ null	2020-08-04		OSDE BINARIO - Pl. 210
Natalia Andes	Mauro Milesi	Contractura Muscular Lumbar	\$ null	2020-08-02		JERARQUICOS SALUD
Cristina Castro	Mauro Milesi	Contractura Muscular Cervical	\$ null	2020-08-06		IAPOS - DPTO LA CAPITAL

Informes de Patologías

Gráfico de Resultados



5. Conclusiones y Trabajos Futuros

Este apartado tiene como objetivo presentar los principales aportes y experiencias que se obtuvieron durante la realización de este proyecto final de carrera para trabajos futuros.

5.1 Principales aportes

La realización del Proyecto nos permitió conocer muchas herramientas las cuales se están usando fuertemente en el mercado laboral el día de hoy teniendo una primera experiencia con alguna de ellas. Nos permitió también afianzar y refrescar conocimiento adquirido durante toda la carrera de Ingeniería. Es necesaria mucha práctica en métodos ágiles y la planificación eficiente para organizar el trabajo y cumplir los objetivos a tiempo.

También nos permitió cumplir el ciclo de vida completo del desarrollo de un producto software, no solo análisis, sino también desarrollo e implementación.

5.2 Experiencias

La experiencia más jugosa del proyecto fue la Planificación del trabajo y el seguimiento de éste mediante las herramientas utilizadas. Teniendo en cuenta que el éxito de un proyecto se basa no sólo en la calidad del código fuente sino en lograr que esté en tiempo y forma para el cliente.

La experiencia lograda en la utilización de la metodología ágil fue riquísima, ya que nos permitió mejorar nuestras habilidades y el trabajo en equipo. Se nos presentaron un sin fin de contratiempos y supimos sortearlos con facilidad y negociación.

Fue muy beneficioso el hecho de haber establecido un estándar de codificación muy fuerte al comienzo del proyecto y que cualquiera podría cambiar el código en caso de que eso traiga un beneficio sobre la calidad del mismo. Se produjo una fuerte recodificación en todo momento durante el desarrollo.

5.3 Trabajos Futuros

Como trabajo futuro sobre este proyecto, queda un amplio camino por recorrer:

Buscar nuevos clientes para vender uso del sistema a Consultorios de Kinesiólogos.

Generar una app para Android, para algunos módulos puntuales como cargas de variables o consultas de turnos.

6. Glosario

6.1 Técnico de Consultorio

Kinesiólogo: Profesional de la Salud que estudia el comportamiento del movimiento en el ser humano.

Patología: La palabra patología significa “estudio de la enfermedad”

Tratamiento: En las ciencias en general, tratamiento es el procedimiento aplicado a una o más de las variables independientes que generará o no un cambio en los valores de la(s) variables dependientes, cuya finalidad es la curación o el alivio de las enfermedades o síntomas

Plantilla Tratamiento: Diferentes implementaciones de un tratamiento, en la cual se cambian sus variables de intensidad o metodologías.

Sesiones: Partes, o pasos a seguir dentro de un Tratamiento.

Recurso Modelo: Los Recursos utilizados comúnmente en los tratamientos. Llámese Herramientas, lugares físicos que facilitan las tareas del Kinesiólogo.

Recurso: Instancias de los Recursos Modelos, ya que en un consultorio puede haber varios.

Variables a Medir: Variables definidas por un Kinesiólogo con la intención de medir el tratamiento y su evolución.

Responsive: Técnica de diseño web que busca la correcta visualización de una misma página en distintos dispositivos.

6.2 Técnico Software

Backup

También llamada copia de seguridad o respaldo es una copia de toda la información de un sitio web de forma que, en caso de pérdida o borrado accidental de los datos, pueda recuperarse la información mediante un proceso de restauración de los datos

Burndown:

El gráfico de avance o “burndown” es el gráfico que actualiza el equipo en las reuniones de seguimiento del sprint, para monitorizar el ritmo de avance, y detectar de forma temprana posibles desviaciones sobre la previsión que pudieran comprometer la entrega al final de sprint.

Interfaz de usuario:

La interfaz de usuario es el medio con que el usuario puede comunicarse con una máquina, equipo, computadora o dispositivo, y comprende todos los puntos de contacto entre el usuario y el equipo

Versionado - GIT

Ramas

Una rama representa una línea independiente de desarrollo.

Repositorio

En nuestro caso, Bitbucket, es el lugar donde se almacenan los datos actualizados e históricos de cambios, en el mismo están integrando también los mecanismos de actualización.

Línea base ("baseline")

Son "una especificación o producto que se ha revisado formalmente y sobre los que se ha llegado a un acuerdo, y que de ahí en adelante sirve como base para un desarrollo posterior y que puede cambiarse solamente a través de procedimientos formales de control de cambios" según la traducción del estándar IEEE 610.12-1990 realizada en el libro de Pressman que ya recomendamos.

Por lo tanto, una línea base se define como un producto que acaba de ser aprobado y que define la “base” de ese producto que para ser modificado deberá pasar por un protocolo de control de cambios. También puede verse como un punto de referencia en la configuración de un proyecto que marca un estado estable en algún producto del proyecto.

Abrir rama ("branch") o ramificar

Un módulo puede ser branched o bifurcado en un instante de tiempo de forma que, desde ese momento en adelante se tienen dos copias (ramas) que evolucionan de forma independiente siguiendo su propia línea de desarrollo.

Desplegar

Un despliegue crea una copia de trabajo local desde el repositorio. Se puede especificar una revisión concreta. Por lo general se obtiene la última copia del Repositorio relacionado a la Rama, es un procedimiento típico si se quiere comenzar de cero sobre una rama.

"Publicar" o "Enviar" ("commit")

Un commit sucede cuando una copia de los cambios hechos a una copia local es escrita o integrada sobre el repositorio. Cuando se realiza un commit, se debe realizar una nota aclaratoria especificando los cambios subidos.

Conflicto

Un conflicto ocurre cuando el sistema no puede manejar adecuadamente cambios realizados por dos o más usuarios en un mismo archivo.

Resolver

El acto de la intervención del usuario para atender un conflicto entre diferentes cambios al mismo archivo.

Integración o fusión ("merge")

Una integración o fusión une dos conjuntos de cambios sobre un fichero o un conjunto de ficheros en una revisión unificada de dicho fichero o ficheros.

Mock

Son objetos que se usan para probar que se realizan correctamente llamadas a otros métodos, se utilizan para verificar el comportamiento de los objetos.

7. Bibliografía

- Jeff Sutherland - "SCRUM - The Art of doing Twice the Work in Half the Time", Año 2014
- Beck, K. - "Extreme Programming Explained. Embrace Change", Addison Wesley, 2000.
- Highsmith, J. - "Agile Software Development Ecosystems". Addison-Wesley. 2002
- Jeffries, Anderson, Hendrickson, "Extreme Programming Installed". Addison-Wesley. 2001
- Beneficios de Scrum: <https://proyectosagiles.org/beneficios-de-scrum/>
- Riesgos de la metodología Scrum: <https://www.obs-edu.com/int/blog-project-management/scrum/los-riesgos-de-la-metodologia-scrum>
- Programación Extrema: www.extremeprogramming.org
- Gráfico de Avance: https://www.scrummanager.net/bok/index.php?title=Gr%C3%A1fico_de_avance

8 Anexos

En este Capítulo están contenidos los anexos mencionados en el Proyecto.

8.1 Análisis Inicial de Habilidades

Fue necesario realizar un análisis inicial de habilidades para poder planificar el desarrollo del proyecto y los Roles a asignar en cada sprint.

Consideramos utilizar una escala de colores, para ver gráficamente cuando tendríamos que reforzar trabajo o colaborar más activamente:

Escala:

Buena	Media	Baja	Nada
-------	-------	------	------

Resultado:

	Inalbon	Rosset	Milesi	RESULTADO GRUPAL
Métodos Ágiles	Buena	Buena	Baja	Media
Análisis de Requerimientos	Buena	Buena	Buena	Buena
Diseño	Media	Media	Baja	Baja
Planificación	Baja	Media	Media	Baja
Testing	Baja	Baja	Media	Baja
Frameworks a utilizar para el desarrollo	Media	Buena	Baja	Media
Disponibilidad horaria semanal	Baja	Baja	Baja	Baja
Lenguaje: PHP	Media	Buena	Baja	Media
Lenguaje: Javascript	Media	Buena	Baja	Media
Enmaquetado de Páginas	Media	Baja	Media	Baja

8.2 Análisis para elección de Calendario

Se analizaron varias opciones compatibles con Symfony 3.4.

La decisión estuvo entre dhtmlxScheduler y FullCalendar. Uno de los integrantes del grupo tenía experiencia trabajando con FullCalendar en sus versiones anteriores.

La versión analizada fue la v4, la cual incorpora aún más funcionalidades a las existentes en las versiones previas.

Entre los puntos fuertes:

- Permite personalizar los eventos (colores, incorporar imágenes, drag and drop)
- Existen varios templates.
- En la página oficial existe documentación (<https://fullcalendar.io/docs>)
- Hay traducciones a todos los idiomas.
- Es gratis.
- Soporta diferentes formas de conexión de eventos.
- Permite definir horarios laborales y otras cuestiones puntuales muy útiles.

Optamos por tattali/calendar-bundle

8.3 Ejemplos de Minutas de Reunión:

Participantes

- [Nelson Inalbon](#)
- [Matías Milesi](#)
- [Víctor Rosset](#)

Objetivos

Definir separación de funcionalidades a desarrollar en el sprint/sprint.

- Definir grupo de trabajo para el sprint.
- Definir funcionalidades a desarrollar por cada grupo en el sprint.
- Definir el GitFlow.

Temas de Discusión:

Tiempo	Ítem	Presentador	Notas
5 min	Separar Funcionalidades	Nelson Inalbon	· Negociación de las funcionalidades
5 min	Asignación de Grupos de Trabajo	Víctor Rosset	Se define que Nelson Inalbon y Matías Milesi realizarán programación en equipo para que Matías Milesi adquiere un poco de experiencia. Víctor Rosset estará en el otro grupo de trabajo
10 min	Asignación de Funcionalidades a Grupo de Trabajo	Matías Milesi	Víctor Rosset se encargará de: <ul style="list-style-type: none"> · Implementación de Usuarios · Implementación de Roles Matías Milesi y Nelson Inalbon se encargarán de: <ul style="list-style-type: none"> · Interfaz de Usuario · Implementación de Consultorio. Matías Milesi iniciará Testing.

Expectativas:

- Proactividad de cada uno de los Integrantes, se espera que cada uno trabaje sobre los puntos Planteados

Decisiones:

- Definición de la barra Lateral por AdminLTE por votación y análisis de comparación presentado por los integrantes.

8.4 Historia Clínica vs Ficha de Turno

Análisis de Implementación de Historia Clínica vs Ficha de Turno. Comparativas planteadas con la finalidad de analizar la forma más elegante y mantenible de implementar una solución al requerimiento planteado por el cliente.

Cambio conceptual de Definición de TRATAMIENTO (solo asociado a una patología) vs Rehabilitación (una serie de trabajos sobre un paciente)

Éste modelado no permitía realizar un correcto seguimiento de una Rehabilitación, ya que Biokinetic Sport quería asociar:

Obj 1	Obj 2
El Turno	Un Resultado de ese Turno, es decir, el valor asignado a las Variables de Seguimiento del Tratamiento en cuestión, obtenidas del análisis del estado del Paciente en dicho Turno.
Una Orden Médica	Una Orden Médica, indica en sesiones, la cantidad de Turnos que requiere el paciente, lo que marca el inicio, y posiblemente el fin de un Tratamiento.
Una Rehabilitación	A una o Varias Órdenes Médicas, ya que dependiendo de qué Tratamiento se trate, existe la posibilidad de tener más de una Orden Médica para el Paciente.

- Es por esto, que podría obtener de una Rehabilitación la información de cada uno de los turnos y los resultados de estos. Pero lo más importante, permitir

obtener conclusiones de una Rehabilitación y así también obtener conclusiones de las Órdenes Médicas.

Por esto se mejoró el modelado agregando más Clases que permitan realizar esta Trazabilidad y mejorar así la Historia Clínica.

Ficha de Turno: Resultado y Evaluación del Turno (Plantilla de Historia Clínica)

Ficha Rehabilitación: Asociado a un paciente, a una orden médica

Rehabilitación: Una o Varias Fichas de Rehabilitación (Cuando es una sola, se toma las conclusiones de la Ficha, para evitar doble carga)

8.5 Tratamientos VS Plantilla Tratamientos

Análisis de Implementación de Tratamientos vs Plantilla de Tratamientos. Comparativas planteadas con la finalidad de analizar la forma más elegante y mantenible de implementar una solución al requerimiento planteado por el cliente.

Encargado: Nelson Inalbon

Pulido de Requerimiento y Funcionalidades

Información Importante:

La Plantilla de Tratamiento son los Protocolos a seguir (Tareas y Recursos que ocupan). Tener en cuenta que los Recursos son ESPACIALES y de UTILIZACIÓN (Ejemplo Espacial: BOX o Gym; Ejemplo UTILIZACIÓN: Magneto).

Ejemplo:

Lumbalgia Intensivo: Magneto (50% no demandante), Lámpara (25% no demandante) y Gimnasio (25% demandante)

Lumbalgia Leve: Ultrasonido (50 % demandante), Magneto (25% no demandante) y Masaje (25% demandante)

Opciones consideradas:

	Option 1: Programación y asignación de Turnos Simple	Option 2: Programación y asignación de Turnos con modelado de Plantillas de Tratamientos
Description	Sería simple asignar los turnos desde el primer día. Ya que habría solo una opción Tratamiento para una Patología. Más simple de programar.	Asignación luego del primer turno, permitirle al Kinesiólogo elegir si sigue con el mismo tratamiento. En la Primer turno podría asignar los 10 Turnos con una plantilla. El Kinesiólogo luego de la primer reunión podría cambiar de Plantilla de Tratamiento.
Pros and cons	<ul style="list-style-type: none"> + Fácil Programación para la Secretaria - Sistema Limitado 	<ul style="list-style-type: none"> + Definir mejores tratamientos según el Pacientes + El paciente habría consensuado los horarios con el Kinesiólogo. + Podría elegirse una plantilla Predeterminada según la Orden Medica. - Implicaría mayor trabajo Administrativo
Estimated cost	MEDIUM	MEDIUM

8.6 Requerimientos funcionales

Para todos los requerimientos se definieron las siguientes Métricas de Éxito:

Goal	Metric
Velocidad Esperada en la Story	Velocidad planificada
Velocidad Alcanzada en la Story	Velocidad real
Story points alcanzados	Story points reales

A continuación, detallamos los requisitos que formaron parte de este proyecto, los cuales fueron la guía de avance. Los mismos se encuentran en el orden que se le dio en los 9 *sprints* que se realizaron en el transcurso del desarrollo.

8.6.01 Modelo Básico de Interfaz

Objetivo

- El sistema debe tener una interfaz clara para el usuario. y presentar los datos de forma ordenada y elegante.
- La interfaz deberá brindar al usuario la posibilidad de trabajar con Formularios, Notificaciones, Calendario, Tablas, etc.
- El usuario debe poder ingresar al sistema mediante nombre de usuario y contraseña.
- El sistema deberá trabajar con Roles Admin, Usuario, Kinesiólogo y Secretaria.

Tareas

TAREAS	REQUERIMIENTOS	PRIORIDAD	TASK POINTS
Configuración Layout	Se debe definir una interfaz amigable	MEDIA	4
Configuración de Elementos de Interfaz de Usuario	Configuración de elementos de interfaz de usuario	ALTA	7
Configuración de Formularios	Configuración de Formularios de datos	ALTA	7
Configuración de Notificaciones	Configuración de Notificaciones en la interfaz de usuario	MEDIA	4
Configuración de Calendario	Configuración de Calendario para las funcionalidades necesarias del sistema	MEDIA	5
Configuración de Tablas	Configuración de tablas en el sistema para la presentación de información	BAJA	3
Loguin de usuarios	Se debe ingresar al sistema mediante credenciales de usuario	ALTA	8

8.6.02 ABM de Consultorios

Objetivo

- El sistema debe permitir la creación de un Consultorio.
- El sistema debe permitir al usuario la búsqueda de un consultorio, modificarlo y eliminarlo.
- El usuario debe poder ingresar al sistema los datos del Consultorio en una forma simple y rápida.
- El usuario debe poder ingresar al sistema los datos del Consultorio en una forma simple y rápida.

Datos a completar:

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Nombre	String	SI	
Código	String	SI	
Teléfono	Array	SI	Formato Teléfono
Email	String	SI	Formato Email

Supuestos

- Es necesario que el usuario se haya logeado al sistema.
- Es necesario que el usuario tenga el rol de Administrador.

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de Consultorios	El usuario podrá crear un Consultorio e ingresar los Datos mencionados en la tabla	ALTA	7
Baja de Consultorios	El usuario podrá eliminar un Consultorio siempre que no tenga Kinesiólogos ni Recursos asignados.	BAJA	2
Modificación de Consultorios	El usuario podrá modificar los Datos del Consultorio	MEDIA	4
Mostrar Consultorio	El usuario podrá ver los datos del Consultorio.	MEDIA	4
Búsqueda de Consultorios	El usuario podrá buscar un Consultorio según sus datos.	MEDIA	6
Interfaz de Consultorios	Se deberá crear una interfaz amigable e intuitiva	MEDIA	5
Testing del Módulo de Consultorios	Es necesario crear pruebas de unidad y funcionales para analizar las entidades comprendidas y la funcionalidad del módulo.	ALTA	7

Interacción con el usuario y diseño



The screenshot shows a web browser window with the URL <https://www.draw.io>. The page title is "Nuevo Consultorio". The form contains four input fields: "Nombre", "Codigo", "Telefono", and "Email". At the bottom right of the form, there are two buttons: "Cancelar" (red) and "Crear" (green).

8.6.03 ABM de Usuario

Objetivo

- Instalar y configurar FOSUser Bundle en Symfony.
- El sistema debe permitir cargar usuarios y asignarles roles o perfil para la utilización del sistema.
- El sistema debe permitir la búsqueda de usuarios, modificarlos, cambiar el perfil, y eliminar usuarios.
- El sistema debe permitir la gestión de los roles.
- El sistema será seguro y diferenciará las funcionalidades según los roles.

Datos a completar:

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Nombre de usuario	String	SI	
Contraseña	String	SI	PasswordType
Repetir Contraseña	String	SI	PasswordType
Email	String	NO	Formato Email
Roles	Array	SI	

Supuestos

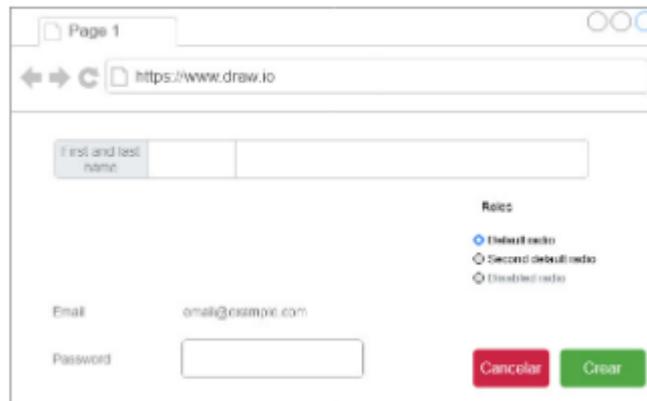
- Necesitamos tener instalado y corriendo docker para comenzar con la instalación de Symfony
- Necesitamos tener una instancia de Symfony instalada y configurada.
- Es necesario que el usuario se haya logeado al sistema para trabajar con los usuarios y roles.
- Es necesario que el usuario tenga el rol administrador para poder funcionalidades de este módulo.

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de Usuarios	Es necesario poder registrar usuarios en el sistema	ALTA	8
Modificación de Usuarios	Es necesario poder editar usuarios	MEDIA	5
Baja de Usuarios	Es necesario poder eliminar usuarios	BAJA	3

Búsqueda de Usuarios	El usuario podrá buscar los datos de los usuarios en el sistema	MEDIA	6
Mostrar Usuario	El usuario podrá ver los datos de los usuarios	MEDIA	5
Diseño de Interfaz	Es necesario dejar una interfaz clara y sencilla.	MEDIA	5
Testing del Módulo de Usuarios	Es necesario crear pruebas de unidad y funcionales para analizar las entidades comprendidas y la funcionalidad del módulo.	ALTA	7

Interacción con el usuario y diseño



8.6.04 ABM de Pacientes

Objetivo

- El sistema debe permitir cargar pacientes.
- El sistema debe permitir la búsqueda intuitiva y sencilla, edición y eliminación de pacientes.
- El usuario debe poder ingresar al sistema los datos del Paciente en una forma simple y rápida.
- El paciente puede estar asociado a una Obra Social o no.
- El sistema permitirá hacer cálculos de las tarifas según la Obra Social del paciente.

