UNIVERSIDAD TECNOLÓGICA NACIONAL

Tecnicatura Universitaria en Programación

Informe de Práctica Profesional Supervisada (PPS)

DOCENTE A CARGO: Lic. ZAPATA ICART ERNESTO

ALUMNO: LENARDON DAMIAN ABRIL

ORGANIZACIÓN: ERTIC S.R.L. / GRANDI Y ASOCIADOS

SITUADA EN PARANA, ENTRE RÍOS

PERÍODO DESDE: 17/10/2023

HASTA: 17/03/2024

Resumen Ejecutivo

El presente informe detalla las actividades y logros alcanzados durante mi Práctica Profesional Supervisada en Grandi y Asociados. El objetivo principal de esta práctica fue aplicar y profundizar los conocimientos adquiridos durante mi formación académica, además de adquirir experiencia práctica en un entorno profesional.

Durante este período, participé en la creación y gestión de Altas, Bajas y Modificaciones (ABM) de diversas entidades del módulo "Estación de Servicio", así como en la elaboración de informes utilizando Microsoft Report Viewer. Estas tareas me permitieron desarrollar habilidades técnicas en el manejo de SQL Server, C#, HTML, CSS, JavaScript y Angular.js, además de aprender a trabajar con arquitecturas de aplicaciones basadas en el modelo MVC.

Uno de los logros más significativos fue la implementación de un informe detallado sobre un listado policial de ocupantes, que abarcó todas las reservas de un hotel y garantizó una gestión eficiente y segura de la información. Este proyecto individual demostró mi capacidad para aplicar conocimientos teóricos a problemas prácticos, así como para trabajar de manera autónoma y eficaz.

La interacción constante con colegas y supervisores enriqueció mi experiencia, permitiéndome mejorar mis habilidades de comunicación y trabajo en equipo. Este período de práctica no solo ha consolidado mi formación académica, sino que también ha preparado el terreno para mi futura carrera profesional en el desarrollo de software.

En conclusión, la práctica profesional supervisada en Grandi y Asociados ha sido una experiencia invaluable que ha fortalecido mi perfil profesional y me ha proporcionado una base sólida para enfrentar futuros desafíos en el campo de la tecnología de la información.

Contenido

Resumen Ejecutivo 1
Contenido 2
Introducción 3
Aspectos Generales 4
Antecedentes 4
Objetivos 5
Generales 5
Objetivos Específicos 5
Delimitación de la Práctica Profesional 6
Limitaciones 6
Evaluación Institucional 7
Descripción General de la Institución 7
Objetivos de la Empresa 8
Visión 8
Misión 8
Valores 9
Actividad empresarial 10
Estructura Organizacional 10
Grupo Humano 11
Escenario de Trabajo 11
Desarrollo de la Práctica Profesional 13
Descripción de Actividades Desarrolladas 13
Capacitación Inicial 13
Tecnologías y herramientas utilizadas 14
Actividades Desarrolladas 15
Conclusiones 42
Aprendizaje obtenido de la realización de la practica 42
Comentarios personales del trabajo realizado 42
Conclusiones generales 43
Recomendaciones 44
Aportes 45
Anexos 46

Introducción

La presente introducción proporciona una visión general del informe de práctica profesional supervisada realizada en Grandi y Asociados durante el período del 17 de Octubre de 2023 al 17 de Marzo de 2024. Grandi y Asociados es una empresa líder en el sector de desarrollo de software e ingeniera web, especializada en soluciones informáticas integrales.

El principal objetivo de esta práctica fue complementar y aplicar los conocimientos teóricos adquiridos durante mi formación académica en un entorno profesional. Además, se buscó obtener experiencia práctica en el campo del desarrollo de software, específicamente en áreas como la gestión de bases de datos, la creación de informes y el desarrollo de aplicaciones web.

Este informe tiene como objetivo documentar de manera detallada las actividades realizadas durante la práctica, así como los resultados obtenidos y los aprendizajes adquiridos. Se incluirá una descripción de las tareas realizadas, los desafíos enfrentados y las soluciones propuestas. Además, se analizará el impacto de la práctica en mi desarrollo personal y profesional.

El mismo está estructurado en varias secciones, que abarcan desde la descripción del contexto y los objetivos de la práctica hasta la presentación de los resultados y conclusiones. Cada sección se centrará en aspectos específicos de la práctica, siguiendo una metodología clara y organizada.

Esta compilación reviste gran importancia tanto para mi desarrollo profesional como para la evaluación de la práctica por parte de los supervisores académicos.

Aspectos Generales

Antecedentes

Mi nombre es Lenardon Damián Abril, originario de la ciudad de Villa Libertador San Martín, ubicada en la provincia de Entre Ríos, Argentina. Actualmente, resido en la ciudad de Paraná y me desempeño como estudiante de programación, donde he dedicado mi tiempo y esfuerzo a adquirir conocimientos y habilidades en este apasionante campo.

Mi interés por la programación surgió durante mis estudios en mi carrera anterior, Bioingeniería, la cual incluía materias relacionadas con el desarrollo de software desde una perspectiva médica. Aunque mi experiencia previa se limitaba a lo aprendido en esa carrera, la pandemia del 2020 y el creciente protagonismo del desarrollo de software en el mundo me llevaron a replantear mi camino profesional.

La posibilidad de trabajar desde cualquier parte del mundo y la diversidad de tareas que ofrece el desarrollo de software son aspectos que me motivan profundamente. Anhelo poder trabajar en un entorno que me desafíe constantemente y me brinde la oportunidad de contribuir a la creación de soluciones innovadoras que faciliten la vida de las personas.

Mis metas profesionales incluyen formar parte de una empresa u organización destacada en el campo del desarrollo de software o desarrollar proyectos que tengan un impacto positivo en la sociedad. Esta experiencia laboral representa para mí la oportunidad de sumergirme en el mundo empresarial y conocer de cerca cómo se desarrollan y ejecutan proyectos en este ámbito dinámico y en constante evolución.

Objetivos

Generales

- Adquirir experiencia práctica en el campo del desarrollo de software.
- Ampliar mis conocimientos y habilidades en programación y desarrollo de aplicaciones.
- Comprender el funcionamiento y la dinámica de trabajo dentro de una empresa de tecnología.
- Desarrollar competencias relacionadas con el trabajo en equipo, la comunicación y la resolución de problemas en un entorno laboral.
- Establecer una base sólida para mi futura carrera profesional en el campo de la tecnología de la información.
- Identificar mis áreas de interés y especialización dentro del desarrollo de software.

Objetivos Específicos

- Participar en proyectos de desarrollo de software bajo la supervisión de profesionales experimentados.
- Dominar el uso de herramientas y tecnologías específicas utilizadas en la empresa.
- Contribuir al desarrollo y mejora de aplicaciones existentes mediante la implementación de nuevas funcionalidades o la resolución de problemas técnicos.
- Mejorar mis habilidades de documentación de código y diseño de aplicaciones.
- Obtener retroalimentación constante de mis supervisores y compañeros para mejorar continuamente mi desempeño.
- Cumplir con los plazos y objetivos establecidos para cada proyecto asignado.
- Participar en reuniones y actividades de capacitación para seguir aprendiendo y creciendo profesionalmente.

Delimitación de la Práctica Profesional

- Mantenimiento y desarrollo de funcionalidades del sistema: Implementación de nuevas funcionalidades en el sistema de la empresa para mejorar la experiencia del cliente.
- Resolución de problemas y corrección de errores para garantizar el funcionamiento óptimo del sistema en todo momento.
- Desarrollo de reportes: Diseño y ejecución de consultas de bases de datos para extraer datos relevantes.
- Creación de informes detallados utilizando archivos RDLC para proporcionar análisis significativos a los clientes.
- Registro y seguimiento de actividades: Utilización de la plataforma Redmine para registrar y documentar cada actividad realizada durante la práctica.
- Seguimiento detallado de las horas dedicadas a cada tarea para garantizar una gestión eficiente del tiempo y recursos.

Limitaciones

- Tiempo Disponible: La práctica tiene una duración específica de 14:00 a 18:00 de lunes a viernes, lo que limita la cantidad de tiempo disponible para dedicar a las actividades relacionadas con la práctica.
- Interferencia con Compromisos Académicos: El horario de la práctica está comprendido muy próximo con el horario de clases en la facultad, lo que puede limitar la disponibilidad de tiempo y afectar la capacidad para completar ciertas tareas dentro del horario establecido.
- Acceso Limitado a Recursos: Al realizar la práctica de manera remota con tu propia computadora, es posible que te enfrentes a limitaciones en cuanto a los recursos disponibles, como software específico o acceso a servidores o bases de datos corporativas.

Evaluación Institucional

Descripción General de la Institución

En su sitio web, la empresa Grandi y Asociados se auto reconoce con los siguientes conceptos:

"Somos una Empresa Desarrolladora de Software e Ingeniería Web en Argentina, con más de 35 años de trayectoria brindando soluciones informáticas integrales en la república y países Latinoamericanos. La experiencia adquirida al haber trabajado en diversidad de rubros sumada al equipo de profesionales de la informática y el diseño web que formamos parte, son los que avalan y garantizan nuestros trabajos, brindando la seguridad y confianza que usted necesita.

Los sistemas de gestión que ofrecemos son desarrollados íntegramente por personal de la empresa, lo que asegura la continuidad y soporte post venta. Son sistemas abiertos y configurables, lo que permite una amplia adaptación.

El desarrollo de sitios web está a cargo de profesionales de la informática y diseñadores gráficos, los que en forma conjunta trabajan para lograr sitios web estéticamente atractivos y técnicamente operativos, conjunción fundamental a tener en cuenta al momento de evaluar el diseño".

Objetivos de la Empresa

Garantizar la seguridad y confianza a sus clientes basados en:

- Sostenibilidad: Centrando sus esfuerzos en elevar la calidad de los productos, renovando las habilidades profesionales, siendo íntegros y responsables en su trabajo.
- Innovación: Siendo parte y contribuyendo a la creación de un entorno empresarial innovador que impulsa el desarrollo y estimula el crecimiento regional.
- Crecimiento y competitividad: En un mercado competitivo en el que es imprescindible diferenciarse, Grandi & Asociados ofrece el respaldo tecnológico necesario para poder concentrarse en el crecimiento de sus clientes.

Visión

"Ser la empresa de desarrollo de Software personalizado, para la pequeña y mediana empresa, de más éxito y con los mejores estándares de calidad en el mercado; ayudando con esto a que nuestros clientes sean más rentables y competitivos".

Misión

"Brindar soluciones tecnológicas al alcance de nuestros clientes a un costo accesible y de alta calidad. Tenemos como misión especial exceder constantemente, con nuestros productos y servicios, todas las expectativas de nuestros clientes, satisfaciendo sus necesidades y ayudándolos en el logro de sus metas y objetivos; ofreciéndoles soluciones integrales, que abarquen los procesos de negocios dentro de su empresa, ayudando a lograr las diferencias competitivas que les permitan perpetuarse y crecer. Esto lo hacemos con personal calificado,

satisfecho y comprometido con la empresa. Presentando a nuestros socios las utilidades esperadas y proporcionando a los empleados posibilidades de desarrollo que les permitan alcanzar crecimiento personal y profesional".

Valores

Integridad: *Ser honestos, justos y coherentes en todas nuestras acciones.*

Responsabilidad: *Asumir compromisos y cumplirlos.*

Compromiso: *Esfuerzo continuo para elevar la calidad de nuestro trabajo.*

Espíritu de Equipo: *Conservar un ambiente de trabajo confortable que inspire el trabajo en equipo y fomente las relaciones humanas.*

Respeto Mutuo: *Mantener el respeto y cordialidad tanto entre los integrantes de la empresa como con los clientes y asociados.*

Superación Continua: *Permanecer renovando nuestras habilidades profesionales.*

Vocación de Servicio: *Predecir y exceder, oportunamente, las expectativas de nuestros clientes.*

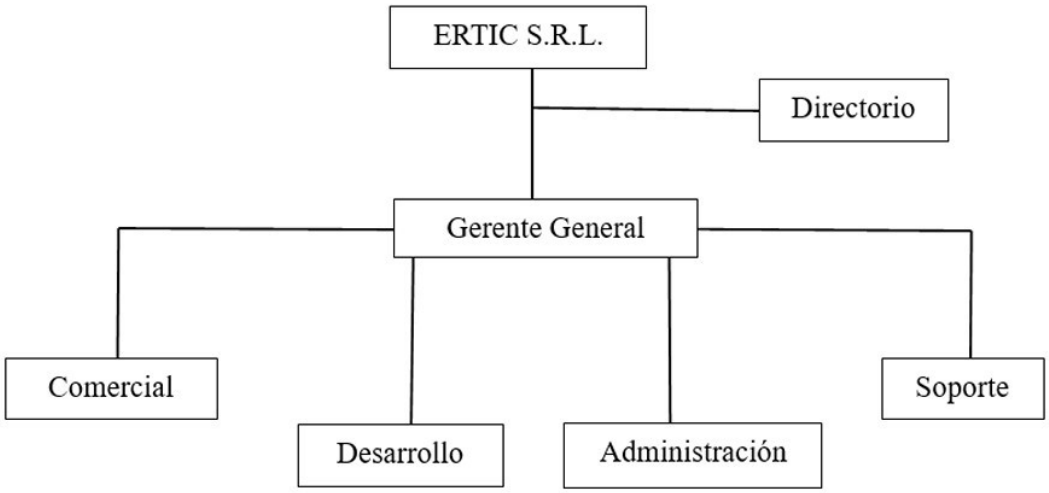
Actividad empresarial

La empresa se especializa en el desarrollo de software para la gestión integral de diversas organizaciones. Por un lado, cuenta con una aplicación altamente compleja que se adapta de manera personalizada a las necesidades y requisitos específicos de cada cliente. Este enfoque personalizado se refleja en la asignación de accesos y permisos únicos para cada tipo de usuario, lo que determina qué interfaces pueden visualizar como así también la información necesaria de cada rubro. Por otra parte, la empresa, brinda servicios de diseño y creación de sitios web y/o tienda e-commerce.