Datos a completar:

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Nombre	String	SI	
Apellido	String	SI	
Fecha de Nacimiento	Date Time	SI	
DNI	String	SI	
Nacionalidad	Country	SI	
Email	String	SI	
Obra Social	ObraSocial	NO	Podría no tener Obra Social

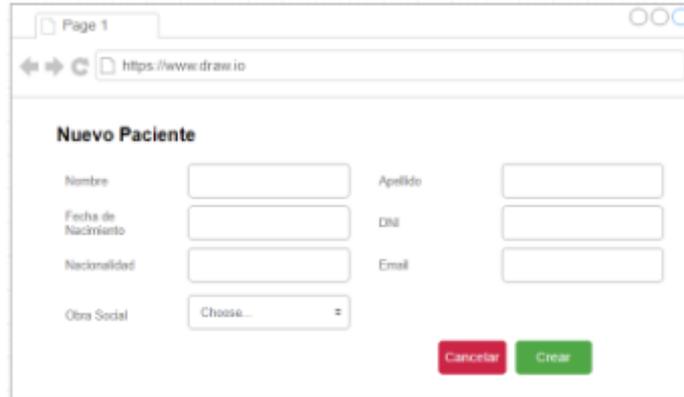
Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo, Secretario, Administrador.

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de Pacientes	El usuario podrá crear Pacientes e ingresar los Datos mencionados en la tabla	ALTA	7
Baja de Pacientes	El usuario podrá eliminar Pacientes.	BAJA	2
Modificación de Pacientes	El usuario podrá modificar los Datos de Pacientes.	MEDIA	4
Mostrar Paciente	El usuario podrá ver los datos de un Paciente.	MEDIA	4
Búsqueda de Paciente	El usuario podrá buscar un Paciente para acceder a su información.	MEDIA	6
Interfaz de Paciente	Se deberá crear una interfaz amigable e intuitiva	MEDIA	5
Testing del Módulo de Paciente	Implementar pruebas unitarias y funcionales para el módulo.	ALTA	7

Interacción con el usuario y diseño



The image shows a web browser window displaying a form titled "Nuevo Paciente". The form contains the following fields: "Nombre" (text input), "Apellido" (text input), "Fecha de Nacimiento" (date input), "DNI" (text input), "Nacionalidad" (text input), "Email" (text input), and "Otra Social" (dropdown menu with "Chase..." selected). At the bottom right of the form are two buttons: "Cancelar" (red) and "Crear" (green).

8.6.05 ABM de Turnos

Objetivos

- El sistema debe permitir agregar, modificar o eliminar Turnos.
- El sistema debe permitir buscar Turnos de forma intuitiva.
- El sistema debe permitir la carga de Turnos rápidos
- El sistema deberá analizar la factibilidad del turno al momento de ser seleccionado, buscando los recursos disponibles y analizando si los recursos a utilizar estarán disponibles para ese Turno. La asignación y sugerencia del Turno, deberá contemplar Recursos disponibles en el Consultorio para un día y una hora específica.
- El sistema debe permitir asignar turnos siempre, a pesar de que el consultorio no tenga los recursos necesarios para hacerle frente. En este caso, el sistema deberá informar de esta irregularidad.
- El sistema debe permitir asignar un turno en una vista tipo calendario.
- El sistema debe permitir acceder a los turnos mediante el calendario.
- Un Turno debe estar asociado a un Paciente, el mismo puede o no tener Obra Social.
- El sistema debe permitir crear un turno sin la información del Paciente. Es muy recomendable luego de realizado el turno asociar al mismo a un Paciente, si es que no se asoció en primera instancia.

Datos a Completar:

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Fecha	Date Time	SI	
Días	Array	SI	
Hora	Time	SI	
Paciente	String	SI	
Tratamiento	Tratamiento	SI	
Patología	Patología	SI	
Recursos	Recursos	NO	Puede reservarse según el Tratamiento asociado
Obra Social	ObraSocial	NO	Podría no tener Obra Social
Kinesiólogo	Kinesiólogo	SI	
Sesiones	Integer	NO	
Sesiones Realizadas	Integer	NO	

Supuestos

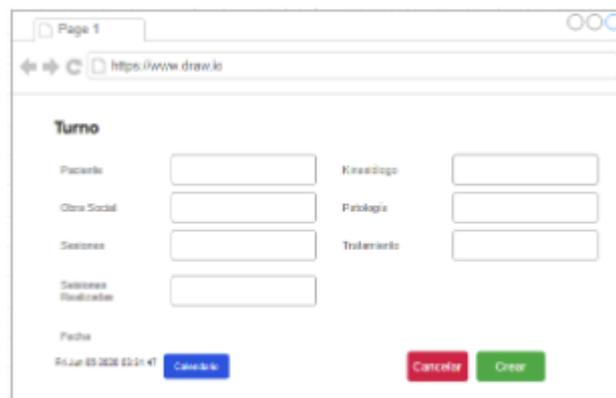
- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Secretario Kinesiólogo

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Crear Turnos rápidos	El usuario podrá crear un Turno rápido	ALTA	6
Buscar Turnos	El usuario podrá realizar una búsqueda intuitiva de turnos rápidos	BAJA	3
Interfaz de Turnos rápidos	Se deberá crear una interfaz intuitiva y amigable	MEDIA	4
Crear Turnos según recursos disponibles	Se deberá crear la lógica para asignar automáticamente los turnos según los recursos disponibles en el consultorio	ALTA	7
Eliminar Turnos liberando recursos	Se deberá realizar la eliminación de un turno verificando que se pongan disponibles los recursos asignados al turno eliminado.	BAJA	2
Modificar Turnos gestionar recursos	Se deberá editar los turnos respetando la disponibilidad de turnos existente	MEDIA	5

Mostrar Turno	El usuario podrá visualizar los datos de un turno.	MEDIA	4
Búsqueda de Turno	El usuario podrá realizar una búsqueda intuitiva de turnos.	MEDIA	6
Adecuar la interfaz de turnos	Se deberá crear una interfaz que complemente la información de turnos, respecto de la ventana de turno rápido	MEDIA	6
Testing del Módulo de Turnos	Implementar pruebas unitarias y funcionales para el módulo.	ALTA	7
Disponibilidad de Recursos	El sistema gestionará la disponibilidad de Recursos para brindar información acerca de los turnos disponibles	MEDIA	6

Interacción con el usuario y diseño

8.6.06 ABM de Kinesiólogos

Objetivo

- El sistema debe permitir cargar Kinesiólogos en una forma simple y rápida.
- El sistema debe permitir al Usuario agregar, modificar o eliminar Kinesiólogos.
- El usuario podrá buscar Kinesiólogos de forma intuitiva.

- El Kinesiólogos es el responsable del turno, y los tratamientos estarán asociados a sus especialidades.
- El Kinesiólogos debe estar asociado a un consultorio y puede tener turnos asociados
- El Kinesiólogos puede tener Especialidades asociadas.

Datos a Completar

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Nombre	String	SI	
Apellido	String	SI	
DNI	String	SI	Formato DNI
Email	String	SI	Formato Email
N° Matricula	String	SI	
Especialidades	Array	SI	
Nacionalidad	String	SI	Formato Nacionalidad
Fecha de Nacimiento	Date	SI	Formato Fecha

Supuestos

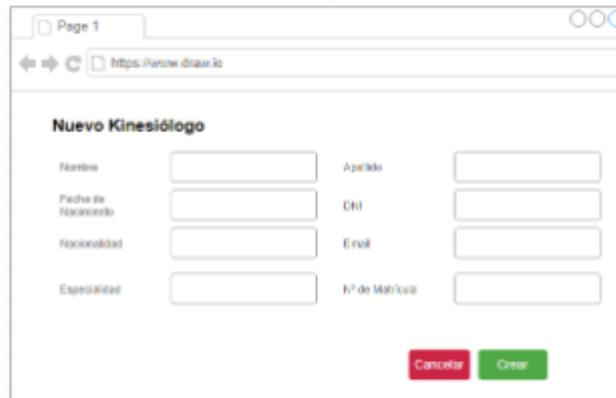
- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de @Administrador.

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de Kinesiólogos	El usuario podrá crear un Kinesiólogo e ingresar los Datos mencionados en la tabla	ALTA	7
Baja de Kinesiólogos	El usuario podrá eliminar un Kinesiólogo	BAJA	2

Modificación de Kinesiólogos	El usuario podrá modificar los Datos de un Kinesiólogo.	MEDIA	4
Mostrar Kinesiólogo	El usuario podrá ver los datos de un Kinesiólogo. Es necesario ver todos los datos del Kinesiólogo y su especialidad.	MEDIA	4
Búsqueda de Kinesiólogos	El usuario podrá buscar Kinesiólogos.	MEDIA	6
Interfaz de Kinesiólogo	Se deberá crear una interfaz amigable e intuitiva	MEDIA	5
Testing del Módulo de Kinesiólogo	Implementar pruebas unitarias y funcionales para el módulo.	ALTA	7

Interacción con el usuario y diseño



8.6.07 ABM de Patologías

Objetivo

- El sistema debe permitir crear Patologías.
- El sistema debe permitir ingresar al sistema los datos de Patología de forma simple.
- El sistema debe permitir la búsqueda intuitiva y sencilla, edición y eliminación de Patologías.
- Las Patologías están asociadas al tratamiento y el usuario debería poder en forma simple, asociar diferentes Tratamientos para las mismas.
- Una Patología puede tener asociado 0 a varios Tratamientos

Datos a completar:

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
-------	------	-------------	---------------

Nombre	String	SI	
Código	String	NO	
Descripción	String	NO	

Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Administrador.

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de Patologías	El usuario podrá crear una Patología e ingresar los Datos mencionados en la tabla	MEDIA	5
Baja de Patologías	El usuario podrá eliminar un Patología.	BAJA	2
Modificación de Patologías	El usuario podrá modificar los Datos de una Patología.	MEDIA	4
Mostrar Patología	El usuario podrá ver los datos de una Patología.	MEDIA	4
Búsqueda de Patología	El usuario podrá buscar Kinesiólogos.	BAJA	3
Interfaz de Patología	Se deberá crear una interfaz amigable e intuitiva	MEDIA	4
Testing del Módulo de Patologías	Implementar pruebas unitarias y funcionales para el módulo.	MEDIA	6

Interacción con el usuario y diseño



The screenshot shows a web browser window with the address bar displaying 'https://www.draw.io'. The main content area contains a form titled 'Nueva Patología'. The form has three input fields: 'Nombre', 'Apellido', and 'Descripción'. At the bottom right of the form, there are two buttons: a red 'Cancelar' button and a green 'Crear' button.

8.6.08 ABM Tipo de Recursos

Objetivo

- El sistema debe permitir agregar, modificar o eliminar Tipos de recursos.
- El sistema debe permitir ingresar de forma simple los datos del Tipo de Recurso.
- El sistema debe permitir buscar Tipos de Recursos fácilmente.

Datos a Completar:

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Nombre	String	SI	
Tipo	String	SI	
Familia	String	SI	
Icono	Icono	SI	
Imagen	Archivo	NO	

Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo, Secretario,

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de Tipo de Recursos	El usuario podrá crear Tipos de Recursos e ingresar los Datos mencionados en la tabla	MEDIA	4
Baja de Tipo de Recursos	El usuario podrá eliminar Tipos de Recursos	BAJA	2
Modificación de Tipo de Recursos	El usuario podrá modificar los Datos de los Tipos de Recursos	MEDIA	4
Búsqueda de Tipo de Recursos	El usuario podrá ver los datos de un Tipo de Recurso	MEDIA	4
Mostrar Tipo de Recurso	El usuario podrá buscar Tipos de Recursos	BAJA	2
Interfaz de Tipo de Recursos	Se deberá crear una interfaz amigable e intuitiva	MEDIA	4

Interacción con el usuario y diseño



8.6.09 Módulo de Recursos

Objetivo

- El sistema debe permitir agregar, modificar o eliminar Recursos.
- El sistema debe permitir buscar recursos fácilmente.
- El sistema debe permitir asociar un tipo de recurso a un recurso dado.
- El usuario debe poder ingresar al sistema los datos de los Recursos en una forma simple y rápida.
- El sistema deberá administrar los Recursos para poder asignarlos a los Tratamientos y sus turnos.

Datos a Completar:

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
N° de serie	Integer	SI	
Cod. Inventario	String	SI	
Consultorio	Consultorio	SI	
Tipo de recurso	Tipo de Recurso	SI	

Supuestos

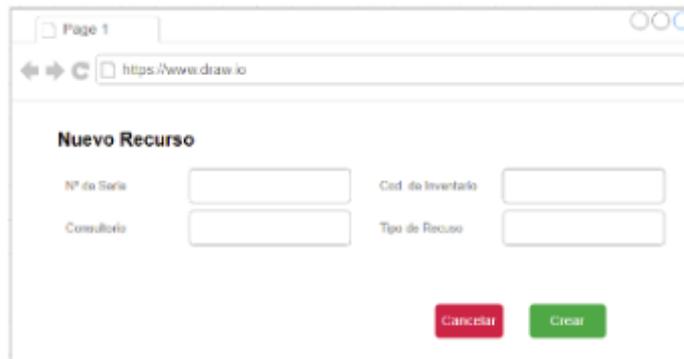
- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo, Secretario.

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
--------	---------------	-----------	-------------

Alta de Recursos	El usuario podrá crear Recursos e ingresar los Datos mencionados en la tabla	ALTA	7
Baja de Recursos	El usuario podrá eliminar Recursos	BAJA	2
Modificación de Recursos	El usuario podrá modificar los Datos de Recursos	MEDIA	4
Búsqueda de Recurso	El usuario podrá buscar un Recurso fácilmente	MEDIA	5
Mostrar un Recurso	El usuario podrá ver la información de un Recurso	BAJA	
Interfaz de Recurso	Se deberá crear una interfaz amigable e intuitiva	MEDIA	4
Testing de Módulo Recursos	Implementar pruebas unitarias y funcionales para el módulo.	MEDIA	6

Interacción con el usuario y diseño



8.6.10 ABM de Obras Sociales

Objetivo

- El sistema debe permitir agregar, modificar o eliminar Obras Sociales.
- El sistema debe permitir buscar Obras Sociales de forma intuitiva.
- El usuario puede también generar y modificar los diferentes planes para los pacientes.
- Los pacientes están asociados a una Obra Social, y cada uno, según la Obra Social, puede tener diferentes planes.
- El usuario debe poder ingresar al sistema los datos de los Obras Sociales en una forma simple y rápida.

Datos a Completar:

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Nombre	String	SI	
Código	String	SI	
Teléfono	String	SI	
Email	String	SI	
Nombre Arancel	Array	NO	
Monto	Integer	NO	
Validez	Date	NO	

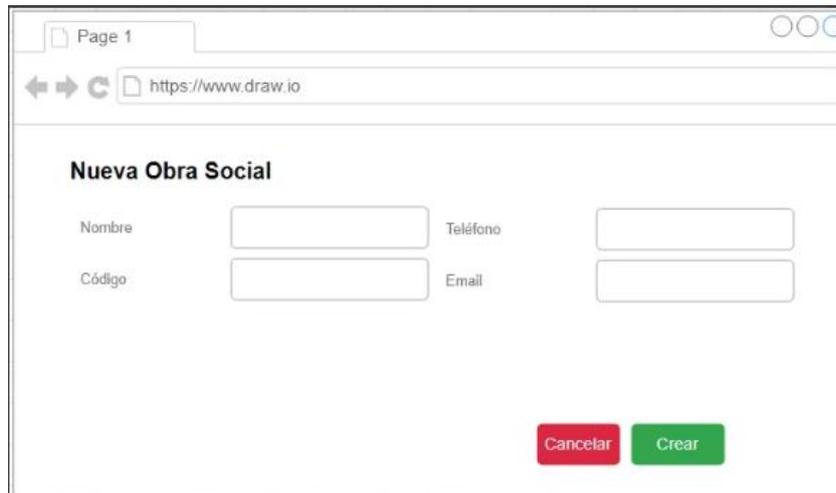
Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo o Secretario.

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de O.S.	El usuario podrá crear Obras Sociales.	ALTA	7
Baja de O.S.	El usuario podrá eliminar Obras Sociales	BAJA	2
Modificación de O.S.	El usuario podrá modificar los Datos de Obras Sociales	MEDIA	4
Mostrar O.S.	El usuario podrá ver los datos de una Obra Social.	MEDIA	4
Búsqueda de O.S.	El usuario podrá buscar Obras Sociales según sus datos.	MEDIA	6
Interfaz de O.S.	Se deberá crear una interfaz amigable e intuitiva	MEDIA	5
Testing del Módulo de O.S.	Implementar pruebas unitarias y funcionales para el módulo.	ALTA	7
Asociar O.S. a Paciente	El usuario podrá asociar una Obra Social a un paciente y detallar el plan correspondiente	MEDIA	6

Interacción con el usuario y diseño



8.6.11 Gestión por calendario funcional

Objetivo

- El sistema debe permitir seleccionar los días y horas mediante la vista de un calendario, para definir turnos
- El sistema debe permitir ver de forma clara e intuitiva, información del día, horarios ocupados y disponibles en la distribución de turnos.
- El sistema debe permitir acceder a los turnos mediante el calendario.

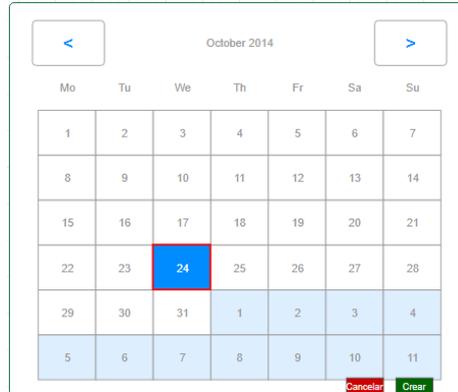
Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo o Secretario.

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Interfaz del Calendario	Se deberá crear una interfaz amigable e intuitiva	MEDIA	4
Mostrar Turnos reservados	El usuario podrá ver los turnos en el calendario, por día, semana, o mes.	MEDIA	4
Selección de fecha de turnos por calendario	El usuario podrá seleccionar la fecha y hora directamente desde el calendario	ALTA	7

Interacción con el Usuario y Diseño



8.6.12 Notificaciones y Mensajes

Objetivo

- El sistema debe permitir crear Notificaciones.
- Las notificaciones deberán tener un estado que indique su situación en un momento determinado y se pueda diferenciar entre ellos.
- El sistema debe permitir modificar y eliminar notificaciones existentes.
- El sistema debe generar automáticamente notificaciones de turnos.
- El sistema debe brindar una interfaz adecuada y un mecanismo para la presentación de eventos de dichas notificaciones.

Datos a completar:

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Estado	String	SI	
Tipo de Notificación	String	SI	
Texto	String	SI	
Usuario	Usuario	SI	

Supuestos

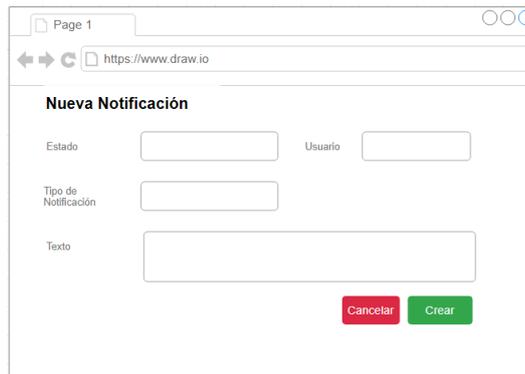
- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo o secretario.

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de Notificación	El sistema deberá permitir crear notificaciones	MEDIA	4

Baja de Notificación	El sistema deberá permitir eliminar notificaciones	BAJA	2
Modificación de Notificación	El sistema deberá permitir la modificación de las Notificaciones	MEDIA	4
Ver Notificación	El sistema deberá mostrar los datos de la notificación seleccionada	MEDIA	3
Diseño de Interfaz	Se deberá crear una interfaz amigable e intuitiva	MEDIA	4
Notificaciones generadas automáticamente	El sistema generará y pondrá automáticamente notificaciones de turnos	ALTA	7
Testing del Módulo de Notificaciones	Implementar pruebas unitarias y funcionales para el módulo.	ALTA	6

Interacción con el usuario y diseño



The screenshot shows a web browser window with the URL <https://www.draw.io>. The page title is "Nueva Notificación". The form contains the following elements:

- Estado:** A text input field.
- Usuario:** A text input field.
- Tipo de Notificación:** A text input field.
- Texto:** A larger text area for input.
- Buttons:** A red "Cancelar" button and a green "Crear" button.

8.6.13 ABM de Tratamientos

Objetivo

- El sistema deberá permitir agregar, modificar o eliminar Tratamientos.
- El sistema deberá permitir buscar Tratamientos de forma intuitiva.
- El sistema deberá permitir asignarle Patologías y Recursos a los Tratamientos.
- El sistema deberá permitir la asignación de Datos de Evolución a los Tratamientos
- El sistema deberá ingresar al sistema los datos de los Tratamientos en una forma simple y rápida.

- El sistema debe permitir establecer una relación de tiempo, en actual el Recurso será utilizada en un Tratamiento

Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Administrador o Kinesiólogo.

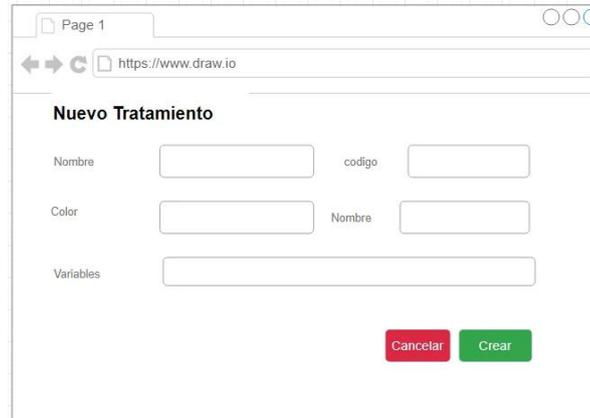
Datos a completar:

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Nombre	String	SI	
Código	String	SI	
Color	Color	SI	
Patología	Patología	SI	
Variables	VariablesDeTratamientos	NO	

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de Tratamientos	El usuario podrá crear un Tratamientos e ingresar los Datos mencionados en la tabla	ALTA	8
Baja de Tratamientos	El usuario podrá eliminar Tratamientos	ALTA	7
Modificación de Tratamientos	El usuario podrá modificar los Datos de Tratamientos	MEDIA	4
Mostrar Tratamiento	El usuario podrá ver los datos de un Tratamiento	MEDIA	5
Buscar Tratamientos	El usuario podrá buscar los tratamientos	MEDIA	6
Interfaz de Tratamientos	Se deberá crear una interfaz amigable e intuitiva	MEDIA	5
Testing del módulo de Tratamientos	Implementar pruebas unitarias y funcionales para el módulo.	ALTA	7

Interacción entre Usuario y Diseño



8.6.14 Gestión de Aranceles

Objetivo

- El sistema debe permitir asociar las Órdenes médicas con los cobros.
- El sistema debe permitir gestionar el estado de las órdenes, enviadas para cobro y cobradas.
- El sistema debe permitir la búsqueda y filtrado de Órdenes Médicas según el estado.
- El sistema debe permitir asociar las Órdenes Médicas con un kinesiólogo, y dicha asociación dar la posibilidad de generar una liquidación de saldos cobrados.

Datos a completar:

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Importe Kinesio	Integer	NO	
Importe Kinesio con Internación	Integer	NO	
Importe Kinesio con Magneto	Integer	NO	
Importe Hidroterapia	Integer	NO	
Importe Neurológico	Integer	NO	
Importe Drenaje	Integer	NO	
Importe Respiratorio	Integer	NO	
Importe Domicilio	Integer	NO	

Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo o Secretario.

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de Aranceles de O.S.	El usuario podrá cargar los aranceles de las prácticas de cada O.S.	MEDIA	5
Modificación de Aranceles de O.S.	El usuario podrá modificar los aranceles de las prácticas de cada O.S.	BAJA	3
Interfaz de Aranceles de O.S.	Se deberá crear una interfaz amigable e intuitiva	MEDIA	5

8.6.15 ABM Variables de seguimiento

Objetivos

- El Usuario podrá agregar, modificar o eliminar Variables de Seguimiento.
- El usuario podrá asociar las Variables de Tratamientos a uno o más Tratamientos.
- El sistema deberá permitir ingresar un valor de esas variables luego de cada turno, para así poder hacer un seguimiento del paciente.

Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo.

Datos a completar

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Nombre	String	SI	
Unidad de Medida	String	SI	

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de Variables de S.	El usuario podrá crear una Variables de S.	MEDIA	4
Baja de Variables de S.	El usuario podrá eliminar Variables de S.	BAJA	2

Modificación de Variables de S.	El usuario podrá modificar Variables de S.	MEDIA	4
Mostrar Variable de S.	El usuario podrá ver los datos de una Variable de S.	BAJA	3
Búsqueda de Variables de S.	El usuario podrá buscar Variables de S.	MEDIA	4
Tratamientos: Asociar Variables de S.	El usuario podrá asociar Variables a uno o varios Tratamientos. Lo que permitirá registrar la evolución del paciente cuando un paciente tenga asociada esa Patología.	MEDIA	4
Turnos: carga información de Variables de S.	Registrar valores de esas variables en los turnos de sus Tratamientos asociados	ALTA	7
Interfaz de Variables de S.	Se deberá crear una interfaz amigable e intuitiva	MEDIA	5
Testing del Módulo de Variables de Seguimiento	Implementar pruebas unitarias y funcionales para el módulo	ALTA	7

Interacción con el Usuario y Diseño



8.6.16 ABM Rehabilitación

Objetivo

- El sistema debe permitir Crear, modificar y eliminar Rehabilitaciones
- El sistema debe permitir listar los datos de el o los Tratamientos, Órdenes Médicas y Obras Sociales que el paciente tenga registrados de manera simple y rápida.
- El sistema debe permitir la asociación de una Rehabilitación y una o más órdenes sociales.

- El sistema debe mostrar un listado de los turnos pertenecientes a dicha rehabilitación
- El sistema debe mostrar una interfaz simple e intuitiva.

Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de kinesiólogo o secretario.

Datos a completar

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Paciente	Paciente	SI	
Plan de O.S.		NO	
Patología	Patología	SI	
Plantilla de Tratamiento	Tratamiento	SI	
Kinesiólogo	Kinesiólogo	SI	
Total de Sesiones	Integer	NO	
Sesiones ya Realizadas	Integer	NO	

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de Rehabilitaciones	El usuario podrá crear Rehabilitaciones	ALTA	7
Baja de Rehabilitaciones	El usuario podrá eliminar Rehabilitaciones	MEDIA	4
Modificación de Rehabilitaciones	El usuario podrá realizar modificaciones en las Rehabilitaciones	MEDIA	5
Mostrar Rehabilitación	El usuario podrá ver los datos de las Rehabilitaciones	MEDIA	5
Buscar Rehabilitaciones	El usuario podrá buscar las Rehabilitaciones según los datos presentados	MEDIA	4
Diseño de Interfaz	Se deberá crear una interfaz amigable e intuitiva	MEDIA	5
Testing del Módulo de Historia Clínica	Implementar pruebas unitarias y funcionales para el módulo.	ALTA	7

8.6.17 ABM de Orden Médica

Objetivo

- El sistema debe permitir agregar, modificar o eliminar Órdenes Médicas.
- El sistema debe permitir buscar una Orden Médica de forma intuitiva.
- Los pacientes están asociados a una Obra Social, y cada uno, según la Obra Social, puede tener diferentes planes, los cuales poseen un arancel determinado.
- El sistema debe permitir gestionar el estado de las órdenes, enviadas para cobro y cobradas.
- El sistema debe permitir generar un informe según los turnos y la liquidación de cobro de Órdenes Médicas.

Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo o Secretario.

Datos a completar

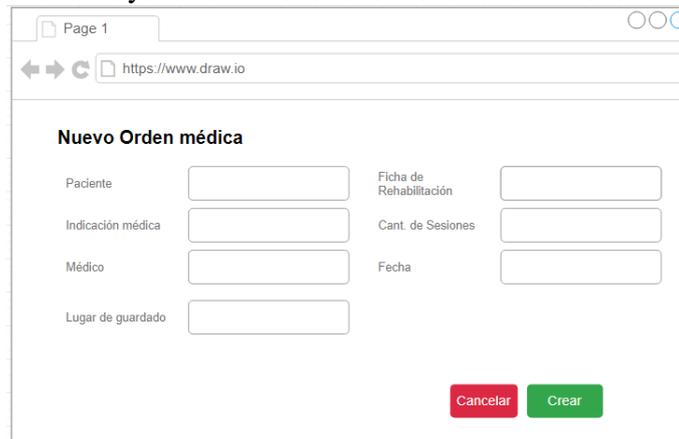
CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Paciente	Paciente	NO	
Ficha de Rehabilitación	Rehabilitación	NO	
Indicación médica	String	NO	
Cant. Sesiones	Integer	NO	
Médico	String	NO	
Fecha Orden médica	Date	NO	
Lugar de Guardado	Lugar de Guardado	NO	

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de O.M.	El usuario podrá crear un O. M.	ALTA	7
Baja de O.M.	El usuario podrá eliminar O.M.	BAJA	2
Modificación de O.M.	El usuario podrá modificar los Datos de O. M.	MEDIA	4
Mostrar O.M.	El usuario podrá ver los datos de una O. M.	MEDIA	4

Búsqueda de O.M.	El usuario podrá buscar las O.M.	MEDIA	6
Interfaz de O.M.	Se deberá crear una interfaz amigable e intuitiva	MEDIA	5
Testing del Módulo de O. M.	Implementar pruebas unitarias y funcionales para el módulo.	ALTA	7

Interacción con Usuario y Diseño



8.6.18 ABM Especialidades Médicas

Objetivo

- El sistema debe permitir agregar, modificar o eliminar Especialidades Médicas.
- El sistema debe permitir buscar una Especialidad Médica de forma intuitiva.
- Los kinesiólogos estarán asociados a una Especialidad Médica.

Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo o Secretario.

Datos a completar

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Nombre	String	SI	
Código	String	SI	
Alias	String	SI	

Tareas:

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de Especialidades M.	El usuario podrá crear una Especialidad Médica e ingresar los Datos mencionados en la tabla	MEDIA	5
Modificación de Especialidades M.	El usuario podrá eliminar una Especialidades Médica	MEDIA	4
Baja de Especialidades M.	El usuario podrá modificar los Datos de una Especialidad Médica	BAJA	2
Búsqueda de Especialidades M.	El usuario podrá ver los datos de una Especialidad Médica	BAJA	2
Mostrar Especialidad M.	El usuario podrá buscar una Especialidad Médica	BAJA	2
Interfaz de Especialidad M.	Se deberá crear una interfaz amigable e intuitiva	MEDIA	4

8.6.19 Informes de Turnos

Objetivo

- El sistema debe permitir generar un informe según los turnos.
- El sistema debe permitir la exportación del informe.

Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo o Secretario.

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Generar informe según los Turnos de un Kinesiólogo	El usuario podrá generar un informe diario con los turnos asignados para el día actual	ALTA	7
Exportar Informes	El usuario podrá exportar el informe de turnos en formato pdf	MEDIA	4
Testing del Módulo de Informes	Implementar funcionales para el test de este requerimiento	MEDIA	5

8.6.20 Módulo de Sesiones de Tratamientos

Objetivo

- El sistema deberá permitir agregar, modificar o eliminar Sesiones de Tratamientos.
- El sistema deberá permitir asignarle una sesión a un Tratamiento.
- El sistema deberá ingresar al sistema los datos de las sesiones de Tratamientos en una forma simple y rápida.
- El sistema debe permitir establecer una relación de % de tiempo, en el que el Recurso será utilizada en una Sesión de Tratamiento.

Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo

Datos a completar

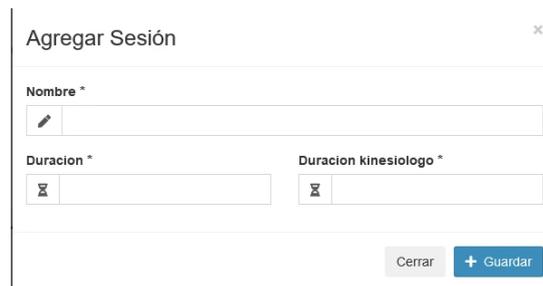
CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Nombre	String	SI	
Duración	Integer	SI	
Tiempo de Kinesiólogo	Integer	SI	

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de Sesión de Tratamientos	El usuario podrá crear una Sesión de Tratamientos e ingresar los Datos mencionados en la tabla	ALTA	8
Copia de Sesión de Tratamiento	El usuario copiar una Sesión de Tratamientos para duplicar una existente, dentro del mismo Tratamiento	MEDIA	4
Baja de Sesión de Tratamientos	El usuario podrá eliminar una Sesión de Tratamientos.	BAJA	3
Modificación de Sesión de Tratamientos	El usuario podrá modificar los Datos de una Sesión de Tratamientos.	MEDIA	6
Asociación de par Recurso-Tiempo	El usuario podrá asociar Sesiones de Tratamientos a uno o varios Recursos junto con un tiempo estimado de uso	MEDIA	5

	en el Tratamiento correspondiente		
Asociar Tratamiento	El usuario podrá asociar a una Sesión de Tratamientos a un Tratamiento	ALTA	8

Interacción con el Usuario y Diseño



8.6.21 Informe: Variable de Seguimiento

Objetivo

- El sistema debe permitir generar un informe según los Tratamientos y las V. de Seguimiento asignadas.
- El sistema debe permitir seleccionar variables para combinar en un informe de V. de Tratamiento.
- El sistema debe permitir la exportación del informe.
- El sistema debe permitir generar gráficas de análisis de evolución de las Variables de Seguimiento de un tratamiento Seleccionado.

Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo o Secretario

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Generar Informe según Variables de S.	El usuario podrá generar Informes que permita hacer un análisis completo de las Variables asociadas a los Tratamientos.	ALTA	7
Exportar Informes	El usuario podrá exportar el informe en	MEDIA	6

	formato pdf.		
Testing Módulo informes V. de S.	Implementar pruebas funcionales para el módulo	MEDIA	6

8.6.22 ABM Lugar de Guardado

Objetivo:

- El sistema debe permitir la creación, modificación y baja de Lugares de Guardado.
- El sistema debe permitir buscar un Lugar de Guardado
- Las Ordenes médicas pueden estar asociados a un Lugar de Guardado.

Supuestos

- El Usuario debe estar logeado en el Sistema.
- El Usuario debe tener permisos de Kinesiólogo o Secretario

Datos a completar

CAMPO	TIPO	OBLIGATORIO	RESTRICCIONES
Lugar	String	SI	

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Alta de L. de G.	El usuario podrá crear Lugares de G.	MEDIA	5
Baja de L. de G..	El usuario podrá eliminar Lugares de G.	MEDIA	4
Modificación de L. de G.	El usuario podrá modificar los Datos de los Lugares de G.	MEDIA	4
Mostrar L. de G.	El usuario podrá ver los datos de un Lugar de G.	MEDIA	4
Búsqueda de L de G.	El usuario podrá buscar Lugares de G.	MEDIA	5
Interfaz de L. de G.	Se deberá crear una interfaz amigable e intuitiva	MEDIA	5
Asociar O. M.	El usuario podrá asociar una Orden Médica a un Lugar de Guardado determinado	BAJA	2

8.6.23 Diseño de Dashboard

Objetivo:

- El sistema debe presentar la información resumida, específica del día, para que el usuario tenga un panorama general diario.
- El sistema debe permitir acceder a las funcionalidades más importantes para el kinesiólogo.
- El sistema debe permitir generar rápidamente turnos, gestión de pacientes, visualización del calendario, etc.

Tareas

TAREAS	REQUERIMIENTO	PRIORIDAD	TASK POINTS
Generar acceso a funcionalidades cotidianas	El usuario podrá acceder a las funcionalidades del sistema fácilmente, con énfasis en las actividades cotidianas.	ALTA	8
Edición de distribución de accesos a funcionalidades	El usuario tendrá agrupadas las funcionalidades por menú.	MEDIA	7
Interfaz del tablero		MEDIA	5

8.7 Herramientas

En este apartado presentaremos una breve descripción de las herramientas utilizadas durante el desarrollo del proyecto, comentando brevemente sus principales características y su versión utilizada.

Las Herramientas Elegidas para realizar el Proyecto fueron seleccionadas buscando la mayor integración posible entre las mismas en todo el desarrollo del Software. Es decir, que la integración entre ellas fue un factor determinante, así también que sean Open-Source o bien de libre utilización hasta la cantidad de usuarios integrantes del grupo del desarrollo.

Jira Software

Consideramos que Jira brindaría un espacio de trabajo colaborativo, ordenando las tareas e integrando las demás herramientas. Ya que está pensada principalmente para equipos ágiles y cuenta con una variedad más que interesante de herramientas compatibles que se pueden sumar en cualquier momento del desarrollo.

Según sus desarrolladores Jira es "La herramienta de desarrollo de software líder de los equipos ágiles". Pensada para que todos los miembros de tu equipo de software puedan planificar, seguir y publicar software.

Elegimos esta herramienta ya que es muy potente en los siguientes aspectos:

- Planificación - Manejo de Historias de usuario e incidencias, planificación de sprints y distribución de tareas.
- Supervisión - Permite una vista completa de todo el contexto de trabajo.
- Informes - Permite ver datos en tiempo real en forma gráfica
- Manejar flujo de Trabajo - Se puede adaptar el workflow a cada grupo de trabajo.
- Permite integración con Bitbucket - Permitirá mantener el código actualizado
- Gestión de Conocimiento - Consta de una variedad de templates importantes para documentar el trabajo.

Es de uso gratuito hasta 10 Usuarios y cuenta con 2 GB de almacenamiento de archivos.

Para conocer sobre su licencia: <https://www.atlassian.com/es/software/jira/pricing>



Bitbucket

Según sus desarrolladores "Bitbucket es más que solo gestión de código Git. Bitbucket ofrece a los equipos un sitio para planificar proyectos, colaborar en el código, probar y desplegar."

Esta herramienta nos permitirá gestionar el código con un gestor GIT, facilitando el versionado de código y el control de cambios. Entre sus aspectos más importantes, los factores decisivos en la elección del mismo fueron:

- Repositorio Privados Gratuitos e ilimitados
- Integración con Jira
- Entrega Continúa integrada - Permite compilar, probar y realizar el despliegue.
- Revisión de código - Permite crear lista de comprobaciones

Está integrado con Jira y comparte la política de almacenamiento y límite usuarios.

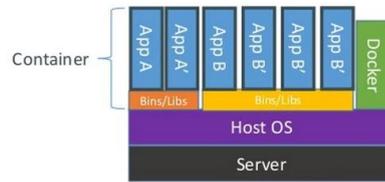
Para conocer sobre su licencia: <https://www.atlassian.com/legal/cloud-terms-of-service>



Docker

Docker es una herramienta open-source que nos permite realizar una ‘virtualización ligera’, con la que poder empaquetar entornos y aplicaciones que posteriormente podremos desplegar en cualquier sistema que disponga de esta tecnología.

Permite crear múltiples sistemas totalmente aislados entre sí, sobre la misma máquina o sistema anfitrión.



Docker aprovecha el sistema operativo sobre el cual se ejecuta, comparte el kernel del sistema operativo anfitrión e incluso parte de sus bibliotecas.

Sus Principales ventajas son:

- Poder tener un entorno de desarrollo unificado, de esta manera los miembros del equipo de desarrollo tuvimos todo el tiempo las mismas versiones de MySQL, PHP, nginx, previniendo cualquier incompatibilidad o problema.
- Las instancias se arrancan en pocos segundos.
- Es fácil de automatizar e implantar en entornos de integración continua.
- Existen multitud de imágenes que pueden descargarse y modificarse libremente.

Es de utilización gratuita para nuestro caso, para conocer más sobre el licenciamiento:

<https://www.docker.com/legal/docker-terms-service>



MySql

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base datos de código abierto más popular del mundo.

Para conocer sobre su licenciamiento: <https://www.oracle.com/legal/terms.html>



NGINX

Es un servidor web/proxy inverso ligero de alto rendimiento y un proxy para protocolos de correo electrónico (IMAP/POP3).

Es software libre y de código abierto, licenciado bajo la Licencia BSD simplificada; también existe una versión comercial distribuida bajo el nombre de Nginx Plus.

Es multiplataforma, por lo que corre en sistemas tipo Unix (GNU/Linux, BSD, Solaris, Mac OS X, etc.)



Symfony

Symfony es un completo framework diseñado para optimizar el desarrollo de las aplicaciones web basado en el patrón Modelo Vista Controlador. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web.

Uno de los puntos fuertes de Symfony es su escalabilidad. Permite segmentar el código en bundles para que el proyecto pueda seguir creciendo. Con un Bundle encapsulamos una o varias funcionalidades.

Utilizaremos la versión Symfony 3.4 LTS.



PHPUnit

PHPUnit se creó con idea de que cuanto antes se detecten los errores en el código antes podrán ser corregidos. Este conocido framework para PHP nos permite crear y ejecutar juegos de tests unitarios de manera sencilla Como todos los frameworks de pruebas unitarias.

El objetivo de las pruebas unitarias es aislar cada parte del programa y demostrar que las partes de forma individual son correctas. Una prueba unitaria proporciona un contrato escrito que la pieza de código debe satisfacer. Como resultado, las pruebas unitarias encuentran problemas en las fases iniciales del desarrollo de software.



BOUML

BOUML es una herramienta para modelar UML 2 y generar código C++, Java, Idl, Php, Python and MySQL. Utilizamos este software para hacer la ingeniería inversa de nuestro código y así generar los diagramas. Teniendo en cuenta el principio de minimizar las tareas de documentación. (Para más información, ingresar a <https://www.bouml.fr/>)



LucidChart

Lucidchart brinda una solución intuitiva y colaborativa para la creación de diagramas. Enfocado en calidad, seguridad y asequibilidad facilita la decisión.