Entre sus principales clientes se destacan empresas de gestión de cobranza, hoteles, restaurantes, delivery, heladerías, supermercados, venta de indumentaria, venta de equipos, playas de estacionamiento, estaciones de servicios, consultorios médicos y centros de estética.

Estructura Organizacional

Estructura Organizacional



Grupo Humano

Durante el transcurso de la Práctica Profesional Supervisada en la firma Grandi y Asociados, fuimos parte de un equipo integrado por cinco alumnos de la carrera, distinguido por su excelencia interpersonal y su enfoque colaborativo en la ejecución de las tareas asignadas. Además, contamos con el respaldo y la disposición de los profesionales de la empresa en el abordaje de desafíos relacionados con la configuración de la aplicación y el desarrollo de código. Asimismo, recibimos la orientación de una figura notablemente dedicada, quien nos asistió tanto en aspectos personales como administrativos. Por otro lado, bajo la tutela directa de uno de los socios de la firma, fuimos instruidos en la ejecución de las solicitudes pertinentes dentro del ámbito laboral.

Escenario de Trabajo

La Práctica Profesional Supervisada se desarrolló de forma virtual, utilizando la plataforma Discord para la comunicación entre los cinco integrantes del grupo y con el resto del equipo que conforma Grandi y Asociados. En la primera etapa de capacitación, cada integrante abordaba los temas de manera individual, según sus intereses y necesidades de aprendizaje. Posteriormente, en la segunda etapa, las actividades prácticas se desarrollaron mayoritariamente de forma colaborativa y grupal. En este contexto, cada nueva tarea asignada comenzaba con una reunión informativa y explicativa detallada sobre el modelo de negocio pertinente por parte de los distintos colaboradores de la empresa. Se abordaba el funcionamiento deseado y se proporcionaba una descripción exhaustiva de las tablas de la base de datos relevantes, con el propósito de facilitar una comprensión profunda del flujo de la información dentro del sistema.

A partir de toda esta información el equipo de trabajo entraba en la etapa de ejecución de las diferentes tareas para lograr el cumplimiento óptimo y eficiente de los requerimientos. Estas actividades incluían el análisis de las solicitudes, el desarrollo del código y la resolución de problemas que pudieran surgir en el proceso.

Cada practicante poseía una rama de trabajo y al finalizar la jornada, se subían los cambios realizados en el repositorio de GIT en SourceTree. Al concluir cada petición, se realizaba un merge a la rama "máster-pasantía" con la previa aprobación y autorización de alguno de los programadores más experimentados de la empresa.

Desarrollo de la Práctica Profesional

Descripción de Actividades Desarrolladas

Capacitación Inicial

Al comenzar la práctica profesional, se realizó una reunión remota con Lucas Grandi, dueño y responsable de la empresa, durante la cual se establecieron las pautas generales de trabajo. En esta reunión inicial, se nos presentó una visión preliminar de varios aplicativos desarrollados por la empresa, lo que nos permitió obtener una comprensión básica del entorno de desarrollo. Asimismo, se nos proporcionó una guía detallada de los temas y conocimientos necesarios para llevar a cabo nuestras tareas con eficiencia.

A lo largo de las siguientes 98 horas, mis actividades se centraron en tareas de investigación, con el objetivo de familiarizarme con las tecnologías empleadas en la empresa. Esta investigación incluyó el estudio de material de lectura y audiovisual proporcionado tanto por Grandi & Asociados como por fuentes oficiales en la red, garantizando así un nivel de comprensión equiparable al de mis compañeros.

Adicionalmente, durante esta etapa inicial, dediqué 8 horas a la instalación y configuración de mi entorno de trabajo. Este proceso incluyó la descarga, instalación y configuración de diversas herramientas, que se detallarán más adelante. También configuré un repositorio contenedor del proyecto y una base de datos local, fundamentales para probar los desarrollos realizados durante la práctica.

Tecnologías y herramientas utilizadas

Las herramientas utilizadas durante mi práctica profesional fueron diversas, abarcando desde la plataforma empleada para las reuniones hasta las herramientas de gestión de versiones de los desarrollos realizados, incluyendo, por supuesto, aquellas específicamente utilizadas para cumplir con las peticiones asignadas. A continuación, se presenta una breve descripción de cada una de ellas:

- **Discord:** Esta herramienta de comunicación por voz, video y texto fue fundamental para la coordinación diaria. Estábamos incluidos en un canal denominado "Entrenamiento", que disponía de una sala de videoconferencia a la cual debíamos ingresar todos los días durante nuestras actividades, así como un chat donde podíamos realizar consultas a los demás miembros de la empresa y compartir información entre los practicantes.
- **Redmine:** Redmine es una aplicación web de gestión de proyectos en la que se registraron todas las actividades realizadas durante la práctica, junto con el tiempo invertido en cada una y una clasificación según su tipo (investigación, reunión, capacitación, desarrollo, etc.).
- **SourceTree:** Esta aplicación facilita la interacción con repositorios GIT mediante una interfaz gráfica amigable. A través de SourceTree, compartíamos los avances en las distintas peticiones en desarrollo.
- **Microsoft SQL Server 2019:** Este sistema de administración de datos es eficaz y confiable, proporcionando un almacén de datos completo y seguro para sitios web ligeros y aplicaciones de escritorio.
- **Microsoft SQL Server Management Studio 19.1:** Este entorno integrado permite administrar cualquier infraestructura de SQL. Proporciona herramientas para configurar, supervisar y gestionar instancias de SQL Server y bases de datos.

- **IDE Microsoft Visual Studio 2022:** Este entorno de desarrollo integrado se utilizó para la mayoría de las actividades de desarrollo. Ofrece herramientas como editor de código fuente, compilación automática y depuración.
- **IDE Microsoft Visual Studio 2015:** Esta versión específica del IDE se empleó exclusivamente para diseñar y modificar archivos de tipo RDLC, utilizados para generar los distintos reportes solicitados.
- **Entity Framework:** Este asignador de relaciones de objetos permite crear una capa de acceso a datos limpia, portátil y de alto nivel con .NET en una variedad de bases de datos. Soporta consultas LINQ, seguimiento de cambios, actualizaciones y migraciones de esquemas.
- **AngularJS:** Este framework de diseño y plataforma de desarrollo permite crear aplicaciones de una sola página eficientes y sofisticadas.

Actividades Desarrolladas

Durante nuestra práctica profesional, llevamos a cabo un total de cinco actividades, las cuales se gestionaban mediante peticiones que especificaban el tipo de tarea (Desarrollo, Tarea genérica o Reportería), un número identificatorio y una descripción detallada. Las primeras cuatro peticiones fueron desarrolladas de manera conjunta por todo el equipo, mientras que la última fue asignada de forma individual, con una petición distinta para cada practicante.

Antes de proceder a la descripción de las actividades realizadas, es fundamental comprender la estructura del sistema de Grandi y Asociados para facilitar la comprensión de ciertos conceptos y referencias a archivos que se mencionarán posteriormente. El sistema se centra en dos proyectos principales: ProyectoCapas y WebGestionComercial.

En el ProyectoCapas encontramos cuatro clases de archivos bien organizados y diferenciados:

- **Lógica de Negocio:** en esta sección se encuentran todos los archivos de clases de entidades utilizados para representar la base de datos dentro del sistema. Cada entidad tiene dos clases asociadas: *entidad.cs,* que detalla todas las propiedades públicas de la entidad, así como los métodos internos y privados utilizados para definir su comportamiento; y *entidadAdmin.cs,* que contiene todos los métodos públicos que brindan control y acceso a los métodos de Alta, Baja y Modificación definidos de forma privada en el archivo anterior.

- **Mapeo:** Aquí se encuentran los archivos que mapean la estructura de la base de datos en el sistema. Estos archivos establecen la correlación entre las entidades de la base de datos y clases del sistema, así como todas las relaciones existentes entre las distintas tablas.

- **DTO (Data Transfer Object):** Los archivos ubicados en esta sección definen los objetos que llevan datos a través de las distintas capaz del sistema como la de presentación, la de negocio y la de acceso a datos.

- **AplicacionWeb.Core:** Contiene los archivos .cshtml de cada una de las vistas utilizadas en el sistema (views), así como los archivos .js asociados a cada una de ellas (Scripts). En estos archivos se definen e implementan comportamientos y llamadas a los distintos Web Services utilizados.

En el proyecto WebGestionComercial encontramos dos secciones bien definidas:

- **Api:** Contiene dos Web Services por entidad. Uno de ellos realiza peticiones a la base de datos utilizando consultas en lenguaje LINQ. LINQ es un lenguaje utilizado en la integración de capacidades de consulta directamente en el lenguaje C#. El otro Web Service

interactúa con los métodos públicos de la entidad para llevar a cabo las operaciones de Alta, Baja y Modificación (ABM).

- **Controllers:** Aquí se encuentran los controladores MVC (Modelo Vista Controlador), los cuales son responsables de interactuar con la capa de presentación mostrando las distintas vistas del sistema. Los controladores manejan la lógica de la aplicación y actúan como intermediarios entre las vistas y los modelos de datos.

Además, en este proyecto también se encuentran vinculadas las carpetas de Views y Scripts del ProyectoCapas. Esta estructura organizada facilita el desarrollo, mantenimiento y comprensión del sistema, ya que separa claramente las diferentes capas y responsabilidades dentro del proyecto.

A continuación, se detallará cada una de las cinco actividades realizadas durante el período de práctica profesional.

Petición #1902

La primera petición constó de dos partes. En primer lugar, se nos solicitó lo siguiente:

En el ABM de comprobantes, cambiar las etiquetas en el comportamiento 'Comprobante de recepción de mercadería app' por 'Aparece en emisión app' y 'Aparece en recepción de mercadería app' por 'Aparece en relacionados app'.

Para comprender mejor esta tarea, es importante destacar que la mayoría de las etiquetas HTML utilizadas en las distintas vistas del sistema que Grandi y Asociados ofrece a sus clientes están almacenadas en archivos con extensión .resx, también conocidos como

archivos de recursos. Estos archivos se utilizan para almacenar datos estáticos que pueden recuperarse en tiempo de ejecución, como los elementos HTML tipo *label*, y son útiles para almacenar cadenas de texto en diferentes idiomas, adaptándose a la configuración regional del usuario.

En el archivo de recursos utilizado en este sistema, las etiquetas están identificadas con un nombre, un valor (el cual se visualiza en la pantalla del cliente) y un comentario. El sistema de Grandi y Asociados cuenta con múltiples vistas separadas por los distintos rubros mencionados anteriormente en este informe. Cada cliente y cada usuario tiene acceso a determinadas vistas según sus permisos, lo que permite que este sistema sea flexible y se ajuste a las necesidades individuales.

Para llevar a cabo esta tarea, primero tuvimos que navegar por el sistema hacia el ABM de comprobantes y desplegar un acordeón de Comportamiento para identificar el archivo .cshtml donde se utilizaban las etiquetas visualizadas como “Comprobante de recepción de mercadería app” y “Aparece en recepción de mercadería app”. Una vez identificado dicho archivo, pudimos conocer el nombre asociado a esas etiquetas y, de esta forma, cambiar su valor por el requerido desde el archivo de recursos correspondiente.

Nombre	Valor	Comentario
ApareceEnRecepcionMercaderiaApp	Aparece en relacionados app	Etiqueta

Imagen 1. Cambio de valor en etiqueta

Nombre	Valor	Comentario
EsComprobanteRecepcionMercaderia	Aparece en emisión app	Etiqueta

Imagen 2. Cambio de valor en etiqueta

Imagen 3. Acordeón Comportamiento dentro del ABM Tipo de Operación

Las Imágenes 1 y 2 muestran el archivo de recursos donde modificamos las etiquetas mencionadas, y la Imagen 3 muestra la pantalla del ABM Comprobantes donde se utilizan.

La segunda parte de esta petición consistió en agregar un elemento *input* de tipo texto en el ABM de consultorios médicos que hiciera referencia a las sucursales. Este elemento debía tener una característica particular: al escribir tres caracteres, se despliega una lista de todas las sucursales asociadas a esos caracteres, permitiendo la selección de una de ellas.

Este tipo de comportamiento, conocido como *suggest*, se utiliza frecuentemente en el sistema para reemplazar de manera efectiva al C*omboBox* tradicional, especialmente en casos donde las opciones a visualizar pueden ser numerosas. Este enfoque mejora la usabilidad del sistema al actuar como un filtro amigable para los usuarios.

Para resolver esta parte de la petición, primero agregamos los elementos *label* e *input* de tipo texto en el archivo .cshtml utilizado para gestionar el alta o la edición de consultorios médicos. Además, se emplearon algunas variables para que el archivo JavaScript asociado a esta vista pudiera determinar en tiempo de ejecución si se trataba de un alta o una modificación de consultorios.

En el caso de una modificación, el formulario se carga con los datos previamente asociados al consultorio seleccionado. En caso de un alta, los campos se presentan con leyendas orientativas para el usuario, utilizando *placeholders*.

```
<div class="grupo-form">
    <label for="sucursal" class="etiqueta-input">@Etiquetas.GetString("Sucursal")</label>
    <div class="contenedor-textbox" ng-show="vm.nuevo || vm.editandoSucursal">
        <input id="sucursal"
               autocomplete="off"
               name="sucursal"
               type="text"
               ng-model="vm.sucursal"
               placeholder="@Etiquetas.GetString("IngreseSucursal")"
               typeahead-min-length="3"
               typeahead-wait-ms="500"
               typeahead-on-select="vm.seleccionoSucursal($item, $model, $label, $event)"
               typeahead="sucursal.label as sucursal.label for sucursal in vm.getSucursales($viewValue) | sucursal"
               class="input-medio">
    </div>
```

Imagen 4. Fragmento de código HTML referido al campo sucursales

El siguiente paso fue dotar de funcionalidad a este campo, lo cual se logró mediante la llamada a un Web Service existente en el sistema, específicamente al SucursalEmpresaConsultasController, a través del método traerSucursalesSuggest. Este Web Service es el encargado de obtener la información relacionada con las sucursales disponibles que coinciden con los caracteres ingresados por el usuario.

```
    getSucursales: function (hint) {
        return this._api.traerSucursalesSuggest(hint, 0, 0)
            .then(function (resultado) {
                return resultado;
            }, function (error) {
                this.mensajesError = errores;
            }.bind(this));
    },
```

Imagen 5. Método getSucursales()

Desde el archivo *NuevoEditarConsultorioMedico.cshtml*, se realiza la llamada al método *getSucursales()* como se observa en la última línea, antes del cierre del de la Imagen 4. Este método está definido en el archivo JavaScript *consultoriomedico-nuevo-editar-vm.js*,

donde se efectúa la llamada al método mencionado anteriormente, tal como se muestra en la Imagen 5.

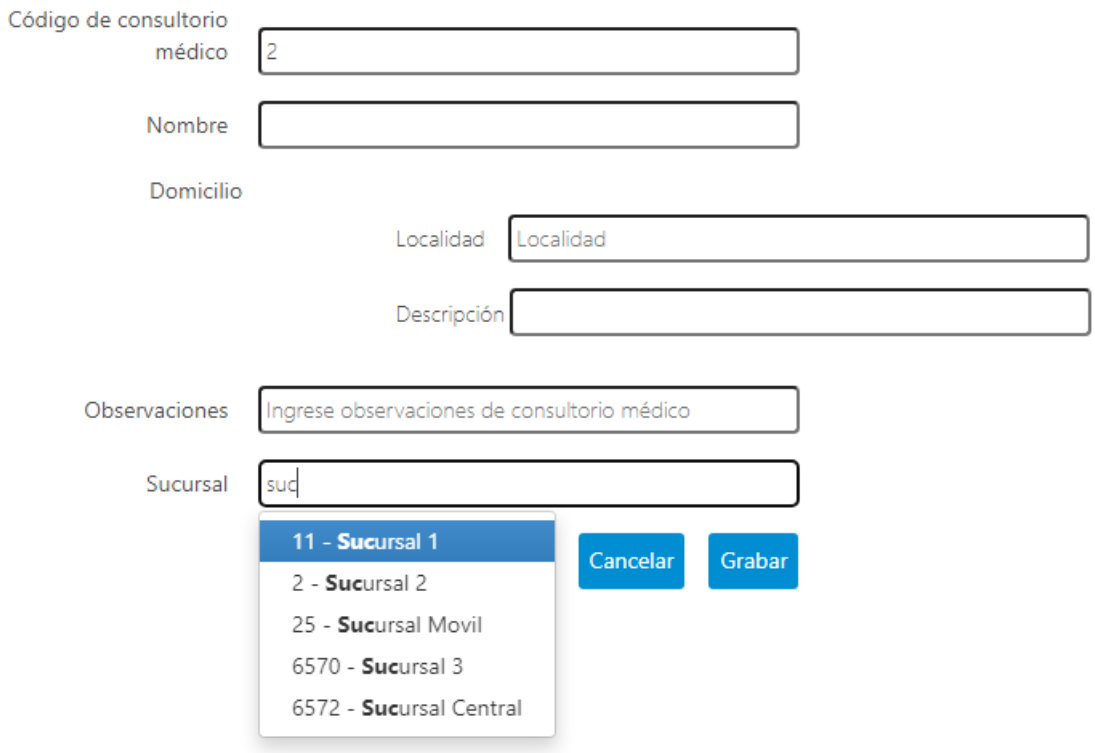


Imagen 6. Formulario de Alta y Modificación de consultorios

La última parte de esta petición implicó la modificación de la estructura de la tabla utilizada para los consultorios médicos en la base de datos, así como la clase de entidad relacionada con los consultorios, el archivo de mapeo asociado y el archivo de transferencia de datos (DTO).

Para llevar a cabo la modificación en la base de datos, se nos proporcionó un script que ejecutamos en nuestras bases de datos locales. Este script agregó el campo "código de sucursal" a la tabla de consultorios como una clave foránea que hace referencia a la tabla de sucursales. Es importante destacar que todas las modificaciones en la base de datos estuvieron a cargo de Lucas Grandi.

Además, se añadió la propiedad "codigoSucursal" en la clase de entidad *ConsultorioMedico.cs*, la cual hace referencia a los consultorios dentro de la lógica de negocio del proyecto. Esta misma propiedad también se incluyó en el archivo *ConsultorioMedicoDTO.cs*, que se utiliza para la transferencia de datos entre las distintas capas del sistema.

En el archivo *consultorioMedicoMap.cs*, se agregó la relación correspondiente en Entity Framework para poder mapear correctamente los cambios realizados en la base de datos al sistema.

Petición #1921

Para cumplir con la solicitud recibida, llevamos a cabo el desarrollo de cinco módulos de Alta, Baja y Modificación (ABM) para un sistema de estaciones de servicio. Estos módulos corresponden a las entidades de Islas, Surtidores, Modelos de Surtidores, Tanques y Mangueras. Previamente a esta etapa, participamos en una reunión de dos horas durante la cual se nos proporcionó una explicación detallada del funcionamiento requerido para el sistema, así como las reglas de negocio específicas pertinentes a este contexto.

Antes de iniciar este desarrollo, realizamos una investigación sobre la estructura de las clases dentro del proyecto y el flujo de información entre las diferentes capas. Esto nos permitió comprender mejor la estructura del proyecto mencionada anteriormente y crear nuestros archivos basados en los existentes para las otras entidades utilizadas en el sistema.

Nuestro primer paso fue la creación de las tablas necesarias en la base de datos para estas cuatro entidades. El script para la creación de estas tablas fue facilitado por Lucas Grandi. Una vez que las tablas estuvieron establecidas, procedimos a la creación de las clases

correspondientes, conforme a las directrices discutidas durante la reunión inicial. Esto implicó la generación de archivos individuales para cada entidad, tales como *Isla.cs*, *IslasAdmin.cs*, *IslaMap.cs* e *IslasDTO.cs*, destinados a la estructura, mapeo y DTO de la entidad Isla, replicando este proceso para las demás entidades.

Posteriormente, procedimos a la creación de los archivos script y views utilizados para cada entidad, los cuales fueron nombrados como *Isla-administrar-vm.js*, *Isla-nuevo-editar-vm.js* e *Isla-mod.js*, por un lado, y *AdministrarIsla.cshtml* y *NuevoEditarIsla.cshtml*, por otro.

Para comprender mejor el propósito de estos archivos, es necesario entender cómo funciona el sistema. Después de que el usuario ingresa a través de la pantalla de inicio de sesión con su nombre de usuario y contraseña, se encuentra con un menú lateral que presenta diversas opciones, como Sistema, Stock, Ventas, y en este caso Estaciones. Dentro de este último ítem, como se puede ver en la Imagen 7, hay un submenú que permite seleccionar cada una de las entidades que hemos desarrollado.

Imagen 7. Menú y submenú de estaciones

Una vez que se accede a uno de los elementos de este submenú, se llega a la primera de las dos vistas asociadas, que es la de Administrar Islas en este caso específico (ver Imagen 8).

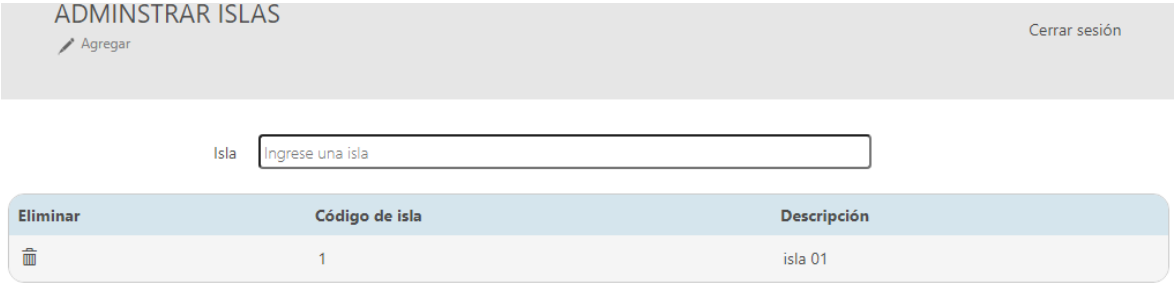


Imagen 8. Vista Administrar Islas

En esta vista se muestra una grilla con todas las islas disponibles y un buscador de islas para agilizar el proceso en caso de que la grilla sea muy extensa. Cuando se selecciona una isla existente o se opta por crear una nueva seleccionando la opción "agregar", se accede a la segunda vista.

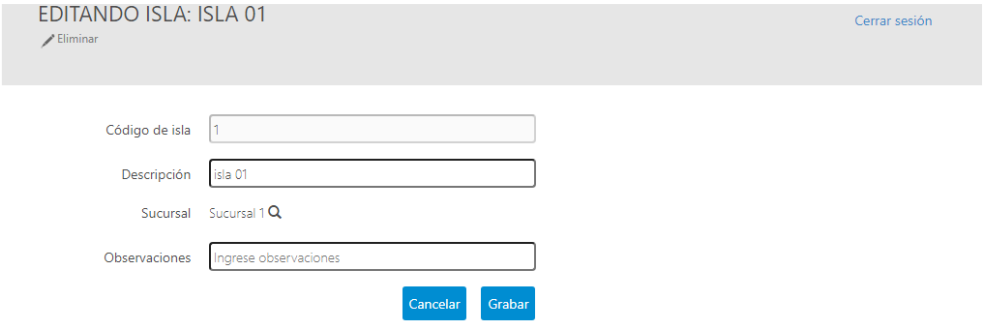


Imagen 9. Vista Nuevo Editar Isla

La segunda vista a la que el usuario puede acceder es la de "nuevo-editar" (ver Imagen 9), donde se pueden cargar los datos de una Isla nueva o modificar los datos de una isla existente. Este es el motivo por el cual tenemos dos archivos con extensiones .cshtml y dos archivos .js con una nomenclatura descriptiva de estos comportamientos.

Mencioné un tercer archivo de extensión .js llamado isla-mod. En este archivo se definen distintos métodos de control para cada una de las dos vistas, además de proveer objetos y filtros utilizados en las vistas para gestionar información de manera más eficiente.

Tanto el buscador implementado en la vista de administrar como el campo de sucursal en la creación o edición tienen el mismo funcionamiento del "*suggest*" de sucursales desarrollado en la solicitud anterior. De esta manera, y de forma progresiva, fuimos incorporando y utilizando de manera constante los conocimientos del sistema que íbamos adquiriendo con el paso de los días y las solicitudes.