Lucidchart es una herramienta de diagramación basada en la web, que permite a los usuarios colaborar y trabajar juntos en tiempo real, creando diagramas de flujo, organigramas, esquemas de sitios web, diseños UML, mapas mentales, prototipos de software y muchos otros tipos de diagrama. Construida con estándares web, como HTML5 y JavaScript.



8.8 Lenguajes Utilizados para el desarrollo

PHP

PHP es un lenguaje de programación de propósito general de código del lado del servidor. Aplicado al desarrollo web de contenido dinámico.

PHP puede ser desplegado en la mayoría de los servidores web y en todos los sistemas operativos y plataformas sin costo alguno. El lenguaje PHP se encuentra instalado en más de 20 millones de sitios web y en un millón de servidores.



JavaScript

JavaScript (abreviado comúnmente JS) es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo, en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.



HTML

HTML, sigla en inglés de HyperTextMarkup Lenguaje (lenguaje de marcas de hipertexto), hace referencia al lenguaje de marcado para la elaboración de páginas web. Es un estándar que sirve de referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes,

videos, juegos, entre otros. Es un estándar a cargo del World Wide Web Consortium (W3C) o Consorcio WWW, organización dedicada a la estandarización de casi todas las tecnologías ligadas a la web, sobre todo en lo referente a su escritura e interpretación. Se considera el lenguaje web más importante siendo su invención crucial en la aparición, desarrollo y expansión de la World Wide Web (WWW). Es el estándar que se ha impuesto en la visualización de páginas web y es el que todos los navegadores actuales han adoptado.

8.9 Bundles en Symfony

Con el fin de estandarizar el trabajo del equipo, se establecieron un conjunto de buenas prácticas para construir interfaces de usuarios y programación. Mayormente utilizando las ventajas brindadas por un framework tan potente como Symfony, en cual se pueden instalar fácilmente ya que uno de los puntos fuertes de Symfony es su escalabilidad. Permite segmentar el código en bundles para que el proyecto pueda seguir creciendo. Con un Bundle encapsulamos una o varias funcionalidades.

No explicaremos los Bundles básicos de Symfony, ya que son los que da soporte a la mayoría de los desarrollos: FrameworkBundle, AppBundle, StofDoctrineExtensions Bundle, SecurityBundle, MonologBundle.

Los Bundles utilizados fueron los siguientes:

TwigBundle

Twig es un motor de plantilla para el lenguaje de programación PHP. Es un producto de código abierto autorizado bajo Licencia BSD y mantenido por FabienPotencier. Nace con el objetivo de facilitar a los desarrolladores de aplicaciones web que utilizan la arquitectura MVC el trabajo con la parte de las vistas, gracias a que se trata de un sistema que resulta muy sencillo de aprender y capaz de generar plantillas con un código preciso y fácil de leer.

DoctrineBundle

El Proyecto Doctrine es un conjunto seleccionado de bibliotecas PHP enfocadas principalmente en proporcionar servicios de persistencia y funcionalidades relacionadas. Sus proyectos premiados son un Objectrelationalmapper (ORM) y la Capa de Abstracción de la Base de Datos sobre la cual está construido. Una de sus características clave es la

opción de escribir consultas de bases de datos en un dialecto SQL orientado a objetos patentado llamado Doctrine QueryLanguage (DQL), inspirado en Hibernate HQL. Esto proporciona a los desarrolladores una alternativa poderosa a SQL que mantiene la flexibilidad sin requerir duplicación de código innecesaria.

Doctrine te permite trabajar con bases de datos de una manera mucho más interesante que solo recuperar filas de una tabla basada en columnas de un arreglo. En cambio, Doctrine te permite persistir objetos completos a la base de datos y recuperar objetos completos desde la base de datos. Esto funciona asociando una clase PHP a una tabla de la base de datos, y las propiedades de esa clase PHP a las columnas de la tabla:

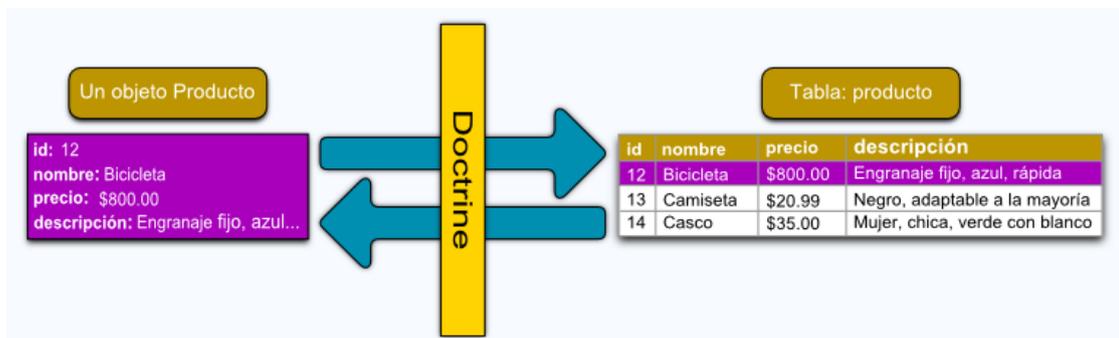


Fig. 2.2-01: Muestra la relación entre un objeto y una tabla en B.D.

SensioFrameworkExtraBundle

FrameworkBundle es el predeterminado que implementa un marco MVC básico, pero robusto y flexible. SensioFrameworkExtraBundle lo extiende para agregar convenciones y anotaciones de forma elegante. Permite lograr controladores más concisos.

FOSUserBundle

FOSUserBundle agrega soporte para un sistema de usuario respaldado por base de datos en Symfony2 o superior. Proporciona un marco flexible para la gestión de usuarios que tiene como objetivo manejar tareas comunes como el registro de usuarios y la recuperación de contraseñas.

Las características incluyen:

- Los usuarios pueden almacenarse a través de Doctrine ORM o MongoDB / CouchDB ODM
- Soporte de registro, con una confirmación opcional por correo electrónico.

- Soporte de restablecimiento de contraseña
- Integrado a las pruebas de unidad

JMSerializerBundle

Esta biblioteca le permite (des) serializar datos de cualquier complejidad. Actualmente, es compatible con XML y JSON. También le proporciona un rico conjunto de herramientas para adaptar la salida a sus necesidades específicas. Las características incorporadas incluyen:

- (Des-) serializar datos de cualquier complejidad; referencias circulares y estrategias de exclusión complejas se manejan con gracia.
- Admite muchos tipos de PHP integrados (como fechas, intervalos)
- Se integra con Doctrine ORM.
- Configurable a través de XML, YAML o anotaciones

AdminLTEBundle

Este es un paquete de Symfony que brinda una interfaz amigable y de alto diseño a las aplicaciones. Está integrada en un alto grado con los bundles de Symfony, sus principales puntos fuertes son:

- Soporte para Symfony 4.x
- Soporte para KNPMenuBundle
- Soporte para FOSUserBundle
- Soporte de Webpack-Encore para construir activos
- Manejo dirigido por eventos de entradas de menú, tareas y notificaciones
- Traducciones para: inglés, alemán, italiano, checo, español, ruso, árabe, finlandés, japonés, sueco, portugués (brasileño), holandés, francés, turco, danés, chino, eslovaco, vasco, polaco (ayuda a traducirlo a más idiomas)
- Basado en AdminLTE 2.4.8
- Usando FontAwesome 5

KnpgaufretteBundle

Gaufrette es una biblioteca PHP que proporciona una capa de abstracción del sistema de archivos. Esta capa de abstracción permite desarrollar aplicaciones sin necesidad de saber dónde se almacenarán todos sus archivos multimedia o cómo.

TetranzSelect2EntityBundle

Este es un paquete de Symfony que permite utilizar el popular componente Select2 como reemplazo directo para un campo de entidad estándar en un formulario de Symfony.

DigitalAsceticBaseEntityBundle

DigitalAsceticQueryFilterBundle: QueryFilterBundle trae funcionalidades de filtrado de solicitudes y paginación a Symfony.

MobileDetectBundle

Permite detectar dispositivos móviles, administrar tipos de vista móvil, redirigir a la versión móvil.

CalendarBundle

Proporciona un calendario de eventos para Symfony. Compatible con Doctrine ORM & ODM y API como Google Calendar. (Ver Anexo 8.2 - Análisis para la elección del Calendario).

8.10 Aplicación de las herramientas

En este apartado describiremos brevemente la forma en que se han utilizado las herramientas para desarrollar el proyecto, describiendo brevemente la configuración utilizada.

Configuración de las estaciones de trabajo:

Docker-Desktop: Como se explicó en la elección de las herramientas, Docker ha sido un gran aliado en el desarrollo del proyecto, ya que permitió una abstracción de la instalación de las plataformas en el ambiente de desarrollo, logrando con dos comandos estar listos para desarrollar.

Se Generó el siguiente archivo de configuración para los contenedores:

```
version: "3.1"
services:

  mysql:
    image: mysql:8.0
    container_name: healmate-mysql
    working_dir: /application
    command: --default-authentication-plugin=mysql_native_password
    volumes:
      - ./application
    environment:
      - MYSQL_DATABASE=db
    ports:
      - "8082:3306"

  webserver:
    image: nginx:alpine
    container_name: healmate-webserver
    working_dir: /application
    volumes:
      - ./application
      - ./phpdocker/nginx/nginx.conf:/etc/nginx/conf.d/default.conf
    ports:
      - "8080:80"

  php-fpm:
    build: phpdocker/php-fpm
    container_name: healmate-php-fpm
    working_dir: /application
    volumes:
      - ./application
      - ./phpdocker/php-fpm/php-ini-overrides.ini:/etc/php/7.2/fpm/conf.d/99-overrides.ini

  phpmyadmin:
    depends_on:
      - mysql
    image: phpmyadmin/phpmyadmin
    container_name: healmate-phpmyadmin
    links:
```

```
- mysql:mysql
ports:
  - 8083:80
environment:
PMA_HOST: mysql
```

Recuadro 8.10-01: Archivo de configuración de Docker

También se generaron los archivos dentro de phpdocker, las cuales son cuestiones muy técnicas y pueden verse en el código fuente en caso de que sea requerido.

Para levantar el entorno de desarrollo se llama a los contenedores de la siguiente forma:

```
Windows PowerShell
\healmate> docker-compose up -d
Starting healmate-php-fpm ... done
Starting healmate-mysql ... done
Starting healmate-webserver ... done
Starting healmate-phpmyadmin ... done
```

Fig 8.10-01: Poner en marcha el proyecto en desarrollo

Para detenerlos:

```
\healmate> docker-compose stop
Stopping healmate-phpmyadmin ... done
Stopping healmate-php-fpm ... done
Stopping healmate-webserver ... done
Stopping healmate-mysql ... done
```

Fig 8.10-02: Poner detener el proyecto en desarrollo

Con esto, en la terminal de desarrollo, tenemos todo lo necesario para trabajar localmente e independiente del servidor de producción, es decir, un servidor symfony con base de datos y Nginx.



Fig 8.10-03: Ilustración de la integración de las herramientas

De esta forma se conformó la arquitectura principal del sistema. La misma fue replicada en el servidor de producción (Sin Docker) con Raspbian.

8.11 Testing – Casos de Prueba

En el presente anexo detallaremos los casos de pruebas generados en el transcurso de los *sprints*.

Vale la pena repetir que fueron documentadas las pruebas iniciales de los tipos unitarios y funcionales de los módulos, dados la repetitividad de procesos, y poniendo mucho más énfasis en las Pruebas de Aceptación, siendo estas las que mejor se acoplaron a la hora de trabajar en retroalimentación con el cliente en cada entrega.

8.11.1 Casos de Prueba Automatizados.

Se presentan los casos de los *sprints* 2, 3 y 4, donde se ejemplifica los diferentes tipos de pruebas diferentes, evitando la repetición de casos de prueba similares.

Sprint 2

ABM de Usuarios

Caso de Prueba CP2-03

Prioridad: Medio
Fecha de Pueba: 28/9/2019

Prueba de Creación de nuevo usuario

Descripción:

Aceptación

Verifique que puede dar de alta un nuevo usuario en el sistema

Condiciones Previas:

Debe haber un usuario Administrador creado

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	Ingresar al sistema con un usuario que tenga el rol de Administrador. Datos: <code>_server_/login</code> Nota:	<input checked="" type="checkbox"/>	Logeado correcto	Logeado correcto
2	En el menú lateral, seleccionar "Usuarios" y luego la opción "Nuevo" Datos: <code>_server_/dashboard</code> Nota:	<input checked="" type="checkbox"/>	Redirección a la página de creación de Usuarios	Redirección exitosa
3	Ingresar a la Web de creación de Usuarios Datos: <code>_server_/users/new</code> Nota: A partir de aquí para automatizar automatizado	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
4	Cargar datos del usuario nuevo Datos: Nota: Se prueban además, numerosos casos de datos "no validos" automatizados.	<input checked="" type="checkbox"/>	Cargar los datos	Carga correcta
5	Presionar el botón Guardar Datos: Nota: Se muestra un mensaje de error en el primer dato inválido cargado, secuencialmente en el orden descrito en "Datos de Prueba" del paso 4	<input checked="" type="checkbox"/>	Redirección: <code>_server_/users/view/id_creado</code>	Redirección: <code>_server_/users/view/id_creado</code>
6	Verificar la creación del Usuario Datos: Nota:	<input checked="" type="checkbox"/>	Los datos son verificados correctamente	Datos verificados correctamente

Comentario:

Automatizado en CP02_03Test.php

Diseñada por: Matías Milesi

Programada por: Nelson Inalbon



Prioridad: Medio

Fecha de Pueba: 28/9/2019

Caso de Prueba CP2-04

Prueba de edición de un usuario

Descripción:

Aceptación

Verifique que puede editar un usuario en el sistema

Condiciones Previas:

Debe haber un usuario Administrador creado

Paso N°	Descripción		Resultado Esperado	Resultado Actual
1	Ingresar al sistema con un usuario que tenga el rol de Administrador. Datos: <code>_server_login</code> Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	Logueo correcto
2	Cambiar datos del usuario seleccionado Datos: Generar datos de prueba para tener 100 % de Cobertura de código Nota: Se muestra un mensaje de error en el primer dato inválido cargado.	<input checked="" type="checkbox"/>		
3	Presionar el botón Guardar Datos: Nota:	<input checked="" type="checkbox"/>	Redirección: <code>_server_users/view/id_editado</code>	Redirección: <code>_server_users/view/id_editado</code>
4	Verificar la edición del Usuario Datos: Nota:	<input checked="" type="checkbox"/>	Los datos son verificados correctamente	Datos verificados correctamente

Comentario:

Automatizado en CP02_04Test.php

Diseñada por: Nelson Inalbon

Programada por: Matías Milesi



Caso de Prueba CP2-05

Prueba de asignación de un Rol a un usuario

Descripción:

Aceptación

Probar que se pueden asignar Roles a un Usuario

Condiciones Previas:

Debe haber un usuario Administrador creado

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	Ingresar al sistema con un usuario que tenga el rol de Administrador. Datos: <code>_server_/login</code> Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	Logueo correcto
2	Ingresar a la Web de Edición de Usuarios Datos: <code>_server_/users/edit/id_editado</code> Nota: A partir de aquí para automatizar	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
3	Asignar un Rol al usuario seleccionado Datos: Generar datos de prueba para tener 100 % de Cobertura de código Nota:	<input checked="" type="checkbox"/>		
4	Presionar el botón Guardar Datos: Nota:	<input checked="" type="checkbox"/>	Redirección: <code>_server_/users/view/id_editado</code>	Redirección: <code>_server_/users/view/id_editado</code>
5	Verificar la edición del Usuario Datos: Nota:	<input checked="" type="checkbox"/>	Los datos son verificados correctamente	Datos verificados correctamente

Comentario:

Automatizado en `CP02_05Test.php`

Diseñada por: Nelson Inalbon

Programada por: Víctor Rosset



Prioridad: Alto

Fecha de Pueba: 1/10/2019

Caso de Prueba CP2-06

Prueba de edición de Roles de usuarios

Descripción:

Aceptación

Probar que se pueden editar los Roles a un Usuario

Condiciones Previas:

Debe haber un usuario Administrador creado

Paso N°	Descripción	Estado	Resultado Esperado	Resultado Actual
1	Ingresar al sistema con un usuario que tenga el rol de Administrador. Datos: <code>_server_login</code> Nota:	✓	La página debería cargar correctamente	Logueo correcto
2	En el menú lateral, seleccionar "Usuarios" y luego la opción "Todos" Datos: <code>_server /dashboard</code> Nota:	✓	La página debería cargar correctamente	La página carga correctamente
3	En la página usuarios seleccionar un Usuario y presionar el Botón "Editar" Datos: <code>_server_users</code> Nota:	✓	La página debería cargar correctamente	La página carga correctamente
4	Ingresar a la Web de Edición de Usuarios Datos: <code>_server_users/edit/id_editado</code> Nota: A partir de aquí para automatizar	✓	La página debería cargar correctamente	La página carga correctamente
5	Editar un Rol al usuario seleccionado, eligiendo uno nuevo y deseleccionando el actual. Datos: Generar datos de prueba para tener 100 % de Cobertura de código Nota:	✓		
6	Presionar el botón Guardar Datos: Nota:	✓	Redirección: <code>_server_users/view/id_editado</code>	Redirección: <code>_server_users/view/id_editado</code>
7	Verificar la edición del Usuario Datos: Nota:	✓	Los datos son verificados correctamente	Datos verificados correctamente

Comentario:

Automatizado en CP02_06Test.php

Diseñada por: Nelson Inalbon

Programada por: Víctor Rosset



ABM de Consultorios

Caso de Prueba CP2-07

Prioridad: Alto
Fecha de Prueba: 4/10/2019

Prueba de Creación de Consultorio

Descripción:

Aceptación

Verificar la carga correcta de un Consultorio.

Condiciones Previas:

Debe estar logeado al sistema con roles de administrador o kinesiólogo.

Paso N°	Acceder a la interfaz de creación de Consultorio		Resultado Esperado	Resultado Actual
1	Datos: <code>_server_/consultorios/new</code> Nota:	<input checked="" type="checkbox"/>	Acceso a la interfaz de Creación de Consultorio	La página se carga correctamente
2	Datos: Ingresar: nombre, código, teléfono, email, calle, número, localidad, código postal, provincia, país Nota: Se prueban además, numerosos casos de datos "no validos" automatizados.	<input checked="" type="checkbox"/>	Ingreso de datos exitoso	Campos correctamente completados
3	Datos: Nota: Se muestra un mensaje de error en el primer dato inválido cargado, secuencialmente en el orden descrito en "Datos de Prueba" del paso 2	<input checked="" type="checkbox"/>	Redirección: <code>_server_/consultorio/id_consultorio_nuevo</code>	Acceso a la página con los datos del Consultorio generado

Comentario:

Automatizado en CP02_07Test.php

Diseñada por: Matías Milesi

Programada por Nelson Inalbon



Caso de Prueba CP2-08

Prioridad: Medio
Fecha de Pueba: 4/10/2019

Prueba de Edición de un Consultorio

Descripción:

Aceptación

Verificar la correcta edición de los datos de un Consultorio.

Condiciones Previas:

Debe estar logeado al sistema con permisos de administrador o kinesiólogo. Debe existir en base de datos el Consultorio a editar.

Paso N°	Descripción		Resultado Esperado	Resultado Actual
1	Acceder a la interfaz de edición de Consultorio Datos: <code>_server_/consultorios/id_consultorio</code> Nota:	<input checked="" type="checkbox"/>	Acceso a la interfaz de Edición de Consultorio	La página se cargó correctamente.
2	Editar datos del Consultorio Datos: Edición de los diferentes datos del Consultorio Nota: Se prueban además, numerosos casos de datos "no validos" automatizados	<input checked="" type="checkbox"/>	Edición de datos Exitoso	Campos correctamente editados
3	Presionar el botón Guardar Datos: Nota: Se muestra un mensaje de error en el primer dato inválido cargado, secuencialmente en el orden descrito en "Datos de Prueba" del paso 2 del CP02_08	<input checked="" type="checkbox"/>	Redirección: <code>_server_/consultorio/id_consultorio</code>	Acceso a la página con los datos del Consultorio editado

Comentario:

Automatizado en CP02_08Test.php

Diseñada por: Matías Milesi

Programada por: Nelson Inalbon



Caso de Prueba CP2-09

Prioridad: Medio
Fecha de Pueba: 4/10/2019

Prueba de Eliminación de un Consultorio

Descripción:

Aceptación

Verifica la eliminación de un Consultorio existente en base de datos

Condiciones Previas:

Debe estar logeado al sistema con permisos de administrador o kinesiólogo.