El siguiente paso en nuestra petición fue la creación de los archivos de Web Service en la carpeta API del proyecto WebGestionComercial, como mencioné al comienzo de esta sección. Dichos archivos se denominan *IslaController.cs* (ver Imagen 10) e *IslaConsultasController.cs*, además del archivo controlador MVC llamado *EstacionIslaController.cs* (ver Imagen 11), cuyas funcionalidades ya fueron explicadas.

```
[AutorizarApi]
[AutorizarWebApi(CodigoPrograma = Programa.Codigos.ISLA, CodigoAccion = Accion.Codigos.ALTA)]
public void Alta(IslaNuevoDTO datos)
{
    _IslaAdmin.AltaIsla(datos);
}
```

Imagen 10. Fragmento de la clase IslaController.cs

```
[AutorizarMvc(CodigoPrograma = Programa.Codigos.ISLA, CodigoAccion = Accion.Codigos.ALTA)]
public ActionResult NuevoIsla()
{
    var modelo = new EtiquetasModel();
    return View("NuevoEditarIsla",modelo);
}
```

Imagen 11. Fragmento de la clase EstacionIslaController.cs

Finalmente, para concluir nuestra petición y poder utilizar el desarrollo en el sistema, tuvimos que añadir los nuevos elementos de acceso al menú lateral, como se muestra en la Imagen 7, y configurar los permisos necesarios para visualizarlos. Cada elemento tiene un código único dentro del sistema, el cual se define en el archivo *menu-lateral.js*, como se muestra en la Imagen 12.

```
//Estaciones
permisosEstaciones: "ESTACIONES",
permisosMangueras:  "MANGUERAS_",
permisosIslas:  "ISLAS_____",
permisosSurtidor:  "SURTIDOR__",
permisosModelosSurtidor:  "MODELOSURT",
permisosTanque:"TANQUE____",
permisosVarillaje: "VARILLAJE_",
permisosLanzadorEstaciones: "LANZESTACI",
```

Imagen 12. Fragmento de código de menú-lateral.js

Además, en este archivo se especifican las rutas URL donde se mostrarán las vistas manejadas por los controladores de la Aplicación. Esto se evidencia en la Imagen 13, donde se puede observar el fragmento de código del archivo *menu-lateral.js* en el cual se define dicha ruta junto a la barra del navegador web cargando la vista de administrar Islas.

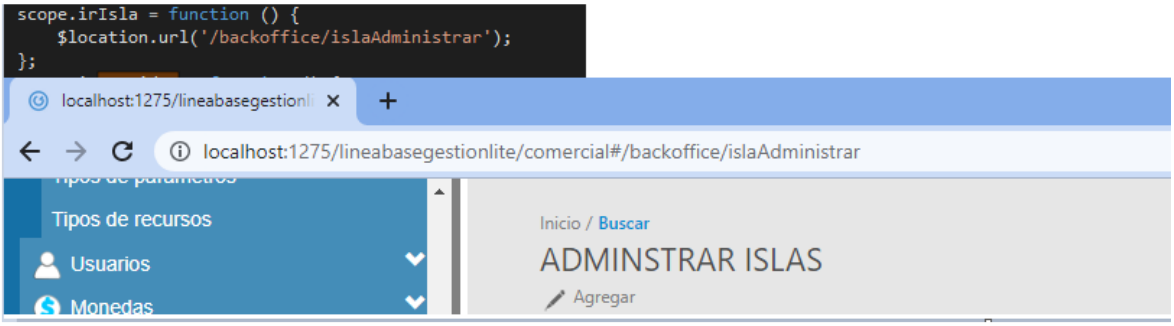


Imagen 13. URL para vista Administrar Islas

El proceso detallado en la explicación de esta petición para el desarrollo de ABM de Islas fue similar al que empleamos para crear los ABM de Surtidor, Modelo de Surtidor, Tanques y Manguera. Cada uno de estos componentes contaba con sus propios campos específicos. En algunos casos, aprovechamos Web Services existentes, como el utilizado en el *suggest* de artículos que se encuentra en el archivo *manguera-nuevo-editar.vm.js*. En otros casos, creamos nuevos Web Services para satisfacer las necesidades de *suggest* de las entidades recién creadas. Un ejemplo de esto es el *suggest* de Surtidor, implementado en el mismo

archivo. Sin embargo, profundizaré en este proceso con más detalle en la explicación de la próxima petición.

Petición #1924

El nuevo desarrollo que nos propusieron incluía agregar la entidad Varillajes al módulo de estaciones de servicio. Esta entidad debía estar relacionada tanto con la entidad de Tanques como con un parte diario de caja.

Los varillajes son mediciones que se realizan a los tanques de combustible para conocer la cantidad de litros que contienen. Es importante destacar que esta medición debe estar asociada a la caja que está abierta en el momento que se realiza, ya que una vez cerrado el parte diario de caja, no se puede editar el varillaje correspondiente. Esta fue la primera restricción que debimos considerar al implementar el ABM solicitado.

La otra restricción con la que nos encontramos fue que, en la información de un varillaje, debíamos registrar tanto el tanque asociado como el tipo de combustible contenido en ese momento. Sin embargo, este último campo no puede modificarse de forma independiente. Si se modifica el tanque asociado, el tipo de combustible se actualiza automáticamente. De esta manera se evita la carga de información incorrecta en cuanto a estos campos.

Estas no fueron las únicas funcionalidades solicitadas en esta petición. También se nos pidió que, en la vista de Administrar Varillajes, aplicáramos una paginación de 20 elementos como máximo por página y que agregáramos un acordeón de filtros para buscar varillajes por rango de fecha, por parte de caja y/o por usuario registrador del varillaje. Además, en la vista de editar tanques debíamos agregar una pestaña donde se mostrarán todos los varillajes existentes asociados al tanque seleccionado.

Para poder cumplir con todas las tareas asignadas, al igual que en las anteriores, tuvimos que realizar un estudio exhaustivo e investigar el sistema en diferentes módulos donde se encontraban comportamientos similares. Cada nueva petición representaba un desafío mayor que el anterior.

El proceso de creación de los archivos utilizados para este ABM es similar al explicado en la petición anterior, por lo que no profundizaré demasiado en su explicación, excepto por los aspectos que diferencien este ABM de los anteriores.

El script para la creación de la tabla varillajes fue proporcionado por Lucas Grandi. A diferencia de otras tablas, en esta ocasión el ID se genera automáticamente en la base de datos y no puede ser modificado en el sistema.

Se crearon los archivos *Varillaje.cs* y *VarillajeAdmin.cs* dentro de la lógica de negocio, *VarillajesMap.cs* para el mapeo, *VarillajesDTO.cs* para el DTO, *AdministrarVarillaje.cshtml* y *NuevoEditarVarillaje.cshtml* para las Views, y *Varillaje-administrar-vm.js*, *Varillaje-mod.js*, y *Varillaje-nuevo-editar-vm.js* para los Scripts. Todos estos archivos se encuentran dentro de la estructura del ProyectoCapas.

De igual manera se crearon los archivos *EstacionVarillajeController.cs* para los controladores MVC, y *VarillajeController.cs* y *VarillajeConsultasController.cs* para los Web Services de esta entidad y se agregaron los permisos de acceso y las URL asociadas para utilizar este desarrollo en el archivo menu-lateral.js. Estos últimos archivos se encuentran dentro de la estructura del proyecto WebGestionComercial.

Al analizar el archivo *NuevoEditarVarillaje.cshtml* notamos algunas diferencias con respecto a las entidades anteriores. En esta vista (ver Imagen 14) encontramos elementos para registrar la fecha utilizando un *DateTimePicker*, un *ComboBox* para seleccionar el parte de

caja diario asociado, campos de texto para observaciones, así como para registrar la cantidad de producto y la cantidad de agua medidas. Además, hay dos campos de texto con métodos *suggest* para el código de tanque y para el usuario varillador.

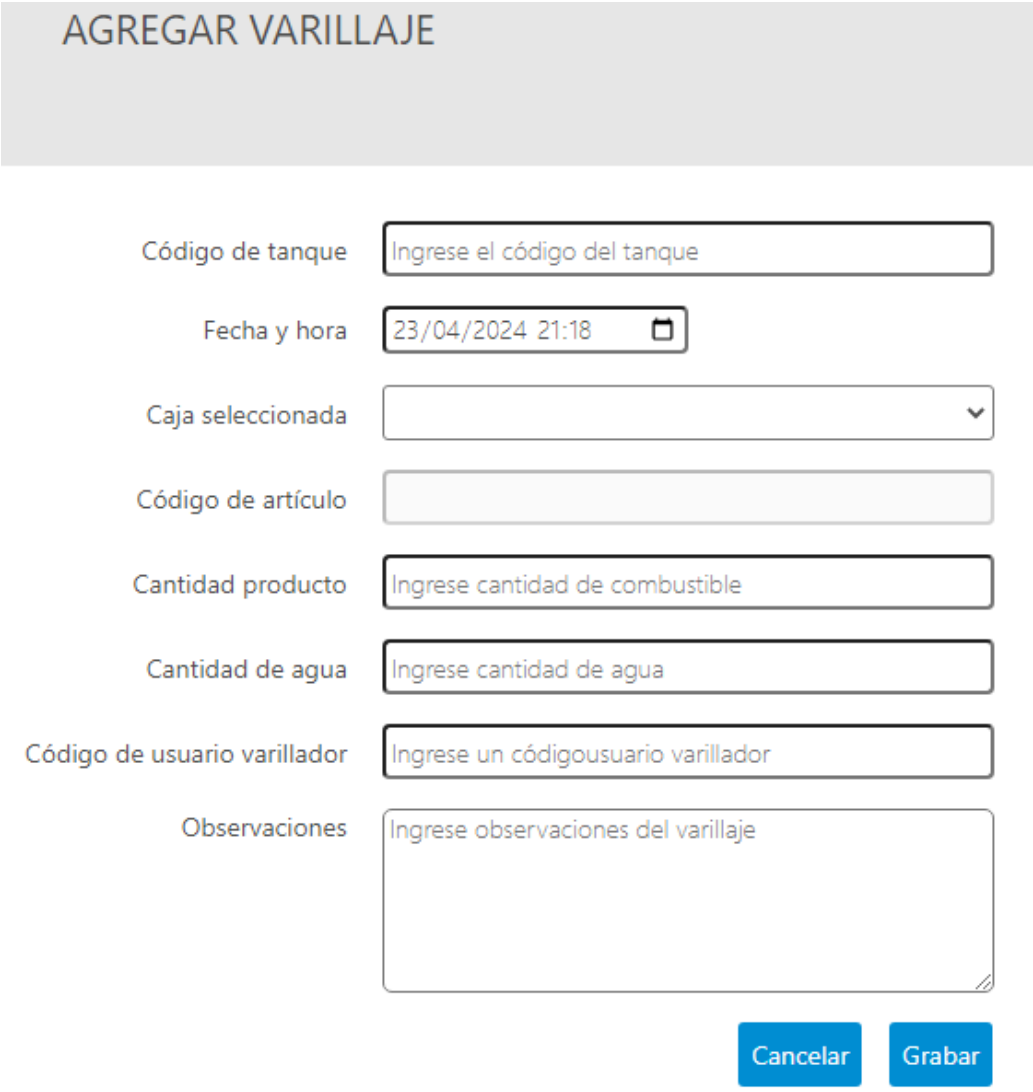


Imagen 14. Formulario nuevo-editar-varillaje

En cuanto a estos campos, el funcionamiento es similar al descrito en la primera petición. Se utiliza un método *suggest* que obtiene datos de un Web Service. El método para la carga de los usuarios ya existía, pero tuvimos que desarrollar el método para la carga del tanque adaptando los métodos creados previamente para sucursales a nuestros nuevos

requerimientos de tanques. Creamos el método *getTanques()* (ver Imagen 15) dentro del script.

```
getTanques: function (hint) {
    return this._api.traerTanquesSuggest(hint, 0, 0)
        .then(function (resultado) {
            return resultado;
        }, function (error) {
            this.mensajesError = errores;
        }.bind(this));
},
```

Imagen 15. Fragmento de código del archivo varillaje-nuevo-editar-vm.js

Este método hace una llamada a la función *traerTanquesSuggest* en el Web Service llamado api-gestion-comercial, el cual contiene todas las funciones de *suggest* y de ABM explicadas hasta el momento.

```
traerTanquesSuggest: function (hint, inicio, cantidad) {
    return this._$http.get(Utiles.prefijoApi(this._subscriptor) + 'TanqueConsultas/Tanques?hint=' + hint + '&inicio=' + inicio + '&cantidad=' + cantidad)
        .then(function (resultado) {
            return resultado.data;
        }, function (error) {
            return this._$q.reject(error.data);
        }.bind(this));
},
```

Imagen 16. Fragmento de código del archivo api-gestion-comercial

La imagen 16 muestra cómo esta función, en su segunda línea, se comunica con el método Tanques del archivo *TanqueConsultasController*. A través de lenguaje LINQ, este método realiza una petición a la base de datos (Request). Dentro de esta consulta, se utiliza un filtro de palabras que el usuario escribió en el campo de texto y se ordena el resultado por el campo descripción de los tanques devueltos por la base de datos en su respuesta (Response). Esta respuesta incluye los datos del objeto artículo asociado al tanque que se utilizará para rellenar el campo deshabilitado al que se hace referencia, así como la restricción de cargar el artículo automáticamente al seleccionar un tanque. Todo esto se muestra en la Imagen 17.

```
var todas = _consultasRepositorio.Queryable<Tanque>();
foreach (var palabra in palabras)
{
    todas =  todas.Where(
                    a => a.CodigoTanque.Contains(palabra) ||
                         a.Descripcion.Contains(palabra));
}

var ItemsADevolver = todas.Select(a => new TanqueDatosSimple {
                                    CodigoTanque = a.CodigoTanque,
                                    Descripcion = a.Descripcion,
                                    articuloDatos = a.Articulo != null ? new ArticuloDatos
                                    {
                                        CodigoArticulo = a.Articulo.CodigoArticulo,  // (codsto)
                                        Nombre = a.Articulo.Nombre
                                    } : null

});

if (inicio > 0)
    ItemsADevolver = ItemsADevolver.Skip(inicio);
if (cantidad > 0)
    ItemsADevolver = ItemsADevolver.Take(cantidad);

return ItemsADevolver.OrderBy(a => a.Descripcion).ToList();
```

Imagen 17. Fragmento de código de la consulta LINQ para el acceso a datos de tanques

Al analizar ahora el archivo *AdministrarVarillaje.cshtml*, a diferencia de las vistas anteriores, encontramos el acordeón de filtros (ver Imagen 18) y la paginación de la grilla de varillajes (ver Imagen 19).

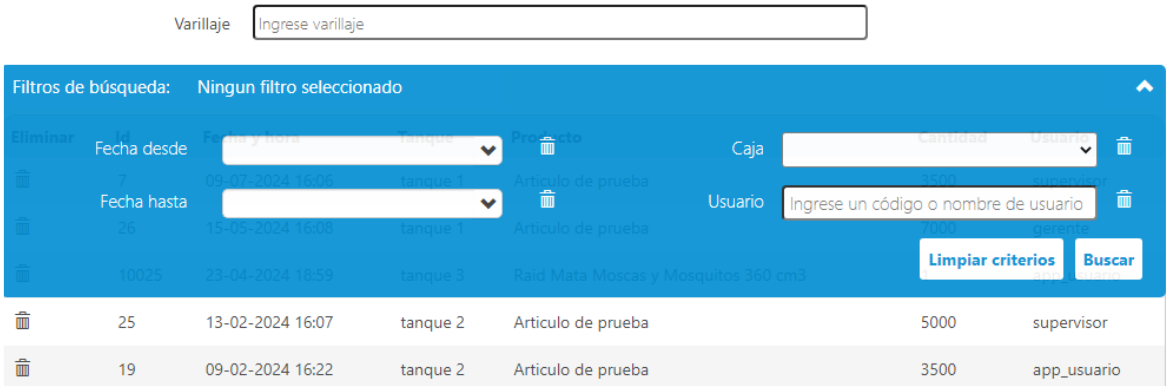


Imagen 18. Acordeón de filtros de varillajes

🗑	14	28-12-2020 16:19	tanque 3	Raid Mata Moscas y Mosquitos 360 cm3	5000	sis
🗑	23	13-11-2020 16:25	tanque 1	Articulo de prueba	10000	supervisor
🗑	11	26-09-2019 16:16	tanque 1	Articulo de prueba	15000	app_usuario

Anterior 1 2 Siguiente

Imagen 19. Paginación de grilla de varillajes

Para explicar estas dos solicitudes, debo referirme al archivo *varillaje-administrar-vm.js*. Aquí se encuentra un método desarrollado para obtener los datos de varillajes teniendo en cuenta los filtros seleccionados y la paginación requerida.

```
getVarillajesPaginadoConFiltros: function (hint, reactivoRefresh = false) {
    var dto = {
        FechaDesde: this.fechaDesdeFiltro != null ?
            new Date(this.fechaDesdeFiltro.split('/')[2], this.fechaDesdeFiltro.split('/')[1] - 1, this.fechaDesdeFiltro.split('/')[0], 0, 0)
            : null,
        FechaHasta: this.fechaHastaFiltro != null ?
            new Date(this.fechaHastaFiltro.split('/')[2], this.fechaHastaFiltro.split('/')[1] - 1, this.fechaHastaFiltro.split('/')[0], 0, 0)
            : null,
        CodParte: this.cajasFiltro != null ?
            this.cajasFiltro.codigoCaja
            : null,
        CodigoUsuario: this.usuario != null ?
            this.usuario.codigo
            : null,
        Hint: hint,
        Inicio: (this.paginacion.paginaActual - 1) * this.paginacion.cantidadItemsPorPagina,
        Cantidad: this.paginacion.cantidadItemsPorPagina,
    };

    return this._api.traerVarillajes(dto).then(function (resultado) {
        this.espera = false;
        this.hintBusqueda = hint;
        this.paginacion.cantidadTotalItems = resultado.totalRegistros;
        this.varillajes = resultado.items;
        return resultado.items;
    }.bind(this), function (error) {
        this.espera = false;
        this.mensajesError = [];
        this.mensajesError.push(error);
    }.bind(this));
},
```

Imagen 20. Fragmento de código del archivo varillaje-administrar-vm.js

En la primera parte del método *getVarillajesPaginadoConFiltros()*, se toman los valores aplicados por el usuario en los cuatro filtros disponibles. A los filtros de fecha se les aplica una transformación en su formato para que coincida con el formato utilizado en la base de datos. Se crea una variable que será utilizada como un DTO con estos cuatro valores (Fecha Desde, Fecha Hasta, Código de caja y Código de usuario). En las últimas dos líneas de la creación de esta variable DTO, se definen la página de inicio y la cantidad de elementos a

mostrar por página. Finalmente, se retorna un método invocando el Web Service encargado de traer la información de los varillajes, pero con la variable DTO como parámetro.

```
traerVarillajes: function (dto) {
    return this._$http.post(Utiles.prefijoApi(this._subscriptor) + 'VarillajeConsultas/VarillajesPaginadoConFiltros', JSON.stringify(dto))
        .then(function (resultado) {
            return resultado.data;
        }, function (error) {
            return this._$q.reject(error.data);
        }.bind(this));
},
```

Imagen 21. Fragmento de código del archivo api-gestion-comercial.js

El Web Service se encarga de convertir el argumento recibido en un archivo tipo JSON (ver Imagen 21) que será leído por la consulta LINQ en el *VarillajesConsultasController.cs* (ver Imagen 22) y es quien finalmente solicita a la base de datos los varillajes que se mostrarán en la grilla.

```
public ResultadoConsulta<VarillajeDatosSimple> VarillajesPaginadoConFiltros(FiltrosVarillajes dto)
{
    if (dto.FechaDesde != null)
        dto.FechaDesde = dto.FechaDesde.Value.Date.AddHours(0).AddMinutes(0).AddSeconds(0);
    if (dto.FechaHasta != null)
        dto.FechaHasta = dto.FechaHasta.Value.Date.AddHours(23).AddMinutes(59).AddSeconds(59);

    var palabras = Palabras(dto.Hint ?? "");
    if (palabras.Length == 0)
        throw new ExcepcionWeb(string.Format(Mensajes.AccionArgumentoNoValido, "hint"));
```

Imagen 22 fragmento de código de archivo varillajeConsultasController.cs

Para concluir esta solicitud, realizamos dos modificaciones en el archivo *nuevoEditarTanque.cshtml*. La primera consistió en mover los botones de grabar y cancelar fuera de la zona de pestañas que estábamos implementando. Para lograrlo, simplemente modificamos la ubicación de un que contenía los controles <button> responsables de estas funciones, situándolos después del encabezado de la vista. De esta forma, el usuario siempre tendrá visible los botones y podrá grabar o cancelar la acción independientemente de la pestaña en la que se encuentre.

La segunda modificación realizada fue la adición de pestañas para mostrar la información específica sobre un tanque o los varillajes asociados a él. Para obtener los datos de los

varillajes asociados a un tanque, seguimos un proceso similar al explicado anteriormente para el uso de los filtros. Dentro del archivo *tanque-nuevo-editar-vm.js*, creamos un método llamado *getVarillajesTanque()*. Este método invoca a la función *traerVarillajesTanque* del Web Service, el cual se comunica con *varillajesConsultasController.cs*. Este controlador, a su vez, realiza una consulta a la base de datos utilizando LINQ, con el id del tanque como restricción y la fecha de los varillajes como campo de ordenamiento.

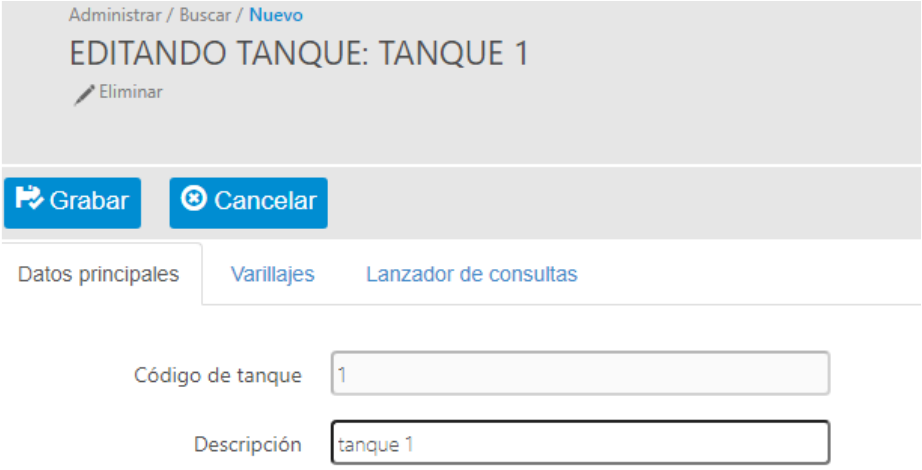


Imagen 23. Fragmento de la vista nuevoEditarTanque.cshtml

Petición de Reporteria sin denominación

A partir de este punto, iniciamos una nueva etapa en nuestra práctica profesional. Esta se centró en el diseño de diversos tipos de informes proporcionados por el sistema, relacionados con diferentes aspectos necesarios para la toma de decisiones de los clientes. Para obtener todos los datos necesarios para la generación de estos informes, fue fundamental comprender en detalle la estructura de la base de datos y enfocarnos en el lenguaje SQL.

Durante la práctica profesional nos familiarizamos con dos tipos de informes:

1. **Informes en formato XLS:** Estos informes presentan los datos en Excel, manteniendo los nombres de las columnas obtenidas en las consultas SQL.

2. **Informes en formato PDF:** Para este tipo de informes, diseñamos visualmente y organizamos los datos utilizando un archivo RDLC donde cargamos los datos obtenidos en la consulta SQL. Este formato de informe es estéticamente más atractivo y además permite la aplicación de diversas funciones, como cálculos matemáticos, manipulación de fechas y horas, entre otras.

Para esta solicitud, debimos diseñar los siguientes reportes en los dos formatos:

1. Tanques existentes
2. Varillajes existentes
3. Varillajes por tanque

En el caso de Tanques existentes y Varillajes existentes, tanto en su formato XLS como en PDF, tuvimos que agregar un acceso en *menú-lateral.js*, dentro del submenú de Estaciones. Este acceso lleva a una vista denominada "lanzador de consultas", la cual es utilizada en varios módulos del sistema. Para lograr esto, fue necesario adaptar una de las vistas ya existentes. En cambio, para varillajes por tanques tuvimos que agregar una nueva pestaña en la vista *nuevoEditarTanque.cshtml* donde ubicamos el lanzador de consultas.

Los lanzadores de consultas consisten en dos ComboBox: uno que lista todos los reportes asociados a ese módulo y otro para filtrar los reportes según el formato en que se generan. Además, cuentan con un campo de observaciones donde se proporciona una breve descripción de la información que ofrece el reporte seleccionado, así como un espacio de previsualización de reportes para aquellos configurados para esta opción.

Los reportes pueden contener distintos filtros para su generación, los cuales se muestran en un acordeón de filtros ubicado debajo de la selección de reportes.