Paso N°	Acceder a la Interfaz de edición de Consultorio		Resultado Esperado	Resultado Actual
1	Datos: <code>_server_/consultorios/id_consultorio</code> Nota:	<input checked="" type="checkbox"/>	Acceso a la interfaz de edición	La página se carga correctamente
2	Datos: Nota:	<input checked="" type="checkbox"/>	Visualizar el mensaje de verificación de la eliminación del Consultorio	El mensaje se muestra correctamente
3	Datos: Nota:	<input checked="" type="checkbox"/>	Redirección: <code>_server_/consultorios</code>	Se accede a la página de destino como corresponde

Comentario:

Automatizado en CP02_09Test.php

Diseñada por: Matías Milesi

Programada por Nelson Inalbon



Rutas del Sistema

Caso de Prueba CP2-12

Prioridad: Alto
Fecha de Pueba: 4/10/2019

Prueba de las rutas iniciales del sistema

Descripción:

Aceptación

Probar si la web responde correctamente al requerimiento de las URL implementadas en la primer etapa. (con código nulo)

Condiciones Previas:

El Sistema debe tener cargado datos de prueba.

Paso N°	Navegar en la Web a las rutas generadas como base para el sistema.		Resultado Esperado	Resultado Actual
1	Datos: Nota:	<input checked="" type="checkbox"/>	El sistema permite acceder a los contenidos del sitio correctamente.	La página carga correctamente

Comentario:

Automatizada

Diseñada por: Nelson Inalbon

Programada por Matías Milesi



Sprint 3

ABM de Pacientes

Caso de Prueba CP3-01

Prioridad: Medio

Fecha de Pueba: 15/10/2019

Prueba de creación de un Paciente

Descripción:

Aceptación

Verificación del alta de un paciente en el sistema

Condiciones Previas:

Debe estar logeado al sistema con roles de administrador, secretario o kinesiólogo.

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	Ingresar al sistema, con rol de administrador, secretario o kinesiólogo Datos: <code>_server_login</code> Nota:	<input checked="" type="checkbox"/>	<code>_server_dashboard</code>	Correctamente logeado
2	En el menú lateral, seleccionar, Pacientes y luego el submenú Nuevo Datos: <code>_server_dashboard</code> Nota:	<input checked="" type="checkbox"/>	<code>_server_pacientes/new</code>	Redirección correcta
3	Cargar datos del paciente Datos: Se debe ingresar nombre, apellido, fecha de nac., DNI, nacionalidad y email. Nota: Se prueban además, numerosos casos de datos "no validos" automatizados.	<input checked="" type="checkbox"/>		
4	Presionar el botón Guardar Datos: <code>_server_pacientes/new</code> Nota: Se muestra un mensaje de error en el primer dato inválido cargado, secuencialmente en el orden descrito en "Datos de Prueba" del paso 4	<input checked="" type="checkbox"/>	Redirección: <code>_server_pacientes/id_nuevo_paciente</code>	Correctamente guardado
5	Verificar la creación del Paciente Datos: Nota:	<input checked="" type="checkbox"/>	Verificación de datos según la carga de paso 3	Datos verificados correctamente

Comentario:

Automatizado en CP03-01Test.php

Diseñada por: Víctor Rosset

Programada por: Matias Milesi



Prueba de edición de un Paciente

Descripción:

Aceptación

Verificar la correcta edición de los datos de un Paciente.

Condiciones Previas:

Debe estar logeado al sistema con permisos de administrador, secretario o kinesiólogo.

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	Ingresar al sistema, con rol de administrador, secretario o kinesiólogo Datos: <code>_server_/login</code> Nota:	<code>_server_/dashboard</code>	Correctamente logeado
2	En el menú lateral, seleccionar, Pacientes y luego Todos Datos: <code>_server_/dashboard</code> Nota:	<code>_server_/pacientes</code>	Correctamente direccionado
3	Seleccionar un paciente a editar Datos: <code>server_/pacientes</code> Nota:	<code>_server_/pacientes/id_pacient e/edit</code>	Correctamente direccionado
4	Edita los datos del paciente Datos: <code>__server_/pacientes/id_paciente/edit</code> - nombre, apellido, fecha de nac., DNI, nacionalidad y mail. Nota: Se prueban además, numerosos casos con datos no válidos	-	
5	Presionar el botón Guardar Datos: <code>server_/pacientes/id_paciente/edit</code> Nota: Se muestra un mensaje de error en el primer dato editado con un valor inválido.	<code>_server_/pacientes/id_pacient e</code>	Correctamente guardados los cambios
6	Verificar la edición de los datos Datos: Nota:	Verificación de los datos editados del paciente	Datos verificados correctamente

Comentario:

Automatizado en `CP03_02Test.php`

Diseñada por: Victor Rosset

Programada por: Víctor Rosset



ABM de Kinesiólogos

Caso de Prueba CP3-04

Prioridad: Medio

Fecha de Pueba: 26/10/2019

Prueba de creación de Kinesiólogo

Descripción:

Aceptación

Prueba automatizada de creación de Kinesiólogos

Condiciones Previas:

Debe estar logeado al sistema con permisos de administrador, secretario o kinesiólogo.

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	Ingresar a la Web de creación de Kinesiólogo Datos: <code>_server_/kinesiólogos/new</code> Nota: Automatizar desde aquí	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	Cargar datos del Kinesiólogo nuevo Datos: Generar datos de prueba para tener 100 % de Cobertura de código Nota:	<input checked="" type="checkbox"/>		
3	Presionar el botón Guardar Datos: Nota:	<input checked="" type="checkbox"/>	Redirección al Kinesiólogo creado	
4	Verificar la creación del Kinesiólogo Datos: Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	

Comentario:

Automatizado en CP03-07Test.php

Diseñada por: Matías Milesi

Programada por: Víctor Rosset

APROBADA?


Prueba de edición de un Kinesiólogo

Descripción:

Aceptación

Prueba automatizada de edición de un Kinesiólogos

Condiciones Previas:

Debe estar logeado al sistema con permisos de administrador, secretario o kinesiólogo.

Paso N°	Acceder a la interfaz de edición de Kinesiólogo		Resultado Esperado	Resultado Actual
1	Datos: <code>_server_/kinesiólogos/edit/d_kinesiologo</code> Nota:	<input checked="" type="checkbox"/>	Acceso a la interfaz de Edición de Kinesiólogo	La página se cargó correctamente.
2	Datos: Edición de los diferentes datos del Kinesiólogo Nota: Se prueban además, numerosos cáos de datos "no validos" automatizados	<input checked="" type="checkbox"/>	Edición de datos Exitoso	Campos correctamente editados
3	Datos: Nota:	<input checked="" type="checkbox"/>	Redirección: <code>_server_/kinesiólogos/id_kinesiologo</code>	Acceso a la página con los datos del kinesiólogo editado

Comentario:

Automatizado en CP03-08Test.php

Diseñada por: Nelson Inalbon

Programada por: Matías Milesi



Prueba de Eliminación de un Kinesiólogo

Descripción:

Aceptación

Probar si se puede eliminar un Kinesiólogo previamente creado

Condiciones Previas:

Se debe haber realizado la prueba CP3-14

Paso N°			Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Kinesiólogo" y luego la opción "Todos"</p> <p>Datos: <code>_server_/dashboard</code></p> <p>Nota: Para facilitar la realización de las Pruebas, ver Manual de Usuario</p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente	
2	<p>Seleccione el Kinesiólogo del paciente con nombre "Pruebas", buscándolo en el buscador y presionando "Buscar"</p> <p>Datos: <code>ver /kinesiólogos/edit/id Kinesiólogo.</code></p> <p>Nota:</p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
3	<p>sobre el registro, presione el botón, "Ver"</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>		
4	<p>Presionar el botón Eliminar</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Redirección a la página de los Kinesiólogos	

Comentario:

Diseñada por: Matias Milesi

Programada por: Nelson Inalbon



ABM de Patologías

Caso de Prueba CP3-07

Prioridad: Medio
Fecha de Pueba: 26/10/2019

Prueba de Creación de una Patología

Descripción:

Aceptación

Probar que se puede crear una Patología

Condiciones Previas:

Debe estar logeado al sistema con permisos de administrador, secretario o kinesiólogo.

Paso N°	Descripción		Resultado Esperado	Resultado Actual
1	Ingresar a la Web de creación de una nueva Patología Datos: _server_/patologias/new Nota: Automatizar desde aquí	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	Cargar datos de la Patología nueva. Datos: Generar datos de prueba para tener 100 % de Cobertura de código Nota:	<input checked="" type="checkbox"/>		
3	Presionar el botón Guardar Datos: Nota:	<input checked="" type="checkbox"/>	Redirección a la patología creada	
4	Verificar la creación de la Patología Datos: Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	

Comentario:

Automatizado en CP03-10Test.php

Diseñada por: Nelson Inalbon

Programada por: Víctor Rosset



Caso de Prueba CP3-14

Prioridad: Medio
Fecha de Pueba: 26/10/2019

Prueba de edición de una patología

Descripción:

Aceptación

Probar que se pueden editar las Patologías.

Condiciones Previas:

Debe estar logeado al sistema con permisos de administrador, secretario o kinesiólogo.

Paso N° 1	<p>Acceder a la interfaz de edición de la Patología</p> <p>Datos: <code>_server_/patologias/edit/id_patologia</code></p> <p>Nota:</p>	<input checked="" type="checkbox"/>	<p><i>Resultado Esperado</i></p> <p>Acceso a la interfaz de Edición de la Patología</p>	<p><i>Resultado Actual</i></p> <p>La página se cargó correctamente.</p>
Paso N° 2	<p>Editar datos de la Patología</p> <p>Datos: Edición de los diferentes datos de la Patología</p> <p>Nota: Se prueban además, numerosos casos de datos "no validos" automatizados</p>	<input checked="" type="checkbox"/>	<p><i>Resultado Esperado</i></p> <p>Edición de datos Exitoso</p>	<p><i>Resultado Actual</i></p> <p>Campos correctamente editados</p>
Paso N° 3	<p>Presionar el botón Guardar</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	<p><i>Resultado Esperado</i></p> <p>Redirección. <code>_server_/patologias/id_patologia</code></p>	<p><i>Resultado Actual</i></p> <p>Acceso a la página con los datos de la Patología editada</p>

Comentario:

Automatizada en CP03-11Test.php

Diseñada por: Matías Milesi

Programada por: Nelson Inalbon



Sprint 4

ABM de Turnos

Caso de Prueba CP4-01

Prioridad: Medio
Fecha de Pueba: 1/1/2020

Prueba de creación de un Turno

Descripción:

Aceptación

Verifique que puede dar de alta un Turno.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción		Resultado Esperado	Resultado Actual
1	Ingresar a la Web de creación de Turno Datos: _server_turnos/new Nota: Automatizar desde aquí	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	Cargar datos del turno nuevo Datos: Generar datos de prueba para tener 100 % de Cobertura de código Nota:	<input checked="" type="checkbox"/>		
3	Presionar el botón Guardar Datos: Nota:	<input checked="" type="checkbox"/>	Redirección al turno creado	
4	Verificar la creación del Turno Datos: Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	

Comentario:

Automatizado en CP03-04Test.php

Diseñada por: Nelson Inalbon

Programada por: Víctor Rosset



Caso de Prueba CP4-02

Prueba de edición de un Turno

Descripción:

Aceptación

Verifique que puede editar un Turno.

Condiciones Previas:

Debe estar logeado al sistema. Deben existir turno creados

Paso N°			Resultado Esperado	Resultado Actual
1	Ir a la página de edición de Turnos. Datos: <code>_server_/turnos/Id_seleccionado/edit</code> Nota:	✓	La página debería cargar correctamente	
2	Editar los campos del Turno alternando variantes. Probar cambiar Paciente, Kinesiólogo y fechas Datos: <code>_server_/turnos/Id_scliccionado/edit</code> Nota:	✓		
3	Presionar el botón Guardar Datos: Generar datos de prueba para tener 100 % de Cobertura de código Nota:	✓	Redirección al turno editado	
4	Verificar la edición del Turno Datos: Nota:	✓	La página debería cargar correctamente y los datos deberían haberse editado	La página carga correctamente

Comentario:

Automatizado en CP03-05Test.php

Diseñada por: Víctor Rosset

Programada por Víctor Rosset



8.11.2 Casos de Prueba de Aceptación:

Sprint 2

Interfaz de Usuarios

Caso de Prueba CPA2-01

Prioridad: Medio
Fecha de Pueba: 28/9/2019

Prueba de inicio de sesión

Descripción:

Aceptación

Verificar el correcto inicio de sesión con un usuario y contraseña válidos

Condiciones Previas:

Debe haber usuarios y roles creados.

Paso N° 1	Ingresar a la página de Login Datos: <code>server /login</code> Nota: Prueba a realizar en: Workstations y Servidor en Producción.	<input checked="" type="checkbox"/>	<i>Resultado Esperado</i>	<i>Resultado Actual</i>
			Debe cargar la página de Login	Página carga correctamente
Paso N° 2	Proporcionar un nombre de usuario válido Datos: <code>matias</code> Nota: Las credenciales correctas dependerán del entorno de ejecución de las pruebas.	<input checked="" type="checkbox"/>	<i>Resultado Esperado</i>	<i>Resultado Actual</i>
			Campo relleno	Campo completo
Paso N° 3	Proporcionar una contraseña válida para ese usuario. Datos: <code>contrasenia11</code> Nota: Para facilitar la realización de las Pruebas, ver Manual de Usuario	<input checked="" type="checkbox"/>	<i>Resultado Esperado</i>	<i>Resultado Actual</i>
			Campo relleno	Campo completo
Paso N° 4	Presionar el botón Ingresar/Sign In Datos: Nota:	<input checked="" type="checkbox"/>	<i>Resultado Esperado</i>	<i>Resultado Actual</i>
			El sistema debería permitir el login	Login correcto
Paso N° 5	Verificar el ingreso y redirección al dashboard Datos: Nota:	<input checked="" type="checkbox"/>	<i>Resultado Esperado</i>	<i>Resultado Actual</i>
			El sistema redirecciona a la página de Dashboard	Redirección y carga de página de Dashboard exitosa

Comentario:

Prueba a realizar Manualmente

Diseñada por: Matias Milesi

Asistida Por: Matias Milesi



Prioridad: Medio

Caso de Prueba CPA2-02

Fecha de Pueba: 28/9/2019

Prueba de inicio de sesión con un usuario y/o contraseña no valido

Descripción:

Aceptación

Verificar la negativa de inicio de sesión con un usuario y/o contraseña no valido.

Condiciones Previas:

Debe haber usuarios y roles creados.

Paso N° 1	Ingresar a la página de Logín Datos: server /login Nota: Prueba a realizar en: Workstations y Servidor en Producción.	<input checked="" type="checkbox"/>	Resultado Esperado Debe cargar la página de Login	Resultado Actual Pagina carga correctamente
Paso N° 2	Proporcionar un nombre de usuario inválido Datos: matias Nota: Combinaciones hasta lograr cubrir la mayor cantidad de combinaciones posibles	<input checked="" type="checkbox"/>	Resultado Esperado Campo relleno	Resultado Actual Campo completo
Paso N° 3	Proporcionar una contraseña. Datos: contraseniaa11 Nota: Se deben realizar pruebas de usuario y contraseñas inválidos.	<input checked="" type="checkbox"/>	Resultado Esperado Campo relleno	Resultado Actual Campo completo
Paso N° 4	Presionar el botón Ingresar/Sign In Datos: Nota:	<input checked="" type="checkbox"/>	Resultado Esperado El sistema debería rechazar el login, informando un mensaje de Login inválido.	Resultado Actual El sistema rechaza el login

Comentario:

Prueba a realizar manualmente

Diseñada por: Matías Milesi

Asistida Por: Matías Milesi



ABM de Usuarios

Caso de Prueba CPA2-10

Prioridad: Medio
Fecha de Pueba: 4/10/2019

Prueba de Creación de nuevo usuario

Descripción:

Aceptación

Verifique que puede dar de alta un nuevo usuario en el sistema

Condiciones Previas:

Debe haber un usuario Administrador creado

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	Ingresar al sistema con un usuario que tenga el rol de Administrador. Datos: <code>_server_login</code> Nota:	<input checked="" type="checkbox"/>	Logeado correcto	La página carga correctamente
2	En el menú lateral, seleccionar "Usuarios" y luego la opción "Nuevo" Datos: <code>_server_dashboard</code> Nota:	<input checked="" type="checkbox"/>	Redirección a la página de creación de Usuarios	Redirección exitosa
3	Ingresar a la Web de creación de Usuarios Datos: <code>server/users/new</code> Nota:	<input checked="" type="checkbox"/>	La pagina deberia cargar correctamente	La página carga correctamente
4	Cargar datos del usuario nuevo Datos: Nombre: "Nelson"; Apellido: "Inalbon"; Usuario: "nelson" Nota:	<input checked="" type="checkbox"/>	El sistema debería permitir ingresar los valores	Permite
5	Presionar el botón Guardar Datos: Nota:	<input checked="" type="checkbox"/>	Redirección: <code>_server_users/view/id_creado</code>	Redirección: <code>_server_users/view/id_creado</code>
6	Verificar la creación del Usuario Datos: Nota:	<input checked="" type="checkbox"/>	Los datos son verificados correctamente	Datos verificados correctamente

Comentario:

Prueba Manual de Creación de Usuario

Diseñada por: Matías Milesi

Asistida Por: Nelson Inalbon



Prueba de edición de un usuario

Descripción:

Aceptación

Verifique que puede editar un usuario en el sistema

Condiciones Previas:

Debe haber un usuario Administrador creado

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	Ingresar al sistema con un usuario que tenga el rol de Administrador. Datos: server /login Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	En el menú lateral, seleccionar "Usuarios" y luego la opción "Todos" Datos: _server_/dashboard Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
3	En la página usuarios seleccionar un Usuario y presionar el Botón "Editar" Datos: _server_/users Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
4	Ingresar a la Web de Edición de Usuarios Datos: _server_/users/edit/id_editado Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
5	Cambiar datos del usuario seleccionado Datos: Nombre: "Nelson" por "Victor", Apellido: "Inalbon" por "Milesi", Usuario: "inlson" por "marcos" Nota:	<input checked="" type="checkbox"/>	El sistema debería permitir ingresar los valores	El sistema permite ingresar los valores
6	Presionar el botón Guardar Datos: Nota:	<input checked="" type="checkbox"/>	Redirección: _server_/users/view/id_editado	Redirección: _server_/users/view/id_editado
7	Verificar la edición del Usuario Datos: Nota:	<input checked="" type="checkbox"/>	Los datos son verificados correctamente	Datos verificados correctamente

Comentario:

Prueba manual

Diseñada por: Nelson Inalbon

Asistida Por: Matias Milesi



Sprint 3

ABM de Kinesiólogos

Caso de Prueba CPA3-08

Prioridad: Medio

Fecha de Pueba: 25/10/2019

Prueba de creación de un Kinesiólogo

Descripción:

Aceptación

Verifique que puede dar de alta un Kinesiólogo.

Condiciones Previas:

Debe estar logeado al sistema - Pruebas manuales dependientes - Orden: 1° CPA3-08 2° CPA3-09

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Kinesiólogo" y luego la opción "Nuevo"</p> <p>Datos: <code>_server_/dashboard</code></p> <p>Nota:</p>	La página debería cargar correctamente	La página carga correctamente
2	<p>Ingresar a la Web de creación de Kinesiólogo</p> <p>Datos: <code>_server_/kinesiólogos/new</code></p> <p>Nota:</p>	La página debería cargar correctamente	La página carga correctamente
3	<p>Cargar datos del turno nuevo.</p> <p>Datos: Crear un Kinesiólogo con nombre "Pruebas" y apellido "PruebaApellido"</p> <p>Nota: Los otros datos están testeados en Automatización, así que puede probar la combinación en el momento</p>		
4	<p>Presionar el botón Guardar</p> <p>Datos:</p> <p>Nota:</p>	Redirección al Kinesiólogo creado	Redirección exitosa
5	<p>Verificar la creación del Kinesiólogo</p> <p>Datos:</p> <p>Nota:</p>	La página debería cargar correctamente	La página muestra la información

Comentario:

Procedimiento Manual

Diseñada por: Víctor Rosset

Asistida Por: Víctor Rosset



Prioridad: Medio

Caso de Prueba CPA3-09

Fecha de Pueba: 25/10/2019

Prueba de edición de un Kinesiólogo

Descripción:

Aceptación

Verifique que puede editar un Kinesiólogo.