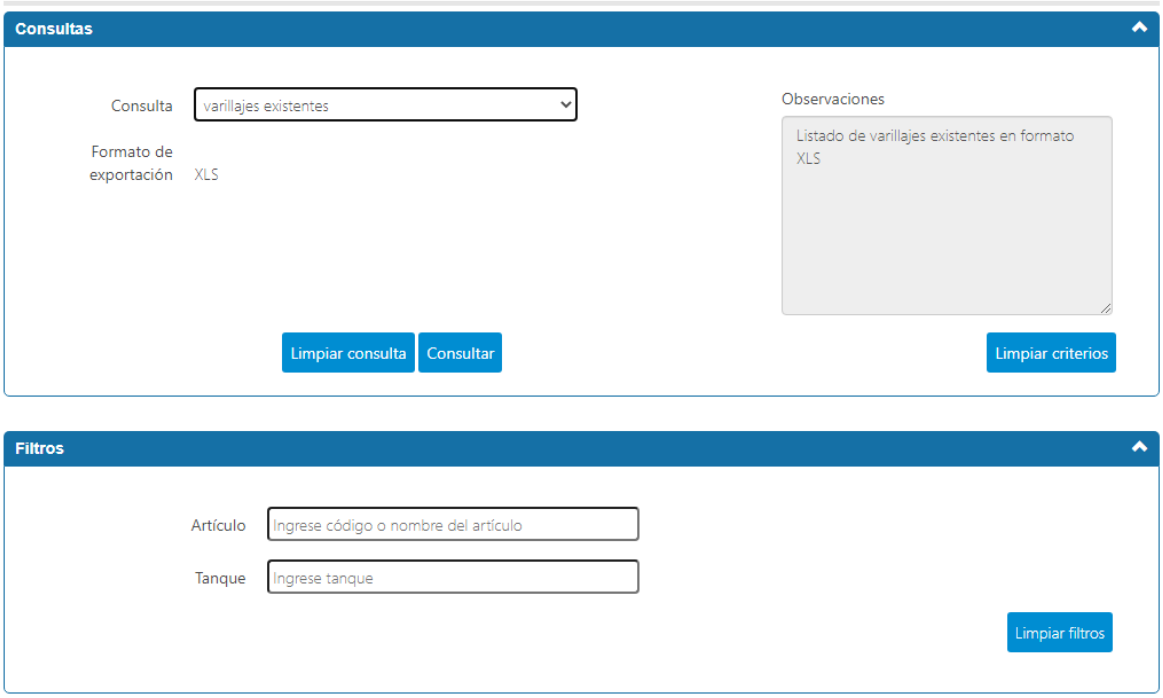


Imagen 24. Lanzador de consultas de estaciones

Para definir estos filtros, modificamos 3 archivos:

1. ***filtrosConsultasDirective.cshtml*:** Aquí se define la visualización del campo utilizado en el acordeón de filtros.

2. ***filtros-consultas-directive.js*:** En este archivo se guardan los valores de filtros seleccionados por el usuario para enviarse al controlador *generadorConsultasController.cs*.

3. ***generadorConsultasController.cs*:** En este archivo, se encuentra un método llamado "metodoGeneral", el cual recibe como parámetros un String con todos los nombres de los filtros elegidos entre signos de mayor y menor que (por ejemplo, <<codigoTanque>>), y un objeto de datos tipo parametrosConsultaGeneral. Este último es una clase diseñada con la definición de todos los campos usados en los filtros. Por lo tanto, si el nombre de un filtro está incluido, se accede al dato asociado que permite obtener de la base de datos la información necesaria para el reporte.

Una vez agregados los lanzadores de consultas, fue necesario diseñar las consultas SQL correspondientes a cada reporte. Para esto, empleamos el lenguaje Transact SQL, el cual permite almacenar temporalmente valores en distintas variables. Tomemos como ejemplo el reporte de "Tanques existentes" en sus dos versiones.

```
declare @CodigoTanque as varchar(10)

set @CodigoTanque = '<<CodigoTanque>>'

select * from E_tanques where @CodigoTanque is null or CodigoTanque = @CodigoTanque
```

Imagen 25. Consulta SQL para reporte tanques existentes XLS

En la consulta para generar el reporte en formato XLS (ver Imagen 25), observamos que:

1. Se declara la variable codigoTanque para almacenar el dato obtenido del filtro homónimo del lanzador de consultas.

2. La variable toma el valor del filtro en la segunda línea, que puede ser un código específico o null si el filtro no fue utilizado.

3. En la última línea, se listan todos los campos de la tabla de tanques con la restricción de CodigoTanque si la variable tiene valor.

En la consulta para el mismo reporte, pero en formato PDF (ver Imagen 26), encontramos una diferencia:

1. Mientras que la declaración y asignación de variables utilizadas para el filtro y la restricción de la consulta en la cláusula WHERE son iguales,

2. El listado de las columnas es uno a uno, utilizando para cada caso un alias de columna específico.

```
declare @CodigoTanque as varchar(10)
set @CodigoTanque = '<<CodigoTanque>>'

select
   CodigoTanque as ColumnaUno,  'CodigoTanque' as NombreColumnaUno,
   Descripcion as ColumnaDos,  'Descripción' as NombreColumnaDos,
   codsto as ColumnaTres,  'Codigo de Articulo' as NombreColumnaTres,
   cast(capacidadlitros as varchar) as ColumnaCuatro,  'Capacidad en Litros' as NombreColumnaCuatro,
   cast(Id_sistema_externo as varchar) as ColumnaCinco, 'Id sistema externo' as NombreColumnaCinco,
   cast(diametro as varchar) as ColumnaSeis, 'Diametro' as NombreColumnaSeis,
   cast(angulo_inclinacion as varchar) as ColumnaSiete, 'Angulo de inclinacion' as NombreColumnaSiete,
   cast(altura as varchar) as ColumnaOcho, 'Altura' as NombreColumnaOcho,
   cast(largo as varchar) as ColumnaNueve,
from   E_tanques  where  @CodigoTanque is null or CodigoTanque = @CodigoTanque
```

Imagen 26. Consulta SQL para reporte tanques existentes PDF

Entonces, para cada campo a mostrar en el informe, se utilizan dos columnas en el DataSet GenericoUno. La primera columna se denomina ColumnaUno, donde se almacenan los valores referidos al campo CodigoTanque de la tabla de tanques. La segunda columna se denomina NombreColumnaUno y muestra el valor del encabezado que va a tener la columna CodigoTanque dentro del reporte.

Este enfoque se repite para todos los campos a mostrar, donde cada campo tiene su propia columna para los valores y otra para los encabezados correspondientes. Además, se debe realizar un casteo de los valores que no sean cadenas de texto para visualizarlos correctamente en el informe.

Los nombres de alias como ColumnaUno, NombreColumnaUno, etc., utilizados en la consulta, en realidad corresponden a nombres de variables incluidas en el objeto DataSet GenericoUno. Este DataSet se encarga de recopilar la información proporcionada por la consulta y volcarla en el archivo RDLC asignado en el lanzador de consultas.

Es importante destacar que el DataSet GenericoUno cuenta con 14 variables diferentes que pueden utilizarse para asignar valores de consultas SQL. Además, existe otro DataSet

llamado GenericoDos con capacidad para alojar 40 variables, el cual se reserva para reportes más complejos, aunque no fue necesario utilizarlo durante la práctica profesional.

Código de tanque: 2 **Fecha y Hora:** 4/14/2024 9:39:59 PM

Usuario: sis

Pagina: 1

Varillajes por tanque (PDF)

ID	Fecha y hora	Nro. Caja	Nro. Articulo	Cantidad de producto	Cantidad de agua	Codigo del varillador
20003	20/02/2024 19:23	5	2	20.00	20000.00	2
30005	14/04/2024 20:01	6	2	12.80	1.70	3
30007	14/04/2024 20:01	6	2	14.20	0.90	3
30009	14/04/2024 20:01	6	2	11.30	1.40	3
30011	14/04/2024 20:01	6	2	13.10	1.10	3
30013	14/04/2024 20:01	6	2	8.70	2.60	3
30015	14/04/2024 20:02	6	2	12.80	1.70	3
30017	14/04/2024 20:02	6	2	14.20	0.90	3
30019	14/04/2024 20:02	6	2	11.30	1.40	3
30021	14/04/2024 20:02	6	2	13.10	1.10	3
30023	14/04/2024 20:02	6	2	8.70	2.60	3

Imagen 27. Ejemplo de Reporte en formato PDF para Varillajes por Tanque

Petición #1929

Durante las últimas dos semanas de nuestra práctica profesional, cada practicante recibió una tarea específica relacionada con la generación de informes en un sistema de gestión hotelera. Previo a esta etapa, tuvimos una reunión detallada conducida por Lucas Grandi,

donde se nos proporcionó una explicación exhaustiva sobre el funcionamiento del sistema y las interrelaciones entre las tablas de la base de datos.

Mi tarea estaba basada en la creación de un reporte en formato PDF que mostrara distintos datos personales sobre los pasajeros que se encontraban alojados en el hotel en distintos momentos, además de datos relacionados a la reserva realizada por el pasajero. Todo esto debía estar agrupado por cada reserva que se encontraba realizada en el sistema.

Las fechas entre las cuales era solicitado el reporte eran brindados por un agente de la policía que se hace presente en el establecimiento. El mismo es listado obligatorio en todo sistema hotelero. Según la ley actual la policía debería pedirlo todos los días.

El proceso de creación de este informe fue similar al utilizado en la petición anterior. En primer lugar, comencé a estudiar las tablas que iba a necesitar y sus relaciones en la base de datos. Con un total de 16 tablas involucradas, desarrolle la consulta SQL por partes hasta obtener la consulta final requerida. Luego, adapte un archivo RDLC utilizado en otros reportes para visualizar la información con agrupamiento de datos. Esto implicó reorganizar algunas variables existentes en el archivo y agregar otras nuevas según mis necesidades.

LISTADO POLICIAL

Emision:«Expr»

RESERVA							
NRO RESERVA	HABITACION	FECHA INGRESO		FECHA EGRESO		PASAJEROS	ESTADO
[ColumnaUno]	[ColumnaDos]	[ColumnaTres]		[ColumnaCuatro]		[ColumnaCinco]	[ColumnaSeis]
APELLIDO Y NOMBRE	DOMICILIO	LOCALIDAD	PROVINCIA	PAIS	TIPO DOC	NRO DOC	FECHA NAC.
[ColumnaSiete]	[ColumnaOcho]	[ColumnaNueve]	[ColumnaDiez]	NombreColumnaUno	ombreColumnaDc	[NombreColumnaTres]	lombreColumnaCuatrc

ACOMPAÑANTES		
APELLIDO Y NOMBRE	TIPO DOC	NRO DOCUMENTO
[NombreColumnaCinco]	[NombreColumnaSeis]	[NombreColumnaSiete]

Página: «Expr» / [&TotalPages]

Imagen 28. Archivo RDLC Listado Policial agrupado por reserva

En la Imagen 28 se presenta el archivo antes mencionado. Las columnas tienen cabeceras fijas, las cuales están adicionadas directamente en el reporte. Debe de informar los datos: Fecha y hora de emisión del listado, Nro. de reserva, Habitación, Fecha de ingreso y partida, Cantidad de pasajeros, Estado al momento de emitir en listado (check in o check out solamente), Nombre y Apellido del pasajero, Domicilio, Localidad, Provincia y País de residencia, Tipo de identificación (DNI, LE, LC, identificación internacional y pasaporte), Numero de identificación, Fecha de nacimiento. Sobre los acompañantes debe brindar los datos de: Apellido y Nombre, Tipo y Número de identificación

Una vez completado el diseño del archivo RDLC y obtenidos todos los datos necesarios en mi consulta SQL, procedí a asignar cada columna a una variable correspondiente en el archivo RDLC.

Finalmente, integré mi reporte al lanzador de consultas de hoteles, especificando la consulta SQL, el reporte asociado, el DataSet a utilizar y las características generales, como nombre, descripción y formato.

LISTADO POLICIAL

Emision:5/30/2024 6:45:10 PM

Desde: 01/01/2023 | Hasta: 30/05/2024

RESERVA							
NRO RESERVA	HABITACION	FECHA INGRESO		FECHA EGRESO		PASAJEROS	ESTADO
1235	403	03/03/2024		12/03/2024		2	Ocupada
APELLIDO Y NOMBRE	DOMICILIO	LOCALIDAD	PROVINCIA	PAIS	TIPO DOC	NRO DOC	FECHA NAC.
	juan de palotes s/n	N/A	N/A	N/A		00000005	N/A

ACOMPAÑANTES		
APELLIDO Y NOMBRE	TIPO DOC	NRO DOCUMENTO
	N/A	

RESERVA							
NRO RESERVA	HABITACION	FECHA INGRESO		FECHA EGRESO		PASAJEROS	ESTADO
1242	703	12/03/2024		16/03/2024		2	Ocupada
APELLIDO Y NOMBRE	DOMICILIO	LOCALIDAD	PROVINCIA	PAIS	TIPO DOC	NRO DOC	FECHA NAC.
Padilla Luca	Parana 2235	N/A	N/A	N/A		55999055	N/A

ACOMPAÑANTES		
APELLIDO Y NOMBRE	TIPO DOC	NRO DOCUMENTO
Ulrich Maximiliano	N/A	33919064

Página: 3 / 12

Imagen 29. Fragmento del reporte Listado Policial

Conclusiones

Aprendizaje obtenido de la realización de la práctica

Durante mi período de práctica profesional en Grandi y Asociados, adquirí conocimientos significativos sobre el funcionamiento de un sistema web integral y la integración de Web Services en su arquitectura. La reutilización de estos servicios resultó crucial para agilizar el desarrollo de nuevas funcionalidades y la expansión de las existentes.

Trabajar en un entorno colaborativo con compañeros de nivel académico similar fue una experiencia altamente enriquecedora. Aprendí a debatir ideas y alcanzar consensos durante el proceso de desarrollo, así como a ofrecer y solicitar ayuda cuando fue necesario.