Condiciones Previas:

Se debe haber realizado la prueba CP3-14 - Pruebas manuales dependientes - Orden: 1° CPA3-08 2° CPA3-09

Paso N° 1	En el menú lateral, seleccionar "Kinesiólogo" y luego la opción "Todos" Datos: <code>_server_/dashboard</code> Nota:	<input checked="" type="checkbox"/>	Resultado Esperado	Resultado Actual
			La página debería cargar correctamente	La página carga correctamente
Paso N° 2	Seleccione el Kinesiólogo del paciente con nombre "Pruebas", buscándolo en el buscador y presionando "Buscar" Datos: <code>_server_/kinesiólogos/edit/Id_Kinesiólogo.</code> Nota:	<input checked="" type="checkbox"/>	Resultado Esperado	Resultado Actual
			La página debería cargar correctamente	La página carga correctamente
Paso N° 3	sobre el registro, presione el botón, "Editar" Datos: Nota:	<input checked="" type="checkbox"/>	Resultado Esperado	Resultado Actual
Paso N° 4	Editar los campos del Kinesiólogo alternando variantes. Datos: <code>_server_/kinesiólogos/Id_seleccionado/edit - Cambiar Nombre y Apellido, de Pruebas a TestNombre, y PruebaApellido a TestApellido.</code> Nota:	<input checked="" type="checkbox"/>	Resultado Esperado	Resultado Actual
			El sistema debería permitir editar los datos	El sistema permite editar los valores
Paso N° 5	Presionar el botón Guardar Datos: Nota:	<input checked="" type="checkbox"/>	Resultado Esperado	Resultado Actual
			Redirección al Kinesiólogo editado	Redirección correcta
Paso N° 6	Verificar la edición del Kinesiólogo Datos: Nota:	<input checked="" type="checkbox"/>	Resultado Esperado	Resultado Actual
			La página debería cargar correctamente y los datos deberían haberse editado	La página carga correctamente

Comentario:

Procedimiento Manual

Diseñada por: Víctor Rosset

Asistida Por: Víctor Rosset



ABM de Patologías

Caso de Prueba CPA3-10

Prioridad: Medio

Fecha de Pueba: 25/10/2019

Prueba de creación de una Patología

Descripción:

Aceptación

Verifique que puede dar de alta una Patología

Condiciones Previas:

Debe estar logeado al sistema - Pruebas manuales dependientes - Orden: 1° CPA3-10 - 2° CPA3-13

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Patologías" y luego la opción "Nueva"</p> <p>Datos: _server_/dashboard</p> <p>Nota:</p>	<p>La página debería cargar correctamente</p>	<p>La página carga correctamente</p>
2	<p>Ingresar a la Web de creación de Patologías</p> <p>Datos: _server_/patologias/new</p> <p>Nota:</p>	<p>La página debería cargar correctamente</p>	<p>La página carga correctamente</p>
3	<p>Cargar datos de la Patología Nueva</p> <p>Datos: Crear una Patología con nombre "Pruebas"</p> <p>Nota: Los otros datos están testeados en Automatización, así que puede probar la combinación en el momento</p>		
4	<p>Presionar el botón Guardar</p> <p>Datos:</p> <p>Nota:</p>	<p>Redirección a la Patología creada</p>	
5	<p>Verificar la creación de la Patología</p> <p>Datos:</p> <p>Nota:</p>	<p>La página debería cargar correctamente</p>	

Comentario:

Procedimiento Manual

Diseñada por: Víctor Rosset

Asistida Por: Víctor Rosset



Probar asociar un Paciente a una Obra Social

Descripción:

Aceptación

Probar asociar un paciente a una Obra social estableciendo un Plan en esa asociación.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Pacientes"->"Todos" Datos: server /dashboard Nota:	<input checked="" type="checkbox"/>	La página debería cargar	La página carga correctamente
2	Buscar el Paciente con nombre "Prueba", colocando Pruebas en la barra de búsqueda y presionando Buscar Datos: Nombre: "Prueba" Nota:	<input checked="" type="checkbox"/>	Debería permitir buscar los valores	Permite la búsqueda
3	En la sección de Obras Sociales, presionar el botón: "Agregar Obra Social" Datos: Nota:	<input checked="" type="checkbox"/>	Debería figurar el botón para presionarlo	Se realiza el click
4	Completar los datos de la asociación con el Plan Datos: Nota:	<input checked="" type="checkbox"/>	Debería permitir ingresar información del plan	Permite
5	Presionar Guardar Datos: Nota:	<input checked="" type="checkbox"/>	Deberá guardar los datos	Actualiza la página
6	Verificar la creación de la asociación. Datos: Nota:	<input checked="" type="checkbox"/>	La asociación debería haber sido creada	Se verifica la asociación

Comentario:

Diseñada por: Nelson Inalbon

Asistida Por: Nelson Inalbon



Gestión de Turnos Rápidos

Caso de Prueba CPA3-12

Prioridad: Medio
Fecha de Prueba: 25/10/2019

Prueba de creación de un Turno Rápido

Descripción:

Aceptación

Verifique que puede dar de alta un Turno Rápido.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	✓	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Turnos" y luego la opción "Turno Rápido" Datos: _server_/dashboard Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	Ingresar a la Web de creación de Turno Datos: _server_/turno-rápido/new Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
3	Cargar datos del turno nuevo Datos: Crear un turno con Título "Turno Rápido" Nota: Los otros datos están testeados en Automatización, así que puede probar la combinación en el momento	<input checked="" type="checkbox"/>	El sistema debería permitir ingresar los datos	Permite
4	Presionar el botón Guardar Datos: Nota:	<input checked="" type="checkbox"/>	Redirección al turno creado	Redirección exitosa
5	Verificar la creación del Turno Datos: Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente. Mostrando los datos correctos	La página muestra la información apropiada

Comentario:

Procedimiento Manual

Diseñada por: Víctor Rosset

Asistida Por: Víctor Rosset

APROBADA?



ABM de Patologías

Caso de Prueba CPA3-13

Prioridad: Medio
Fecha de Prueba: 25/10/2019

Prueba de edición de una Patología

Descripción:

Aceptación

Verifique que puede editar una Patología

Condiciones Previas:

Se debe haber realizado la prueba CP3-16 - Pruebas manuales dependientes - Orden: 1° CPA3-10 - 2° CPA3-13

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Patologías" y luego la opción "Todas"</p> <p>Datos: <code>_server_/dashboard</code></p> <p>Nota:</p>	La página debería cargar correctamente	La página carga correctamente
2	<p>Seleccione la Patología con nombre "Pruebas", buscándolo en el buscador y presionando "Buscar".</p> <p>Datos: <code>_server_/patologias/edit/id_Patologia</code></p> <p>Nota:</p>	La página debería cargar correctamente	La página carga correctamente
3	<p>sobre el registro, presione el botón, "Editar"</p> <p>Datos:</p> <p>Nota:</p>		
4	<p>Editar los campos de la Patología alternando variantes, según los datos de Prueba</p> <p>Datos: <code>_server_/patologias/ld_seleccionado/edit</code> - Cambiar Nombre de "Pruebas" a "TestNombre", luego repetir la prueba y volver a colocar el nombre "Pruebas"</p> <p>Nota: Para luego correr la prueba de eliminación, colocar el nombre en "Pruebas"</p>		
5	<p>Presionar el botón Guardar</p> <p>Datos:</p> <p>Nota:</p>	Redirección a la Patología editada	
6	<p>Verificar la edición de la Patología</p> <p>Datos:</p> <p>Nota:</p>	La página debería cargar correctamente y los datos deberían haberse editado	La página carga correctamente

Comentario:

Procedimiento Manual

Diseñada por: Víctor Rosset

Asistida Por: Víctor Rosset



Sprint 4

ABM de Turnos

Prioridad: Medio

Fecha de Pueba: 1/1/2020

Caso de Prueba CPA4-03

Prueba de eliminación de un Turno

Descripción:

Aceptación

Verificar que el sistema permite eliminar un turno determinado

Condiciones Previas:

Debe estar logeado al sistema, Deben existir turnos creados

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos" y luego la opción "Todos"</p> <p>Datos: <code>_server_/dashboard</code></p> <p><i>Nota: Para facilitar la realización de las Pruebas, ver Manual de Usuario</i></p>	La página debería cargar correctamente	La página debería cargar correctamente
2	<p>Filtre los paciente con el nombre "Pruebas" y presione buscar, presionado el boton "Editar" asociado al mismo en el grid</p> <p>Datos: <code>_server_/turnos</code></p> <p><i>Nota:</i></p>	La búsqueda debería ser exitosa	Búsqueda exitosa
3	<p>En la página presionar el botón "Eliminar" y "Confirmar su eliminación"</p> <p>Datos: <code>_server_/turnos/ld_seleccionado/edit</code></p> <p><i>Nota:</i></p>	Mensaje emergente solicitando confirmación	Mensaje correcto
4	<p>Verificar la eliminación del Turno</p> <p>Datos:</p> <p><i>Nota:</i></p>	Que no existe más el turno con el Paciente "Pruebas"	No existe más el turno con el Paciente "Pruebas"

Comentario:

Prueba manual

Diseñada por: Matías Milesi

Asistida Por: Víctor Rosset



Probar el análisis de disponibilidad de Turno según la disponibilidad

Descripción:

Aceptación

Se deberá probar la disponibilidad de los recursos para agregar un turno, según la disponibilidad de recursos en ese horario.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos" -> "Rehabilitaciones" -> "Nueva"</p> <p>Datos: <code>_server_/dashboard</code></p> <p>Nota: Buscar alguna fecha que esté libre de Recursos.</p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente.	La página carga correctamente.
2	<p>Completar los datos del Paciente, Kinesiólogo y Patología.</p> <p>Datos: <code>_server_/fichas-rehabilitaciones/new</code> - Paciente: "Matias Milesi"; Kinesiólogo: "Mauro"; Patología: "Estándar" y Tratamiento: "Estándar"</p> <p>Nota: Puede variar los datos en caso que considere necesario.</p>	<input checked="" type="checkbox"/>	El sistema debería permitir ingresar los datos.	El sistema permite ingresar los datos.
3	<p>Presionar sobre el botón "Martes y Jueves".</p> <p>Datos:</p> <p>Nota: Puede variar los días en caso que considere necesario.</p>	<input checked="" type="checkbox"/>	Se deberían tildar los checks de "Martes" y "Jueves".	Al presionar el botón la selección ocurre.
4	<p>Seleccione una hora y presione el botón calcular</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Deberían aparecer las fechas propuestas.	Aparecen las fechas.
5	<p>Presionar sobre el botón "Guardar"</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	El Sistema debería Redirigir a la vista de la Rehabilitación Creada.	Redirección.
6	<p>Revisar que la Rehabilitación y los turnos se hayan realizado correctamente</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>		Datos correcto.

Comentario:

Diseñada por: Víctor Rosset

Asistida Por: Nelson Inalbon



Caso de Prueba CPA4-05

Prueba de creación de un Turno

Descripción:

Aceptación

Verifique que puede dar de alta un Turno.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos" y luego la opción "Nuevo"</p> <p>Datos: _server_/dashboard</p> <p>Nota:</p>	La página debería cargar correctamente	La página carga correctamente
2	<p>Ingresar a la Web de creación de Turno</p> <p>Datos: _server_/turnos/new</p> <p>Nota:</p>	La página debería cargar correctamente	La página carga correctamente
3	<p>Cargar datos del turno nuevo</p> <p>Datos: Crear un paciente con nombre "Pruebas"</p> <p>Nota: Los otros datos están testeados en Automatización, así que puede probar la combinación en el momento</p>	El sistema deberá permitir ingresar los valores	El sistema permite ingresar
4	<p>Presionar el botón Guardar</p> <p>Datos:</p> <p>Nota:</p>	Redirección al turno creado	Redirección exitosa
5	<p>Verificar la creación del Turno</p> <p>Datos:</p> <p>Nota:</p>	La página debería cargar correctamente	Verificación exitosa

Comentario:

Procedimiento Manual

Diseñada por: Víctor Rosset

Asistida Por: Víctor Rosset



Prueba de edición de un Turno

Descripción:

Aceptación

Verifique que puede editar un Turno.

Condiciones Previas:

Existir turnos creados con el paciente "Pruebas"

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Turnos" y luego la opción "Todos" Datos: _server_/dashboard Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	Seleccione el Turno del paciente con nombre "Pruebas", buscándolo en el buscador y presionando el botón "Buscar" Datos: _server_/turnos. Paciente:"Pruebas", cambiar de "Kinesiólogo" asociado. Nota: Automatizar desde aquí	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
3	sobre el registro, presione el botón, "Editar" Datos: Nota:	<input checked="" type="checkbox"/>		
4	Editar los campos del Turno alternando variantes. Probar cambiar Paciente, Kinesiólogo y fechas Datos: _server_/turnos/ld_seleccionado/edit Nota:	<input checked="" type="checkbox"/>	El sistema debería permitir ingresar los valores	Permite
5	Presionar el botón Guardar Datos: Nota:	<input checked="" type="checkbox"/>	Redirección al turno editado	Redirección correcta
6	Verificar la edición del Turno Datos: Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente y los datos deberían haberse editado	La página carga correctamente

Comentario:

Procedimiento Manual

Diseñada por: Víctor Rosset

Asistida Por: Víctor Rosset



Probar asignar un turno a pesar de no tener recursos disponibles

Descripción:

Aceptación

Probar asignar un turno a pesar de no tener recursos disponibles, el resultado debería ser exitoso

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos" -> "Rehabilitaciones" -> "Nueva"</p> <p>Datos: <code>_server_/dashboard</code></p> <p>Nota: Buscar alguna fecha que no Tenga Recursos Disponibles, de forma que los mensajes sean de "Atención, recursos no disponibles"</p>	La página se debería mostrar correctamente	La página carga correctamente
2	<p>Completar los datos del Paciente, Kinesiólogo y Patología.</p> <p>Datos: <code>_server_/fichas-rehabilitaciones/new - Paciente: "Matías Milesi"; Kinesiólogo: "Mauro"; Patología: "Estándar" y Tratamiento: "Estándar"</code></p> <p>Nota: Puede variar los datos en caso que considere necesario.</p>	El sistema debería permitir ingresar los datos	Permite
3	<p>Presionar sobre el botón "Martes y Jueves"</p> <p>Datos:</p> <p>Nota: Puede variar los días en caso que considere necesario</p>	Se deberían tildar los checks de "Martes" y "Jueves"	Se tildan los días deseados
4	<p>Seleccione una hora y presione el botón calcular</p> <p>Datos:</p> <p>Nota:</p>	Deberían aparecer los días propuestos	Aparecen los días con los horarios propuestos
5	<p>Presionar sobre el botón "Guardar"</p> <p>Datos:</p> <p>Nota:</p>	El Sistema debería Redireccionar a la vista de la Rehabilitación Creada	Redirección
6	<p>Revisar que la Rehabilitación y los turnos se hayan realizado correctamente</p> <p>Datos:</p> <p>Nota:</p>		Datos correcto

Comentario:

Diseñada por: Nelson Inalbon

Asistida Por: Nelson Inalbon

APROBADA?

PASS

ABM de Tipos de Recursos y ABM de Recursos

Caso de Prueba CPA4-08

Prioridad: Medio
Fecha de Prueba: 2/1/2020

Prueba de creación de un Recurso Modelo

Descripción:

Aceptación

Probar que se puede crear un Recurso Modelo

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Recursos"-> "Tipo de Recurso" y luego la opción "Nuevo"</p> <p>Datos: <code>_server_/dashboard</code></p> <p>Nota: Para facilitar la realización de las Pruebas, ver Manual de Usuario</p>	La página debería cargar correctamente	Carga correctamente
2	<p>Ingresar a la Web de creación del Tipo de Recurso</p> <p>Datos: <code>_server_/recursomodelo/new</code></p> <p>Nota:</p>	La página debería cargar correctamente	La página carga correctamente
3	<p>Cargar datos del Tipo de Recurso Nuevo</p> <p>Datos: Crear un recurso modelo con el nombre "RecursoMPuebas", elegir íconos aleatoriamente.</p> <p>Nota: Los otros datos están testeados en Automatización, así que puede probar la combinación en el momento</p>	Debería poder ingresar los datos	Permite
4	<p>Presionar el botón Guardar</p> <p>Datos:</p> <p>Nota:</p>	Redirección al recurso modelo creado	Redirección exitosa
5	<p>Verificar la creación del Recurso Modelo</p> <p>Datos:</p> <p>Nota:</p>	La página debería cargar correctamente	Datos correctos

Comentario:

Prueba Manual

Diseñada por: Nelson Inalbon

Asistida Por: Matías Milesi



Caso de Prueba CPA4-09

Prueba de creación de un Recurso

Descripción:

Aceptación

Probar que se puede crear un Recurso y Asignarlo a un Recurso Modelo

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Recursos" y luego la opción "Nuevo"</p> <p>Datos: _server_/dashboard</p> <p>Nota: Para facilitar la realización de las Pruebas, ver Manual de Usuario</p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	<p>Ingresar a la Web de creación del Recurso</p> <p>Datos: _server_/recursos/new</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
3	<p>Cargar datos del Recurso Nuevo</p> <p>Datos: Crear una Recurso con nombre "RecursoPruebas" y que tenga asociado al Recurso Modelo "RecursoMPuebas"</p> <p>Nota: Los otros datos están testeados en Automatización, así que puede probar la combinación en el momento</p>	<input checked="" type="checkbox"/>	Debería permitir ingresar los datos	Permite
4	<p>Presionar el botón Guardar</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Redirección al Recurso Creado	Redirección exitosa
5	<p>Verificar la creación del Recurso y la correcta asociación al Recurso Modelo.</p> <p>Datos:</p> <p>Nota: Ok</p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente	Carga exitosa

Comentario:

Prueba manual

Diseñada por: Nelson Inalbon

Asistida Por: Matías Milesi



Caso de Prueba CPA4-10

Prueba de Edición de un Recurso Modelo

Descripción:

Aceptación

Probar que se puede Editar un Recurso Modelo

Condiciones Previas:

Haber realizado las pruebas CP04-01, CP04-02 - Pruebas manuales dependientes - Orden: 1° CPA4-08y09 - 2° CPA4-10y11 - 3° CPA4-12y13

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Recursos"-> "Tipo de Recurso" y luego la opción "Todas"</p> <p>Datos: _server_/dashboard</p> <p>Nota: Para facilitar la realización de las Pruebas, ver Manual de Usuario</p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	<p>Seleccione el Tipo de Recurso con nombre "RecursoMPuebas", buscándolo en el buscador y presionando el botón "Buscar"</p> <p>Datos: _server_/recurso/id_rm/edit</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Se debería poder buscar	Búsqueda exitosa
3	<p>sobre el registro, presione el botón, "Editar"</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	presionar	presionado
4	<p>Editar los campos del Recurso Modelo alternando variantes, según los datos de Prueba</p> <p>Datos: _server_/recurso/id_rm/edit - Cambiar Nombre de "RecursoMPuebas" a "RecursoMPuebasTest", luego repetir la prueba y volver a colocar el nombre "RecursoMPuebas"</p> <p>Nota: Para luego correr la prueba de eliminación, colocar el nombre en "Pruebas"</p>	<input checked="" type="checkbox"/>	Se debería poder editar los datos	Edición exitosa
5	<p>Presionar el botón Guardar</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Redirección al Recurso Modelo Editado	Redirección ok
6	<p>Verificar la edición de la Patología</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente y los datos deberían haberse editado	La página carga correctamente

Comentario:

Prueba manual

Diseñada por: Nelson Inalbon

Asistida Por: Nelson Inalbon



Prueba edición de un Recurso

Descripción:

Aceptación

Probar que se puede editar un Recurso

Condiciones Previas:

Haber realizado las pruebas CP04-01, CP04-02

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Recursos" y luego la opción "Todas"</p> <p>Datos: _server_/dashboard</p> <p>Nota: Para facilitar la realización de las Pruebas, ver Manual de Usuario</p>	La página debería cargar correctamente	La página carga correctamente
2	<p>Seleccione el Recurso con nombre "RecrusoPruebas", buscándolo en el buscador y presionando "Buscar".</p> <p>Datos: _server_/recursos/edit/id_rec</p> <p>Nota:</p>	Debería permitir realizar la búsqueda	Búsqueda ok
3	<p>sobre el registro, presione el botón, "Editar"</p> <p>Datos:</p> <p>Nota:</p>		
4	<p>Editar los campos de la Recurso alternando variantes, según los datos de Prueba</p> <p>Datos: _server_/recursos/id_rec/edit - Cambiar Nombre de "RecrusoPruebas" a "RecrusoPruebasTest", luego repetir la prueba y volver a colocar el nombre "RecrusoPruebas"</p> <p>Nota: Para luego correr la prueba de eliminación, colocar el nombre en "Pruebas"</p>	Debería permitir editar los datos	Permite
5	<p>Presionar el botón Guardar</p> <p>Datos:</p> <p>Nota:</p>	Redirección al Recurso editado.	Redirección exitosa
6	<p>Verificar la edición del Recurso</p> <p>Datos:</p> <p>Nota:</p>	La página debería cargar correctamente y los datos deberían haberse editado	La página carga correctamente

Comentario:

Prueba Manual

Diseñada por: Nelson Inalbon

Asistida Por: Nelson Inalbon



Caso de Prueba CPA4-12

Prueba de Eliminación de un Recurso Modelo

Descripción:

Aceptación

Verificar que se puede eliminar un Recurso Modelo

Condiciones Previas:

Haber realizado las pruebas CP04-01, CP04-02 - Pruebas manuales dependientes - Orden: 1° CPA4-08y09 - 2° CPA4-10y11 - 3° CPA4-12y13

Paso N° 1	<p>En el menú lateral, seleccionar "Recursos"-> "Tipo de Recurso" y luego la opción "Todos"</p> <p>Datos: <code>_server_/dashboard</code></p> <p><small>Nota: Para facilitar la realización de las Pruebas, ver Manual de Usuario</small></p>	<input checked="" type="checkbox"/>	<p>Resultado Esperado</p> <p>La página debería cargar correctamente</p>	<p>Resultado Actual</p> <p>La página carga correctamente</p>
Paso N° 2	<p>Seleccione el Tipo de Recurso con nombre "RecursosMPuebas", buscándolo en el buscador y presionando "Buscar".</p> <p>Datos: <code>_server_/recursosmodelo/show/id_rm</code></p> <p><small>Nota:</small></p>	<input checked="" type="checkbox"/>	<p>Resultado Esperado</p> <p>La página debería cargar correctamente</p>	<p>Resultado Actual</p> <p>La página carga correctamente</p>
Paso N° 3	<p>sobre el registro, presione el botón, "Ver"</p> <p>Datos:</p> <p><small>Nota:</small></p>	<input checked="" type="checkbox"/>	<p>Resultado Esperado</p> <p>botón disponible</p>	<p>Resultado Actual</p> <p>botón disponible</p>
Paso N° 5	<p>Presionar el botón Eliminar</p> <p>Datos:</p> <p><small>Nota:</small></p>	<input checked="" type="checkbox"/>	<p>Resultado Esperado</p> <p>Redirección a la página de los Recursos Mod.</p>	<p>Resultado Actual</p> <p>Redirección correcta</p>

Comentario:

Prueba Manual

Diseñada por: Nelson Inalbon

Asistida Por: Matías Milesi



Caso de Prueba CPA4-13

Prueba de Eliminación de un Recurso

Descripción:

Aceptación

Verificar que se puede Eliminar un Recurso

Condiciones Previas:

Haber realizado las pruebas CP04-01, CP04-02. En caso de haber realizado la prueba CP04-06, volver a correr la CP04-01.