Desarrollé habilidades en la gestión eficiente del tiempo para cumplir con los plazos establecidos para cada tarea, lo cual es esencial en un entorno laboral profesional. También me adapté rápidamente a nuevas herramientas y tecnologías que no formaban parte de mi formación académica, demostrando flexibilidad y disposición para aprender continuamente.

Comentarios personales del trabajo realizado

Mi experiencia en Grandi y Asociados fue fundamental para adentrarme en el ámbito laboral de la tecnología de la información. La oportunidad de aplicar los conocimientos adquiridos en la universidad en proyectos reales dentro de un entorno de producción fue invaluable para entender mi posición al salir al mercado laboral.

El equipo de la empresa siempre nos brindó un trato cordial y generoso, mostrándose dispuestos a ayudarnos a comprender conceptos y a estar al tanto de nuestro progreso. Esta actitud facilitó enormemente mi integración y aprendizaje en el entorno laboral.

Enfrentar y superar distintos desafíos durante este período aumentó mi entusiasmo y pasión por esta profesión, reafirmando la elección de mi carrera. Cada obstáculo superado fortaleció mi confianza y habilidades técnicas.

El trabajo en Grandi y Asociados me permitió desarrollar habilidades clave como la gestión del tiempo, la resolución de problemas y la adaptabilidad a nuevas tecnologías, aspectos esenciales para el éxito en el campo de la tecnología de la información.

Conclusiones generales

La culminación de mi práctica profesional en Grandi y Asociados y la elaboración de este informe representan tanto el cierre de un ciclo significativo como el comienzo de un nuevo camino lleno de aprendizaje y desafíos. A lo largo de estos meses, he experimentado un notable crecimiento profesional y personal, enfrentando y superando retos que han complementado de manera invaluable mi formación académica.

Mi tiempo en la empresa ha sido una oportunidad única para adentrarme en el ámbito laboral de la tecnología de la información. La posibilidad de trabajar en un entorno colaborativo con compañeros y profesionales experimentados ha enriquecido mi comprensión de la industria. Participar en un proyecto real de desarrollo de software, desde la creación de ABMs hasta el diseño y generación de reportes, me ha permitido aplicar conceptos clave como el patrón MVC, la inyección de dependencias y el uso de tecnologías como Web API, Entity Framework y SQL Server Reporting Services.

La capacitación proporcionada y el constante apoyo del equipo han sido fundamentales para integrarme eficazmente y abordar las tareas asignadas. El trato cordial y generoso del equipo, siempre dispuesto a ayudar y guiar, ha facilitado mi integración y aprendizaje. Cada semana,

las solicitudes de la empresa incrementaban su nivel de complejidad, convirtiendo cada jornada en una oportunidad de aprendizaje constante y sostenido.

Trabajar en una aplicación utilizada por múltiples clientes me ha brindado una perspectiva práctica sobre los conceptos teóricos estudiados en la universidad y ha sido un desafío considerable, permitiéndome identificar áreas de interés y aquellas en las que debo mejorar. La interacción con diversas perspectivas y el trabajo en equipo han sido elementos clave para enriquecer mis conocimientos y habilidades.

En resumen, mi experiencia en Grandi y Asociados ha sido extremadamente enriquecedora. He adquirido nuevas competencias técnicas, profesionales y personales, y he obtenido una visión más amplia del mundo laboral en el área de desarrollo de software. Me siento profundamente agradecido por la oportunidad de haber formado parte de esta empresa y estoy entusiasmado por aplicar todo lo aprendido en mi futura carrera profesional.

Recomendaciones

Basado en mi experiencia, la habilidad para comunicarse claramente y colaborar con los demás es fundamental en el desarrollo de software. Estas habilidades deben ser cultivadas constantemente para asegurar una colaboración eficaz y lograr resultados exitosos.

También, el buscar participar en una variedad de proyectos, especialmente en los primeros años profesionales, es crucial. La exposición a diferentes tecnologías, problemas y enfoques de desarrollo enriquece la experiencia y amplía el conjunto de habilidades de manera significativa.

Mantener una Actitud Proactiva y de Mejora Continua como estar siempre dispuesto a asumir nuevos desafíos y responsabilidades. Buscar mejorar continuamente tus habilidades técnicas y blandas es clave para el crecimiento profesional.

Por último, adherirse a principios éticos en todas las actividades profesionales es esencial. Esto incluye la honestidad, la integridad y el respeto por el trabajo de los demás.

Aportes

Visto lo realizado durante toda la práctica profesional considero que realicé varios aportes significativos al equipo de trabajo.

Mi habilidad para entender lógicamente las distintas tareas que debíamos resolver me permitió abordar desafíos con un enfoque estructurado y sistemático, facilitando la identificación de soluciones eficientes para los problemas que surgieron.

Mi dedicación a la búsqueda de información relacionada con las tareas solicitadas en sitios especializados ayudó a que el equipo tuviera acceso a recursos actualizados y relevantes, mejorando la calidad y eficiencia de nuestro trabajo.

Y sumando mi disposición para ir más allá de mis responsabilidades habituales y ayudar a mis compañeros en la resolución de problemas garantizó un ambiente de trabajo colaborativo y productivo.

Estos aportes no solo contribuyeron al éxito de los proyectos en los que trabajé, sino que también me permitieron desarrollar habilidades cruciales para mi futuro profesional.

Anexos

Las siguientes imágenes representan la carga horaria diaria en el sistema Redmine, que utiliza la empresa para la gestión de tareas y tiempos dedicados a cada una de ellas:

30/5/24, 20:09 Tiempo dedicado - Detalles - PPS 2023 - Redmine GYA

Tiempo dedicado

Horas: **301.25**

Fecha	Usuario	Actividad	Petición	Comentario	Horas	Estado	Proyecto	
2023-10-30	Abril Lenardon	Reuniones		Reunion con Maria y Lucas sobre sistemas de la Empresa y temario general de capacitación	**2.00**		PPS 2023	
2023-10-30	Abril Lenardon	Investigacion		Investigación sobre IDE SQL Server	**1.00**		PPS 2023	
2023-10-31	Abril Lenardon	Capacitación Interna		Capacitacion con Maria sobre el uso de la herramienta Redmine	**0.50**		PPS 2023	
2023-10-31	Abril Lenardon	Investigacion		Investigacion completa sobre SQL Server y BDD	**2.00**		PPS 2023	
2023-10-31	Abril Lenardon	Investigacion		Investigacion sobre Source Tree	**1.00**		PPS 2023	
2023-11-01	Abril Lenardon	Investigacion		Investigación Arquitectura de las soluciones (Full Web, Cliente - Servidor, Saas, Responsive y demas)	**4.00**		PPS 2023	
2023-11-02	Abril Lenardon	Investigacion		Investigacion Aplicaciones Funcionales (ABM, Tablas Maestras y ERP)	**3.50**		PPS 2023	
2023-11-02	Abril Lenardon	Reuniones		Reunion con compañeros PPS para puesta en comun de dudas y detalles de los temas a investigar	**0.25**		PPS 2023	
2023-11-03	Abril Lenardon	Capacitación Interna		Capacitacion Interna con Gustavo Toobe sobre diseño y funcionamiento de un ABM de la empresa	**1.50**		PPS 2023	
2023-11-03	Abril Lenardon	Investigacion		Investigacion sobre Omnicanalidad y TDD	**2.50**		PPS 2023	

redmine.gya.com.ar/projects/pps-agosto-2021/time_entries?sort=spent_on 1/12

30/5/24, 20:09 Tiempo dedicado - Detalles - PPS 2023 - Redmine GYA

Fecha	Usuario	Actividad	Petición	Comentario	Horas	Estado	Proyecto	
2023-11-06	Abril Lenardon	Investigacion		Investigacion sobre .NET	4.00		PPS 2023	
2023-11-07	Abril Lenardon	Investigacion		Investigacion y repaso sobre Logica de Negocios	3.50		PPS 2023	
2023-11-08	Abril Lenardon	Implementacion		Investigacion C# Avanzado (Delegados, Hash, SignalR)	3.00		PPS 2023	
2023-11-09	Abril Lenardon	Investigacion		Investigacion sobre paquetes Nuget, Programacion Orientada a Aspectos y Entity Framework	4.00		PPS 2023	
2023-11-10	Abril Lenardon	Investigacion		Investigacion Entity Framework	3.00		PPS 2023	
2023-11-10	Abril Lenardon	Capacitación Interna		Capacitacion con Lucas sobre SQL Server (Funcionamiento, Nomenclatura de Tablas)	1.00		PPS 2023	
2023-11-13	Abril Lenardon	Investigacion		Investigacion y busqueda de informacion sobre LinQ	4.00		PPS 2023	
2023-11-14	Abril Lenardon	Investigacion		Investigacion sobre Reflection	1.50		PPS 2023	
2023-11-14	Abril Lenardon	Investigacion		Investigación y busqueda de material sobre Inyección de dependencias	2.50		PPS 2023	
2023-11-15	Abril Lenardon	Investigacion		Investigacion sobre JS	3.00		PPS 2023	
2023-11-16	Abril Lenardon	Investigacion		Repaso completo sobre Java	1.00		PPS 2023	
2023-11-16	Abril Lenardon	Investigacion		Investigacion sobre Angular (Curso Angular 7)	4.00		PPS 2023	
2023-11-17	Abril Lenardon	Investigacion		Investigacion sobre MVC	4.00		PPS 2023	
2023-11-17	Abril Lenardon	Capacitación Interna		Capacitacion con Lucas sobre SQL Server (Nomenclatura y Relacion de las tablas)	1.50		PPS 2023	

redmine.gya.com.ar/projects/pps-agosto-2021/time_entries?sort=spent_on 2/12

30/5/24, 20:09 Tiempo dedicado - Detalles - PPS 2023 - Redmine GYA

Fecha	Usuario	Actividad	Petición	Comentario	Horas	Estado	Proyecto	
2023-11-21	Abril Lenardon	Investigacion		Investigacion sobre Angular (Continuacion Curso)	4.00		PPS 2023	
2023-11-22	Abril Lenardon	Investigacion		Investigacion WebApi	2.00		PPS 2023	
2023-11-22	Abril Lenardon	Investigacion		Investigacion sobre AngularJs	2.00		PPS 2023	
2023-11-23	Abril Lenardon	Investigacion		Investigacion sobre Angular (Continuacion Curso)	4.00		PPS 2023	
2023-11-24	Abril Lenardon	Investigacion		Investigacion sobre Angular (Continuacion Curso)	2.00		PPS 2023	
2023-11-24	Abril Lenardon	Investigacion		Investigacion metodos WebApi (GET, POST, DELETE)	2.00		PPS 2023	
2023-11-27	Abril Lenardon	Investigacion		Investigacion, instalacion y configuracion del entorno de desarrollo en Flutter	2.00		PPS 2023	
2023-11-27	Abril Lenardon	Capacitación Interna		Capacitacion con Mateo sobre desarrollo en Flutter	2.00		PPS 2023	
2023-11-28	Abril Lenardon	Investigacion		Investigacion sobre Flutter (CODELABS)	4.00		PPS 2023	
2023-11-29	Abril Lenardon	Investigacion		Investigacion sobre Flutter (Desarrollo, Layuots)	4.00		PPS 2023	
2023-11-30	Abril Lenardon	Investigacion		Investigacion sobre Reporteria (Dataset, Report Viewer)	4.00		PPS 2023	
2023-12-01	Abril Lenardon	Investigacion		Investigacion sobre Transact SQL	2.00		PPS 2023	
2023-12-01	Abril Lenardon	Capacitación Interna		Capacitacion con Luciano sobre uso de SourceTree y descarga repositorio	1.50		PPS 2023	
2023-12-04	Abril Lenardon	Desarrollo		Desarrollo proyecto genérico para aplicar conocimientos sobre MVC	4.00		PPS 2023	

redmine.gya.com.ar/projects/pps-agosto-2021/time_entries?sort=spent_on 3/12

30/5/24, 20:09 Tiempo dedicado - Detalles - PPS 2023 - Redmine GYA

Fecha	Usuario	Actividad	Petición	Comentario	Horas	Estado	Proyecto
2023-12-05	Abril Lenardon	Configuración		Descarga del repositorio con los proyectos de la empresa	**4.00**		PPS 2023
2023-12-11	Abril Lenardon	Configuración		Configuracion de dependencias del ProyectoCapas y WebGestionComercial	**4.00**		PPS 2023
2023-12-13	Abril Lenardon	Capacitación Interna		Capacitacion con Lucas sobre desarrollo peticion y resolucion de la misma	**2.50**		PPS 2023
2023-12-14	Abril Lenardon	Desarrollo		Resolucion de peticion cambio etiquetas abm comprobantes	**4.00**		PPS 2023
2023-12-15	Abril Lenardon	Corrección de Errores		Correccion de dependencias y actualizacion ramas	**5.00**		PPS 2023
2023-12-18	Abril Lenardon	Desarrollo	~~Desarrollo #1902~~: Para arrancar	Desarrollo en vista y viewmodel (campo Sucursal en ABM Consultorios)	**4.00**	Testing/Finalizada	PPS 2023
2023-12-19	Abril Lenardon	Desarrollo	~~Desarrollo #1902~~: Para arrancar	Desarrollo alta y modificar en base de datos (campo sucursal en ABM Consultorios)	**4.00**	Testing/Finalizada	PPS 2023
2023-12-20	Abril Lenardon	Desarrollo	~~Desarrollo #1902~~: Para arrancar	Finalizacion de la peticion	**1.00**	Testing/Finalizada	PPS 2023
2023-12-20	Abril Lenardon	Asistencia Técnica		Asistencia configuracion de errores	**3.00**		PPS 2023
2024-01-08	Abril Lenardon	Diseño	~~Tarea Generica #1921~~: 4 ABM del sistema de ESTACIONES DE SERVICIO	Presentacion con Lucas del sistema de Estaciones de Servicio	**2.00**	Testing/Finalizada	PPS 2023
2024-01-08	Abril Lenardon	Desarrollo	~~Tarea Generica #1921~~: 4 ABM del sistema de	Desarrollo ABM Surtidores	**2.00**	Testing/Finalizada	PPS 2023