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Recursos" y luego la opción "Todos"</p> <p>Datos: <code>_server_/dashboard</code></p> <p>Nota: Para facilitar la realización de las Pruebas, ver Manual de Usuario</p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	<p>Seleccione el Recurso con nombre "RecursoPruebas", buscándolo en el buscador y presionando "Buscar".</p> <p>Datos: <code>_server_/recursos/show/id_rec</code></p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Debería permitir la búsqueda	Permite la búsqueda
3	<p>sobre el registro, presione el botón, "Ver"</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	botón disponible	ok
5	<p>Presionar el botón Eliminar</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Redirección a la página de los Recursos	Redirección exitosa

Comentario:

Prueba manual

Diseñada por: Matías Milesi

Asistida Por: Nelson Inalbon



ABMde Obras sociales

Caso de Prueba CPA4-14

Prioridad: Medio
Fecha de Pueba: 5/1/2020

Prueba de creación de un Arancel de una Obra Social

Descripción:

Aceptación

Probar que se pueden crear un Arancel de un Tratamiento en una Obra Social

Condiciones Previas:

Debe estar loaeado al sistema

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Obra Social"->"Todas"</p> <p>Datos:</p> <p>Nota:</p>	<p>La página debería cargar correctamente; ruta: "_server_/dashboard"</p>	<p>La página carga correctamente</p>
2	<p>Elegir una Obra Social y presionar el botón "Ver"</p> <p>Datos:</p> <p>Nota:</p>	<p>Debería mostrar las Obras Sociales</p>	<p>Muestra correctamente</p>
3	<p>En la Sección de "Aranceles de Tratamientos", buscar algún Tratamiento que no tenga ingresado aranceles. Presione el botón "Cargar"</p> <p>Nota:</p>	<p>Redirección a la página: "_server_/aranceles/obras-sociales/id_os/tratamiento/id_Tratamiento/new"</p>	<p>Redirección exitosa</p>
4	<p>Modifique el nombre y Monto</p> <p>Datos: Nombre: "Prueba de Arancel"; Monto:"1234.56"</p> <p>Nota:</p>	<p>Debería permitir modificar los datos</p>	<p>Permite</p>
5	<p>Presione sobre el botón "Guardar"</p> <p>Datos:</p> <p>Nota:</p>	<p>Redirección a al página de la "Obra Social"</p>	<p>Redirección exitosa</p>
6	<p>Revisar que el Arancel se haya agregado correctamente.</p> <p>Datos:</p> <p>Nota:</p>	<p>La página de la Obra social debería presentar los datos ingresados.</p>	<p>Datos correctos</p>

Comentario:

Diseñada por: Nelson Inalbon

Asistida Por: Nelson Inalbon



Prueba de Edición de un Arancel de Obra Social

Descripción:

Aceptación

Probar que se puede editar un Arancel de la Obra Social.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Obra Social"->"Todas"</p> <p>Datos:</p> <p>Nota:</p>	<p>La página debería cargar correctamente; ruta: "_server_/dashboard"</p>	<p>La página carga correctamente</p>
2	<p>Elegir una Obra Social y presionar el botón "Ver"</p> <p>Datos:</p> <p>Nota:</p>	<p>Debería mostrar las Obras Sociales</p>	<p>Muestra las Obras Sociales</p>
3	<p>En la Sección de "Aranceles de Tratamientos", buscar el Arancel llamado "Prueba de Arancel". Presione el botón "Editar"</p> <p>Datos:</p> <p>Nota:</p>	<p>Redirección a la página: "_server_/aranceles/obras-sociales/id_os/tratamiento/id_Tratamiento/new"</p>	<p>Redirección ok</p>
4	<p>Ingrese un nombre y Monto</p> <p>Datos: Nombre: "Prueba de Arancel Test"; Monto:"1554.33" - Volver a realizar la prueba y colocar nuevamente: Nombre: "Prueba de Arancel"; Monto:"1234.56"</p> <p>Nota:</p>	<p>Debería permitir editar los valores</p>	<p>Permite editar</p>
5	<p>Presione sobre el botón "Guardar"</p> <p>Datos:</p> <p>Nota:</p>	<p>Redirección a al página de la "Obra Social"</p>	<p>Redirección exitosa</p>
6	<p>Revisar que el Arancel se haya agregado correctamente.</p> <p>Datos:</p> <p>Nota:</p>	<p>La página de la Obra social debería presentar los datos editados.</p>	<p>Presenta los datos editados.</p>

Comentario:

Diseñada por: Nelson Inalbon

Asistida Por: Nelson Inalbon



Sprint 5

ABM de Ordenes Médicas

Caso de Prueba CPA5-01

Prioridad: Medio
Fecha de Pueba: 6/2/2020

Prueba de Creación de una Orden Médica

Descripción:

Aceptación

Verificar el correcto funcionamiento de la Creación de una Orden Médica

Condiciones Previas:

Debe estar logeado al sistema - Pruebas manuales dependientes - Orden: 1° CPA5-01 - 2° CPA5-02 - 3° CPA5-03

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos"->"Ordenes Medicas" y luego la opción "Nueva"</p> <p>Datos: _server_/dashboard</p> <p>Nota:</p>	La página debería cargar correctamente	La página carga correctamente
2	<p>Ingresar a la Web de creación de la Ordenen Médica</p> <p>Datos: _server_/ordenes-medicas/new</p> <p>Nota:</p>	La página debería cargar correctamente	La página carga correctamente
3	<p>Cargar datos de la Órdenes Médicas Nueva</p> <p>Datos: Crear una Orden Médica asociada a una Rehabilitación colocando en Indicaciones "Indicaciones de Pruebas"</p> <p>Nota: Los otros datos están testeados en Automatización, así que puede probar la combinación en el momento</p>	Debería permitir ingresar los datos	Permite ingresar los datos
4	<p>Presionar el botón Guardar</p> <p>Datos:</p> <p>Nota:</p>	Redirección a la Orden Medica	Redirección correcta
5	<p>Verificar la creación de la Órdenes Médicas</p> <p>Datos:</p> <p>Nota:</p>	Los datos deberían ser los cargados	Los datos son correctos

Comentario:

Prueba manual

Diseñada por: Matías Milesi

Asistida Por: Matías Milesi



Caso de Prueba CPA5-02

Prueba de edición de Orden Médica

Descripción:

Aceptación

Probar que se puede editar una Orden Médica

Condiciones Previas:

Debe estar logeado al sistema - Pruebas manuales dependientes - Orden: 1° CPA5-01 - 2° CPA5-02 - 3° CPA5-03

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos"->"Ordenes Médicas" y luego la opción "Todas"</p> <p>Datos: <code>_server_/dashboard</code></p> <p>Nota:</p>	La página debería cargar correctamente	La página carga correctamente
2	<p>Seleccione la Orden médica con indicaciones "Indicaciones de Pruebas", buscándolo en el buscador y presionando "Buscar"</p> <p>Datos: <code>server_/ordenes-medicas/edit/id_om</code></p> <p>Nota:</p>	Debería permitir la búsqueda	Búsqueda exitosa
3	<p>sobre el registro, presione el botón, "Editar"</p> <p>Datos:</p> <p>Nota:</p>		
4	<p>Editar los campos de la Orden Médica alternando variantes, según los datos de Prueba</p> <p>Datos: <code>_server_/ordenes-medicas/ld_seleccionado/edit</code> - Cambiar las indicaciones de "Indicaciones de Pruebas" a "Ind de Pruebas" y luego repetir y volver a "Indicaciones de Pruebas"</p> <p>Nota: Para luego correr la prueba de eliminación, colocar la Indicación en "Indicaciones de Pruebas"</p>	Debería permitir la edición	Edición correcta
5	<p>Presionar el botón Guardar</p> <p>Datos:</p> <p>Nota:</p>	Redirección a la Órdenes Médicas editada	Redirección exitosa
6	<p>Verificar la edición de la Orden Médica</p> <p>Datos:</p> <p>Nota:</p>	La página debería cargar correctamente y los datos deberían haberse editado	La página carga correctamente

Comentario:

Prueba manual

Diseñada por: Matías Milesi

Asistida Por: Matías Milesi



Caso de Prueba CPA5-03

Prioridad: Medio
Fecha de Pueba: 6/2/2020

Prueba de Eliminación de Orden Médica

Descripción:

Aceptación

Probar que se puede eliminar una Orden Médica

Condiciones Previas:

Haber ejecutado la prueba CP04-07 previamente - Pruebas manuales dependientes - Orden: 1° CPA5-01 - 2° CPA5-02 - 3° CPA5-03

Paso N°	Descripción		Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos"->"Ordenes Medicas" y luego la opción "Todos"</p> <p>Datos: <code>_server_/dashboard</code></p> <p><small>Nota: Para facilitar la realización de las Pruebas, ver Manual de Usuario</small></p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	<p>Seleccione la Orden Medica con indicaciones "Indicaciones de Pruebas", buscándolo en el buscador y presionando "Buscar"</p> <p>Datos: <code>cas/show/id_om</code></p> <p><small>Nota:</small></p>	<input checked="" type="checkbox"/>	Debería permitir buscar	Búsqueda exitosa
3	<p>sobre el registro, presione el botón, "Ver"</p> <p>Datos:</p> <p><small>Nota:</small></p>	<input checked="" type="checkbox"/>	Debería existir el botón	ok
5	<p>Presionar el botón Eliminar</p> <p>Datos:</p> <p><small>Nota:</small></p>	<input checked="" type="checkbox"/>	Redirección a la página de Órdenes Médicas	Redirección exitosa

Comentario:

Prueba manual

Diseñada por: Matías Milesi

Asistida Por: Matías Milesi



Gestión por Calendario

Prioridad: Medio

Caso de Prueba CPA5-04

Fecha de Pueba: 1/2/2020

Prueba de mostrado de turnos en calendario

Descripción:

Aceptación

Probar que los datos del Turno se muestran correctamente en el Calendario.

Condiciones Previas:

Debe haber turnos Cargados asociados a Tratamientos que tengan Recursos necesarios para su realización.

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Turnos"->"Todos" Datos: <code>_server_/dashboard</code> Nota:	<input checked="" type="checkbox"/>		La información se presenta correctamente
2	En la Web de Turnos Presionar el botón, "Ver Calendario" Datos: <code>_server_/turnos</code> Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
3	Cambiar de vistas presionando en el botón de "Día" Datos: Nota: La vista de los Turnos debería ser consistente con los Turnos, es decir, debería tener la iconografía de los Recursos Necesarios y el Título	<input checked="" type="checkbox"/>	La vista debería presentar un día con sus Turnos.	La información se presenta correctamente
4	Cambiar de vistas presionando en el botón de "Semana" Datos: Nota:	<input checked="" type="checkbox"/>	La vista debería presentar una semana con sus Turnos	La información se presenta correctamente
5	Cambiar de vistas presionando en el botón de "Mes" Datos: Nota:	<input checked="" type="checkbox"/>	La vista debería presentar un mes con sus Turnos	La información se presenta correctamente
6	Cambiar de vistas presionando en el botón de "Día" Datos: Nota:	<input checked="" type="checkbox"/>	La vista debería presentar un día con sus Turnos.	La información se presenta correctamente
7	Con las flechas de desplazamiento, presionar sobre ">" y luego "<" para cambiar de día. Datos: Nota:	<input checked="" type="checkbox"/>	La vista debería presentar el día siguiente, y luego volver al anterior	La información se presenta correctamente
8	Presionar sobre el botón "Hoy" Datos: Nota:	<input checked="" type="checkbox"/>	La vista debería presentar el día de Actual. Con la información e los Turnos	La información se presenta correctamente

Comentario:

Prueba Manual

Diseñada por: Matías Milesi

Asistida Por: Matías Milesi



Caso de Prueba CPA5-05

Probar acceder a una Turno mediante el calendario

Descripción:

Aceptación

Probar la funcionalidad de acceder a un Turno mediante un click en el calendario

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos"->"Todos"</p> <p>Datos: <code>_server_/dashboard</code></p> <p>Nota:</p>		La información se presenta correctamente
2	<p>En la Web de Turnos Presionar el botón, "Ver Calendario"</p> <p>Datos: <code>_server_/turnos</code></p> <p>Nota:</p>	La página debería cargar correctamente	La página carga correctamente
3	<p>Cambiar de vistas presionando en el botón de "Dia"</p> <p>Datos:</p> <p>Nota: La vista de los Turnos debería ser consistente con los Turnos, es decir, debería tener la iconografía de los Recursos Necesarios y el Título</p>	La vista debería presentar un día con sus Turnos	La información se presenta correctamente
4	<p>Hacer click sobre uno de los Turnos</p> <p>Datos:</p> <p>Nota:</p>	El sistema debería Re direccionar al Turno Seleccionado	El sistema re direcciona correctamente

Comentario:

Diseñada por: Nelson Inalbon

Asistida Por: Nelson Inalbon



ABM de Tratamientos

Caso de Prueba CPA5-06

Prioridad: Medio

Fecha de Pueba: 6/2/2020

Prueba de Creación de un Tratamiento

Descripción:

Aceptación

Probar que se puede Crear un Tratamiento

Condiciones Previas:

Debe estar logeado al sistema - Pruebas manuales dependientes - Orden: 1° CPA5-06 - 2° CPA5-07

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Tratamientos"- "Tratamientos" y luego la opción "Nuevo" Datos: _server_/dashboard Nota:	La página debería cargar correctamente	La página carga correctamente
2	Ingresar a la Web de creación de Tratamientos Datos: _server_/tratamientos/new Nota:	La página debería cargar correctamente	La página carga correctamente
3	Cargar datos del Tratamiento Nuevo Datos: Crear un tratamiento con Nombre "Pruebas" y elegir Patología "Estándar". Elegir un "Color" aleatorio. Nota: Los otros datos están testeados en Automatización, así que puede probar la combinación en el momento	Debería permitir crear un nuevo tratamiento	operación permitida
4	Presionar el botón Guardar Datos: Nota:	Redirección al Tratamiento creado.	Redirección exitosa
5	Verificar la creación del Tratamiento Datos: Nota:	La página debería cargar correctamente	datos correctos
6	Verificar que se genere una Sesión "Estándar" con el "Tratamiento" Datos: Nota:		Sesión Creada

Comentario:

Prueba manual

Diseñada por: Víctor Rosset

Asistida Por: Matías Milesi



Caso de Prueba CPA5-07

Prueba de Edición de un Tratamiento

Descripción:

Aceptación

Probar que se puede editar un Tratamiento

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Tratamientos" -> "Tratamientos" y luego la opción "Todos"</p> <p>Datos:</p> <p>Nota:</p>	<p>La página debería cargar correctamente; ruta: "_server_/dashboard"</p>	<p>La página carga correctamente</p>
2	<p>Seleccione el Tratamiento con Nombre "Pruebas", buscándolo en el buscador y presionando "Buscar".</p> <p>Datos:</p> <p>Nota:</p>	<p>La página debería cargar correctamente</p>	<p>La página carga correctamente</p>
3	<p>sobre el registro, presione el botón, "Editar"</p> <p>Datos:</p> <p>Nota:</p>	<p>Debería Redigir a la página _server_/tratamientos/id_tratamiento/edit</p>	<p>Redirección exitosa</p>
4	<p>Editar los campos del Tratamiento.</p> <p>Datos: Cambiarle nombre de "Pruebas" a "Pruebas Test", luego repetir la prueba y volver a colocar nombre "Pruebas"</p> <p>Nota: Para luego correr la prueba de eliminación, colocar la Indicación en "Indicaciones de Pruebas"</p>	<p>Debería permitir editar los campos</p>	<p>Permite</p>
5	<p>Presionar el botón Guardar</p> <p>Datos:</p> <p>Nota:</p>	<p>Redirección al Tratamiento Editado.</p>	<p>Redirección exitosa</p>
6	<p>Verificar la edición del Tratamiento</p> <p>Datos:</p> <p>Nota:</p>	<p>La página debería cargar correctamente y los datos deberían haberse editado</p>	<p>La página carga correctamente</p>

Comentario:

Prueba Manual

Diseñada por: Víctor Rosset

Asistida Por: Matías Milesi



Gestión por Calendario

Caso de Prueba CPA5-08

Prioridad: Medio
Fecha de Pueba: 8/2/2020

Probar el análisis de disponibilidad de Turno según la disponibilidad

Descripción:

Aceptación

Se deberá probar la disponibilidad de los recursos para agregar un turno, según la disponibilidad de recursos en ese horario.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos" -> "Rehabilitaciones" -> "Nueva"</p> <p>Datos: _server_/dashboard</p> <p>Nota: Buscar alguna fecha que esté libre de Recursos.</p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	<p>Completar los datos del Paciente, Kinesiólogo y Patología.</p> <p>Datos: _server_/fichas_rehabilitaciones/new Paciente: "Matias Milesi"; Kinesiólogo: "Mauro"; Patología: "Estandar" y Tratamiento: "Estandar"</p> <p>Nota: Puede variar los datos en caso que considere necesario.</p>	<input checked="" type="checkbox"/>	El sistema debería permitir ingresar los datos	El sistema permite ingresar los datos
3	<p>Presionar sobre el botón "Martes y Jueves"</p> <p>Datos:</p> <p>Nota: Puede variar los días en caso que considere necesario</p>	<input checked="" type="checkbox"/>	Se deberían tildar los checks de "Martes" y "Jueves"	Al presionar el botón la selección ocurre
4	<p>Presionar sobre el botón "Ver Calendario"</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	El calendario debería aparecer	El calendario aparece
5	<p>Colocar el Calendario en vista por "semana" o "día" y presionar sobre una Hora.</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	El sistema debería analizar la disponibilidad del Turno y mostrar mensajes de notificaciones.	Notificaciones mostradas
6	<p>Cerrar el calendario.</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Deberían verse los días y horarios generados automáticamente.	
7	<p>Presionar sobre el botón "Guardar"</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	El Sistema debería Redireccionar a la vista de la Rehabilitación Creada	Redirección
8	<p>Revisar que la Rehabilitación y los turnos se hayan realizado correctamente</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>		Datos correctos

Comentario:

Diseñada por: Matias Milesi

Asistida Por: Víctor Rosset



Probar asignar un turno a pesar de no tener recursos disponibles

Descripción:

Aceptación

Probar asignar un turno a pesar de no tener recursos disponibles, el resultado debería ser exitoso

Condiciones Previas:

Buscar un día saturado de turnos, al menos en camillas.

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Turnos" -> "Rehabilitaciones" -> "Nueva" Datos: <code>_server_/dashboard</code> Nota: Buscar alguna fecha que no tenga Recursos Disponibles, de forma que los mensajes sean de "Atención, recursos no disponibles"	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	Completar los datos del Paciente, Kinesiólogo y Patología. Datos: <code>_server_/fichas-rehabilitaciones/new</code> - Paciente: "Matias Milesi"; Kinesiólogo: "Mauro"; Patología: "Estándar" y Tratamiento: "Estándar" Nota: Puede variar los datos en caso que considere necesario.	<input checked="" type="checkbox"/>		
3	Presionar sobre el botón "Martes y Jueves" Datos: Nota: Puede variar los días en caso que considere necesario	<input checked="" type="checkbox"/>	Se deberían tildar los checks de "Martes" y "Jueves"	
4	Presionar sobre el botón "Ver Calendario" Datos: Nota:	<input checked="" type="checkbox"/>	El Calendario debería Aparecer	
5	Colocar el Calendario en vista por "semana" o "día" y presionar sobre una Hora. Datos: Nota:	<input checked="" type="checkbox"/>	El sistema debería analizar la disponibilidad del Turno y mostrar mensajes de notificaciones	Notificaciones mostradas
6	Cerrar el calendario. Datos: Nota:	<input checked="" type="checkbox"/>	Deberían verse los días y horarios generados automáticamente.	
7	Presionar sobre el botón "Guardar" Datos: Nota:	<input checked="" type="checkbox"/>	El Sistema debería Re direccionar a la vista de la Rehabilitación Creada	Redirección
8	Revisar que la Rehabilitación y los turnos se hayan realizado correctamente Datos: Nota:	<input checked="" type="checkbox"/>		Datos correcto

Comentario:

Diseñada por: Víctor Rosset

Asistida Por: Víctor Rosset



Sprint 6

ABM de Rehabilitaciones

Caso de Prueba CPA6-01

Prioridad: Medio
Fecha de Pueba: 5/3/2020

Prueba de marcado como cobrado una Orden Médica a cobrar

Descripción:

Aceptación

Probar que se puede marcar como cobrada una Orden Médica

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Turnos"->"Ordenes Medicas" y luego la opción "Todas" Datos: _server_/dashboard Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	Busque la Orden médica con indicaciones "Indicaciones de Pruebas", buscándolo en el buscador y presionando "Buscar". Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
3	Presiones sobre el botón marcar como Cobrada. Datos: Nota:	<input checked="" type="checkbox"/>	Debería informar el envío mediante una ventana emergente	La ventana se mostró correctamente
4	Verificar que figure en estado cobrado luego de la acción Datos: Nota:	<input checked="" type="checkbox"/>	Debería estar en estado cobrada	Cambió a estado cobrada.

Comentario:

Prueba manual

Diseñada por: Nelson Inalbon

Asistida Por: Víctor Rosset



Caso de Prueba CPA6-02

Prueba de envío de Orden Médica a cobrar

Descripción:

Aceptación

Probar que se puede enviar a cobrar una Orden Médica

Condiciones Previas:

Debe estar logeado al sistema

Paso N°			Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos"->"Ordenes Medicas" y luego la opción "Todas"</p> <p>Datos: _server_/dashboard</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	<p>Busque la Orden médica con indicaciones "Indicaciones de Pruebas", buscándolo en el buscador y presionando "Buscar".</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
3	<p>Presiones sobre el botón enviar a Cobrar.</p> <p>Datos: Seleccione un arancel aleatorio.</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Debería informar el envío mediante una ventana emergente	La ventana se mostró correctamente
4	<p>Verificar que figure en estado enviado a cobrar, luego de la acción</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Debería estar en estado enviado	Cambió a estado enviado.

Comentario:

Prueba manual

Diseñada por: Nelson Inalbon

Asistida Por: Víctor Rosset

APROBADA?



ABM de Sesiones de Tratamientos

Caso de Prueba CPA6-03

Prioridad: Medio
Fecha de Pueba: 5/3/2020

Prueba de Creación de Sesiones de Tratamientos

Descripción:

Aceptación

Probar que se pueden crear nuevas Sesiones en un Tratamiento asignándole Recursos

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Tratamientos" -> "Tratamientos" -> "Todos" Datos: Nota:	<input checked="" type="checkbox"/>	La página debería cargar: "_server_/dashboard"	La página carga correctamente
2	Seleccionar un Tratamiento aleatoriamente y presionar el botón "Ver" Datos: Nota:	<input checked="" type="checkbox"/>	La página debería cargar: "_server_/tratamientos/1"	Carga correctamente
3	En el Tratamiento, buscar la Sección: "Sesiones" Datos: Nota:	<input checked="" type="checkbox"/>	Debería tener una sección: Sesiones	Tiene la sección
4	Presionar el botón "+Nueva Sesión" e ingresar los datos Datos: Nombre: "Ses Prueba"; Tiempo de Kinesiólogo: "60" Nota:	<input checked="" type="checkbox"/>	Debería aparecer una ventana emergente	Aparece ventana emergente
5	Presionar el botón "Guardar" Datos: Nota:	<input checked="" type="checkbox"/>	El sistema debería ingresar el valor cargado.	Guardado
6	Presionar sobre la Flecha lateral Recién ingresada y desplegar la vista de Recursos asociados a la Sesión. Presionar el botón "Asociar Recurso" Nota:	<input checked="" type="checkbox"/>	Debería permitir la asociación	Permite la asociación
7	Rellenar los datos de prueba Datos: Seleccionar un Recurso y un tiempo aleatorio válido Nota:	<input checked="" type="checkbox"/>	Debería permitir rellenar los valores	Permite ingresar los valores
8	Presionar el botón "Guardar" Datos: Nota:	<input checked="" type="checkbox"/>	Debería recargarse la página.	Recarga correcta
	Verificar los datos creados Datos: Nota:	<input checked="" type="checkbox"/>	El sistema debería mostrar la información ingresada.	La información mostrada es correcta

Comentario:

Diseñada por: Matías Milesi

Asistida Por: Nelson Inalbon



Prueba de Edición de Sesiones de Tratamientos

Descripción:

Aceptación

Probar que se pueden modificar las Sesiones de un Tratamiento y sus Recursos necesarios.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	✓	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Tratamientos" -> "Tratamientos" -> "Todos" Datos: La página debería cargar: "_server_/dashboard" Nota:	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	Seleccionar un Tratamiento aleatoriamente y presionar el botón "Ver" Datos: La página debería cargar: "_server_/tratamientos/1" Nota:	<input checked="" type="checkbox"/>	Deberían mostrarse los tratamientos	Los tratamientos se muestran correctamente
3	En el Tratamiento, buscar la Sección: "Sesiones" Datos: Nota:	<input checked="" type="checkbox"/>	Debería existir una sección con Sesiones	Existe sección
4	Seleccionamos una y Presionar el botón "Editar", modificar los datos Datos: Nombre: "Ses Prueba"; Tiempo de Kinesiólogo: "30" Nota:	<input checked="" type="checkbox"/>	Se debería poder editar los datos	Se pueden editar
5	Presionar el botón "Guardar" Datos: Nota:	<input checked="" type="checkbox"/>	El sistema debería ingresar el valor cargado	Se guardó el valor
6	Presionar sobre la Flecha lateral Reclén Ingresada y desplegar la vista de Recursos asociados a la Sesión. Elegimos un Recurso y presionamos, "Editar" Nota:	<input checked="" type="checkbox"/>	Debería existir una sección de Recursos	Existe
7	Cambiar el Recursos utilizado por otro y el tiempo. Datos: Seleccionar un Recurso y un tiempo aleatorio válido Nota:	<input checked="" type="checkbox"/>	El sistema debería permitir ingresar valores	Permite ingresar valores.
8	Presionar el botón "Guardar" Datos: Nota:	<input checked="" type="checkbox"/>	Debería recargarse la página	Recarga exitosa
9	Verificar los datos editados Datos: Nota:	<input checked="" type="checkbox"/>	El sistema debería mostrar la información editada.	Los datos se ingresaron correctamente.

Comentario:

Diseñada por: Victor Rosset

Asistida Por: Nelson Inalbon



ABM de Rehabilitaciones

Caso de Prueba CPA6-05

Prioridad: Medio
Fecha de Pueba: 6/3/2020

Prueba de asignación de turnos en calendario mediante nueva Rehabilitación

Descripción:

Aceptación

Probar que se pueden asignar turnos mediante la opción del calendario en la generación de una nueva Rehabilitación.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos" -> "Rehabilitaciones" -> "Nueva"</p> <p>Datos: <code>_server /dashboard</code></p> <p>Nota:</p>	La página debería cargar correctamente	La página carga correctamente
2	<p>Completar los datos del Paciente, Kinesiólogo y Patología.</p> <p>Datos: <code>_server/fichas-rehabilitaciones/new</code> - Paciente: "Matías Milesi"; Kinesiólogo: "Mauro"; Patología: "Estándar" y Tratamiento: "Estándar"</p> <p>Nota: Puede variar los datos en caso que considere necesario.</p>	Debería carga la página de nueva Rehabilitación	La página carga correctamente
3	<p>Presionar sobre el botón "Martes y Jueves"</p> <p>Datos:</p> <p>Nota: Puede variar los días en caso que considere necesario</p>	Se deberían lidar los checks de "Martes" y "Jueves"	ok
4	<p>Presionar sobre el botón "Ver Calendario"</p> <p>Datos:</p> <p>Nota:</p>	El Calendario debería Aparecer	Aparece el calendario
5	<p>Colocar el Calendario en vista por "semana" o "día" y presionar sobre una Hora.</p> <p>Datos:</p> <p>Nota:</p>	El sistema debería analizar la disponibilidad del Turno y mostrar mensajes de notificaciones.	Notificaciones mostradas
6	<p>Cerrar el calendario.</p> <p>Datos:</p> <p>Nota:</p>	Deberían verse los días y horarios generados automáticamente.	Se ven los días y horarios generados
7	<p>Presionar sobre el botón "Guardar"</p> <p>Datos:</p> <p>Nota:</p>	El Sistema debería Redireccionar a la vista de la Rehabilitación Creada	Redirección
8	<p>Revisar que la Rehabilitación y los turnos se hayan realizado correctamente</p> <p>Datos:</p> <p>Nota:</p>		Datos correcto

Comentario:

Prueba Manual

Diseñada por: Victor Rosset

Asistida Por: Matías Milesi

APROBADA?

PASS

Caso de Prueba CPA6-06

Prioridad: Medio
Fecha de Pueba: 6/3/2020

Prueba de asignación de turnos mediante horarios cuando generamos una nueva Rehabilitación

Descripción:

Acceptación

Probar que se pueden asignar turnos mediante la opción de Horarios en la generación de una nueva Rehabilitación.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos" -> "Rehabilitaciones" -> "Nueva"</p> <p>Datos: <code>_server_/dashboard</code></p> <p>Nota:</p>	La página debería cargar	La página carga correctamente
2	<p>Completar los datos del Paciente, Kinesiólogo y Patología.</p> <p>Datos: <code>_server_/fichas-rehabilitaciones/new</code> - Paciente: "Matias Milesi"; Kinesiólogo: "Mauro"; Patología: "E estándar" y Tratamiento: "E estándar"</p> <p>Nota: Puede variar los datos en caso que considere necesario.</p>	El sistema debería permitir ingresar los datos	Permite
3	<p>Presionar sobre el botón "Martes y Jueves"</p> <p>Datos:</p> <p>Nota: Puede variar los días en caso que considere necesario</p>	Se deberían tildar los checks de "Martes" y "Jueves"	
4	<p>Presionar sobre el botón "PM" -> "17 hs". Presionar sobre el botón "Analizar"</p> <p>Datos:</p> <p>Nota:</p>	El sistema debería analizar y mostrar las notificaciones de disponibilidad.	Notificaciones mostradas
5	<p>Presionar sobre el botón "Guardar"</p> <p>Datos:</p> <p>Nota:</p>	El Sistema debería Redireccionar a la vista de la Rehabilitación Creada	Redirección
6	<p>Revisar que la Rehabilitación y los turnos se hayan realizado correctamente</p> <p>Datos:</p> <p>Nota:</p>		Datos correcto

Comentario:

Prueba Manual

Diseñada por: Matias Milesi

Asistida Por: Matias Milesi



Sprint 7

ABM Variables de Seguimiento

Caso de Prueba CPA7-01

Prioridad: Medio
Fecha de Pueba: 1/4/2020

Prueba de Creación de Variables de Seguimiento

Descripción:

Aceptación

Probar que se puede generar una nueva Variable de Seguimiento.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Tratamientos" -> "Variables de Seg." -> "Todas" Datos: Nota:	<input checked="" type="checkbox"/>	La página debería cargar: "_server_/dashboard"	La página carga correctamente
2	Seleccionar la opción "+ Crear Nueva" Datos: Nota:	<input checked="" type="checkbox"/>	La página debería cargar: "_server_/variables"	correcto
3	Rellenar los datos de la Variables de Seg. Con los datos de prueba Datos: Nombre: "Prueba"; UND: "seg" Nota:	<input checked="" type="checkbox"/>	La página debería cargar: "_server_/variables/new"	correcto
4	Presionar el botón "Guardar" Datos: Nota:	<input checked="" type="checkbox"/>	Redirección a: "_server_/variables/id_creado"	correcto
5	Verificar que los valores correspondan a los ingresados. Datos: Nota:	<input checked="" type="checkbox"/>	Debería mostrarla información	Valores correctos

Comentario:

Diseñada por: Matias Milesi

Asistida Por: Nelson Inalbon



Caso de Prueba CPA7-02

Prueba de Edición de Variables de Seguimiento

Descripción:

Aceptación

Probar que se pueden Editar Variables de Seguimiento

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Tratamientos" -> "Variables de Seg." -> "Todas"</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	La página debería cargar: "_server_/dashboard"	La página carga correctamente
2	<p>Buscamos "Prueba" en la cuadro de búsqueda y presionamos "Buscar"</p> <p>Datos: La página debería cargar: "_server_/variables"</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Debería permitir buscar los valores	Permite buscar las Variables de seg.
3	<p>Sobre el Registro Buscado, presionamos "Editar"</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Debería permitir hacer click	Click ok
4	<p>Editamos los valores de la Variable de Seguimiento</p> <p>Datos: Cambiar el Nombre de " Prueba" a " PruebaTest", luego repetir la prueba y volver a colocar el nombre "Prueba"</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Debería permitir editar	Edición correcta
5	<p>Presionamos el botón "Guardar"</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Redirección a: "_server_/variables/id_creado"	Redirección exitosa
6	<p>Verificar que los datos hayan sido modificados</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Los datos deberían haber sido modificados	Los datos fueron modificados

Comentario:

Diseñada por: Víctor Rosset

Asistida Por: Nelson Inalbon



Caso de Prueba CPA7-03

Prueba de Eliminación de Variables de Seguimiento

Descripción:

Aceptación

Probar que se puede eliminar una Variable de Seguimiento

Condiciones Previas:

Dependiente del caso de Prueba anterior

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Tratamientos" -> "Variables de Seg." -> "Todas" Datos: Nota:	<input checked="" type="checkbox"/>	La página debería cargar: "_server_/dashboard"	La página carga correctamente
2	Buscamos "Prueba" en la cuadro de búsqueda y presionamos "Buscar" Datos: Nombre: "Prueba" Nota:	<input checked="" type="checkbox"/>	La página debería cargar: "_server_/variables". Debería permitir buscar	Permite la búsqueda
3	Sobre el Registro Buscado, presionamos "Ver" Datos: Eliminar: "Prueba" Nota:	<input checked="" type="checkbox"/>	Debería re direccionar a la página del registro	Redirección exitosa
5	Presionamos el botón "Borrar" Datos: Nota:	<input checked="" type="checkbox"/>	Redirección a: "_server_/variables/rid_creado"	Redirección exitosa
6	Verificar que los datos hayan sido eliminado Datos: Nota:	<input checked="" type="checkbox"/>	Debería haber eliminado el registro	Registro borrado.

Comentario:

Diseñada por: Matías Milesi

Asistida Por: Nelson Inalbon



Informes: Variables de S.

Prioridad: Medio

Caso de Prueba CPA7-04

Fecha de Pueba: 1/4/2020

Prueba de Gráficos de Variables de Tratamientos

Descripción:

Aceptación

Probar que el sistema Grafica el resultado de las Rehabilitaciones según las mediciones ingresadas.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	✓	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Informes" -> "Tratamientos" Datos: Nota:	✓	La página debería cargar. "_server_/dashboard"	La página carga correctamente
2	Seleccionar una Patología y presionar el botón "Buscar Tratamiento" Datos: valor "Estándar" Nota:	✓	Debería permitir la búsqueda	Permite la búsqueda
3	Seleccionar un Tratamiento y presionar el botón "Buscar Rehabilitaciones" Datos: valor "Estándar" Nota:	✓	Debería permitir la búsqueda	Permite la búsqueda
4	Seleccionar una Rehabilitación Datos: Elegir un valor aleatoriamente Nota:	✓	Debería presentar varias Rehabilitaciones	Muestra varias rehabilitaciones
5	Seleccionar una Variable Datos: Nota:	✓	Debería mostrar varias variables	Muestra variables
6	Presionar el botón "Graficar Rehabilitación" Datos: Nota:	✓	El sistema debería graficar la evolución del Tratamiento	El sistema grafica la evolución.

Comentario:

Diseñada por: Nelson Inalbon

Asistida Por: Nelson Inalbon



Sprint 8

Informes de Turnos

Prioridad: Medio

Caso de Prueba CPA8-01

Fecha de Pueba: 17/5/2020

Pruebas de generación de Informe Diario de Turnos

Descripción:

Acceptación

Probar que se puede generar un PDF con todos los turnos del día en Curso.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Turnos"->"Todos" Datos: La página debería cargar: "_server_/dashboard" Nota:	<input checked="" type="checkbox"/>	Debería carga la página	La página carga correctamente
2	Seleccionamos un Kinesiólogo Datos: Nombre: "Mauro" Nota:	<input checked="" type="checkbox"/>	Deberían mostrarse los Kinesiólogos	Los Kinesiólogos se muestran correctamente
3	Presionamos el botón "Generar PDF Diario" Datos: Nota:	<input checked="" type="checkbox"/>	Se debería Descargar un archivo	Se descargó el archivo
4	Revisar Datos Datos: Nota:	<input checked="" type="checkbox"/>	Los datos mostrados son correctos	Contiene la información correcta

Comentario:

Diseñada por: Nelson Inalbon

Asistida Por: Nelson Inalbon



Pruebas de generación de Informe Diario de Turnos de un día específico.

Descripción:

Aceptación

Probar que se puede generar un PDF con todos los turnos de un día en específico.

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción	Completado	Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Turnos"->"Todos" Datos: Nota:	<input checked="" type="checkbox"/>	La página debería cargar: " server /dashboard"	La página carga correctamente
2	Seleccionamos un Kinesiólogo y una fecha aleatoria. Datos: Nombre: "Mauro" Nota:	<input checked="" type="checkbox"/>	El sistema debería permitir ingresar los datos	Permite
3	Presionamos el botón "Generar PDF Diario" Datos: Nota:	<input checked="" type="checkbox"/>	Se debería Descargar un archivo	
4	Revisar Datos Datos: Nota:	<input checked="" type="checkbox"/>	Los datos mostrados son correctos	

Comentario:

Diseñada por: Matias Milesi

Asistida Por: Matias Milesi



Correcciones y Mejoras

Prioridad: Alto

Caso de Prueba CPA8-03

Fecha de Pueba: 17/5/2020

Pruebas de generación de Informes de Cobros Pendientes

Descripción:

Aceptación

Probar la generación de los Cobros Pendientes

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción		Resultado Esperado	Resultado Actual
1	En el menú lateral, seleccionar "Turnos"->"Órdenes Medicas"->"Todas" Datos: Nota:	<input checked="" type="checkbox"/>	Página: _server_/dashboard	La página carga correctamente
2	En la parte inferior del listado, presionar el botón "Cobro Pendientes" Datos: Nota:	<input checked="" type="checkbox"/>	Debería descargarse un archivo con el detalle de los cobros del ultimo mes	Se decargar el archivo
3	Verificar que los datos sean correctos Datos: Nota:	<input checked="" type="checkbox"/>	Los datos deberían ser correctos	Datos correctos

Comentario:

Diseñada por: Matías Milesi

Asistida Por: Matías Milesi



Caso de Prueba CPA8-04

Prioridad: Alto
Fecha de Pueba: 17/5/2020

Pruebas de generación de Informes de Envíos Pendientes

Descripción:

Aceptación

Probar la generación de los Envíos Pendientes

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción		Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos"->"Órdenes Medicas"->"Todas"</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Página: _server_/dashboard	La página carga correctamente
2	<p>En la parte inferior del listado, presionar el botón Envíos Pendientes"</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Debería descargarse un archivo con el detalle de los cobros del ultimo mes.	Se descarga el archivo
3	<p>Verificar que los datos sean correctos</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Los datos deberían ser correctos	Datos correctos

Comentario:

Diseñada por: Matías Milesi

Asistida Por: Matías Milesi



Prioridad: Alto

Caso de Prueba CPA8-05

Fecha de Pueba: 17/5/2020

Pruebas de generación de Informes de Cobros del Último Mes

Descripción:

Aceptación

Probar la generación de los Cobros del Último mes

Condiciones Previas:

Debe estar logeado al sistema

Paso N°	Descripción		Resultado Esperado	Resultado Actual
1	<p>En el menú lateral, seleccionar "Turnos"->"Órdenes Medicas"->"Todas"</p> <p>Datos: Página: _server_/dashboard</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	La página debería cargar correctamente	La página carga correctamente
2	<p>En la parte inferior del listado, presionar el botón "Cobro Ultimo Mes"</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Debería descargarse un archivo con el detalle de los cobros del ultimo mes.	Se descarga el archivo
3	<p>Verificar que los datos sean correctos</p> <p>Datos:</p> <p>Nota:</p>	<input checked="" type="checkbox"/>	Datos correctos	Los datos son correctos

Comentario:

Diseñada por: Matías Milesi

Asistida Por: Nelson Inalbon