redmine.gya.com.ar/projects/pps-agosto-2021/time_entries?sort=spent_on 4/12

30/5/24, 20:09 Tiempo dedicado - Detalles - PPS 2023 - Redmine GYA

Fecha	Usuario	Actividad	Petición	Comentario	Horas	Estado	Proyecto
			ESTACIONES DE SERVICIO				
2024-01-09	Abril Lenardon	Desarrollo	~~Tarea Generica #1921~~: 4 ABM del sistema de ESTACIONES DE SERVICIO	Desarrollo ABM Surtidores (Continuación)	**4.00**	Testing/Finalizada	PPS 2023
2024-01-10	Abril Lenardon	Desarrollo	~~Tarea Generica #1921~~: 4 ABM del sistema de ESTACIONES DE SERVICIO	Desarrollo ABM Mangueras	**4.00**	Testing/Finalizada	PPS 2023
2024-01-11	Abril Lenardon	Desarrollo	~~Tarea Generica #1921~~: 4 ABM del sistema de ESTACIONES DE SERVICIO	Desarrollo ABM Islas	**4.00**	Testing/Finalizada	PPS 2023
2024-01-12	Abril Lenardon	Asistencia Técnica		Asistencia tecnica solucion de errores, instalacion de dependencias y configuracion de bdd y entorno en proyecto de compañero	**2.00**		PPS 2023
2024-01-12	Abril Lenardon	Desarrollo	~~Tarea Generica #1921~~: 4 ABM del sistema de ESTACIONES DE SERVICIO	Desarrollo de vistas y configuracion (controladores de Manguera para WebGestionComercial)	**2.00**	Testing/Finalizada	PPS 2023
2024-01-15	Abril Lenardon	Desarrollo	~~Tarea Generica #1921~~: 4 ABM del sistema de	Se agregan items de menu lateral	**4.00**	Testing/Finalizada	PPS 2023

redmine.gya.com.ar/projects/pps-agosto-2021/time_entries?sort=spent_on 5/12

30/5/24, 20:09 Tiempo dedicado - Detalles - PPS 2023 - Redmine GYA

Fecha	Usuario	Actividad	Petición	Comentario	Horas	Estado	Proyecto	
			ESTACIONES DE SERVICIO					
2024-01-16	Abril Lenardon	Desarrollo	~~Tarea Generica #1921~~: 4 ABM del sistema de ESTACIONES DE SERVICIO	Solucion problemas de mapeo y desarrollo api Mangueras	**4.00**	Testing/Finalizada	PPS 2023	
2024-01-17	Abril Lenardon	Desarrollo	~~Tarea Generica #1921~~: 4 ABM del sistema de ESTACIONES DE SERVICIO	Se agregan entidades y metodos a la API de entidad Isla y ModelosSurtidor	**4.00**	Testing/Finalizada	PPS 2023	
2024-01-18	Abril Lenardon	Desarrollo	~~Tarea Generica #1921~~: 4 ABM del sistema de ESTACIONES DE SERVICIO	Se agregan funcionalidades al ABM de Surtidores e Islas, se corrigen errores	**4.00**	Testing/Finalizada	PPS 2023	
2024-01-19	Abril Lenardon	Desarrollo	~~Tarea Generica #1921~~: 4 ABM del sistema de ESTACIONES DE SERVICIO	Desarrollo Sistema Estaciones de Servicio (Se corrigen errores en entidad Manguera y Surtidor, se corrigen variables y se agregan suggest faltantes)	**4.00**	Testing/Finalizada	PPS 2023	
2024-01-22	Abril Lenardon	Testing	~~Tarea Generica #1921~~: 4 ABM del sistema de ESTACIONES DE SERVICIO	Prueba de funcionamiento de sistema de Estaciones de Servicio	**2.00**	Testing/Finalizada	PPS 2023	

redmine.gya.com.ar/projects/pps-agosto-2021/time_entries?sort=spent_on 6/12

30/5/24, 20:09 Tiempo dedicado - Detalles - PPS 2023 - Redmine GYA

Fecha	Usuario	Actividad	Petición	Comentario	Horas	Estado	Proyecto	
2024-01-22	Abril Lenardon	Desarrollo	~~Tarea Generica #1921~~: 4 ABM del sistema de ESTACIONES DE SERVICIO	Desarrollo ABM Tanques (Mapeo, DTO, Scripts)	**2.00**	Testing/Finalizada	PPS 2023	
2024-01-23	Abril Lenardon	Desarrollo	~~Tarea Generica #1921~~: 4 ABM del sistema de ESTACIONES DE SERVICIO	Desarrollo ABM Tanque (Finalizacion Script y Vistas)	**4.00**	Testing/Finalizada	PPS 2023	
2024-01-24	Abril Lenardon	Desarrollo	~~Tarea Generica #1921~~: 4 ABM del sistema de ESTACIONES DE SERVICIO	Finalizacion y prueba sistema de Estaciones de Servicio (Correccion etiquetas y errores varios)	**4.00**	Testing/Finalizada	PPS 2023	
2024-01-25	Abril Lenardon	Desarrollo	Desarrollo #1924: ABM de entidad con registracion de movimientos	Desarrollo entidad Varillaje (Mapeo, Scripts y DTOs)	**4.00**	En curso	PPS 2023	
2024-01-26	Abril Lenardon	Desarrollo	Desarrollo #1924: ABM de entidad con registracion de movimientos	Desarrollo ABM Varillaje (Vistas y etiquetas)	**4.00**	En curso	PPS 2023	
2024-01-29	Abril Lenardon	Desarrollo	Desarrollo #1924: ABM	Agregacion de Suggest necesarios y Reparacion Suggest usuarios ABM	**4.00**	En curso	PPS 2023	

redmine.gya.com.ar/projects/pps-agosto-2021/time_entries?sort=spent_on 7/12

30/5/24, 20:09 Tiempo dedicado - Detalles - PPS 2023 - Redmine GYA

Fecha	Usuario	Actividad	Petición	Comentario	Horas	Estado	Proyecto	
			de entidad con registracion de movimientos	Varillajes				
2024-01-30	Abril Lenardon	Desarrollo	Desarrollo #1924: ABM de entidad con registracion de movimientos	Corrección errores de la vista y carga de artículos en la elección del tanque, Agregación combo cajas activas	**4.00**	En curso	PPS 2023	
2024-01-31	Abril Lenardon	Desarrollo	Desarrollo #1924: ABM de entidad con registracion de movimientos	Solucion conflictos en vista VarillajeNuevoEditar. Solución errores en visualizacion de datos asociados a Articulo y Caja	**4.00**	En curso	PPS 2023	
2024-02-01	Abril Lenardon	Desarrollo	Desarrollo #1924: ABM de entidad con registracion de movimientos	Se solucionan errores en vista NuevoEditarVarillaje	**4.00**	En curso	PPS 2023	
2024-02-02	Abril Lenardon	Desarrollo	Desarrollo #1924: ABM de entidad con registracion de movimientos	Agregación pestaña Varillajes en View Tanques, se agrega GetVarillajeTanque en Web Service y se corrigen errores de la grilla de Varillajes	**4.00**	En curso	PPS 2023	

redmine.gya.com.ar/projects/pps-agosto-2021/time_entries?sort=spent_on 8/12

30/5/24, 20:09 Tiempo dedicado - Detalles - PPS 2023 - Redmine GYA

Fecha	Usuario	Actividad	Petición	Comentario	Horas	Estado	Proyecto	
2024-02-05	Abril Lenardon	Desarrollo	Desarrollo #1924: ABM de entidad con registracion de movimientos	Modificacion en diseño y forma de botones (Cancelar y Guardar) en vista Tanques. Puesta en comun sobre funcionamiento de los filtros y se comienza su implementacion	**4.00**	En curso	PPS 2023	
2024-02-06	Abril Lenardon	Desarrollo	Desarrollo #1924: ABM de entidad con registracion de movimientos	Desarrollo Filtros y Paginacion (ABM Varillajes)	**4.00**	En curso	PPS 2023	
2024-02-07	Abril Lenardon	Desarrollo	Desarrollo #1924: ABM de entidad con registracion de movimientos	Se mejora metodo GetPaginadoConFiltro para su funcionamiento en vista AdministrarVarillaje, ademas se agrega funcionalidad de Paginacion	**4.00**	En curso	PPS 2023	
2024-02-08	Abril Lenardon	Desarrollo	Desarrollo #1924: ABM de entidad con registracion de movimientos	Se agrega funcionalidad de los filtros comprendidos en el acordeon. Correccion de la disposicion de los mismos	**4.00**	En curso	PPS 2023	
2024-02-09	Abril Lenardon	Desarrollo	Desarrollo #1924: ABM de entidad con registracion	Finalizacion de la petición con solución de errores de visualización en el titulo de filtros, se agrega label en la edición de varillajes con caja cerrada para avisar que no se puede editar y se soluciona error de	**4.00**	En curso	PPS 2023	

redmine.gya.com.ar/projects/pps-agosto-2021/time_entries?sort=spent_on 9/12

30/5/24, 20:09 Tiempo dedicado - Detalles - PPS 2023 - Redmine GYA

Fecha	Usuario	Actividad	Petición	Comentario	Horas	Estado	Proyecto	
			de movimientos	actualización de articulo al modificar un varillaje				
2024-02-14	Abril Lenardon	Desarrollo		Se agrega seccion Lanzadores de consulta en menu lateral, seccion estaciones. Se agrega pestaña Lanzador de consultas en vista TanqueNuevoEditar.	**4.00**		PPS 2023	
2024-02-15	Abril Lenardon	Desarrollo		Se avanza en agregar funcionalidad al lanzador de consultas de estaciones	**4.00**		PPS 2023	
2024-02-16	Abril Lenardon	Desarrollo		Se agrega funcionalidad Suggest (Surtidor, Islas y Mangueras) en Lanzador de Consultas Estaciones	**4.00**		PPS 2023	
2024-02-19	Abril Lenardon	Desarrollo		se agrega consulta de tanques existentes y varillajes al lanzador de consultas de estaciones. pendiente de finalizacion	**3.50**		PPS 2023	
2024-02-19	Abril Lenardon	Reuniones		Presentación de avances referidos en reporteria a Lucas, lanzador de consultas de estaciones. vistazo a Transact SQL	**0.50**		PPS 2023	
2024-02-20	Abril Lenardon	Desarrollo		Correccion de SQL para reporte de tanques existentes. Agregacion en SQL para reporte de varillajes existentes con filtros de Tanques y Artículos	**4.00**		PPS 2023	
2024-02-21	Abril Lenardon	Capacitación Interna		Capacitacion sobre el funcionamiento de Report Viewer para el diseño de reportes en formato pdf	**0.50**		PPS 2023	
2024-02-21	Abril Lenardon	Desarrollo		Agregacion de funcionalidad Lanzador de Consultas en pestaña de ABM Tanque	**3.50**		PPS 2023	
2024-02-22	Abril Lenardon	Configuración		Descarga y configuración de Visual Studio 2015 para el uso de Report Viewer	**4.00**		PPS 2023	
2024-02-23	Abril Lenardon	Desarrollo		Diseño de Reporte de Tanques existentes en formato pdf con archivo ModeloBasicoHastaDiezColumnas2x9.rldc	**4.00**		PPS 2023	

redmine.gya.com.ar/projects/pps-agosto-2021/time_entries?sort=spent_on 10/12

30/5/24, 20:09 Tiempo dedicado - Detalles - PPS 2023 - Redmine GYA

Fecha	Usuario	Actividad	Petición	Comentario	Horas	Estado	Proyecto	
2024-02-26	Abril Lenardon	Desarrollo		Diseño de Reporte de Varillajes existentes y Varillajes por Tanque en formato pdf con archivos ModeloBasicoHastaSieteColumnas2x9.rldc y ModeloBasicoHastaOchoColumnas2x9 modificados para personalización de los reportes diseñados	**4.00**		PPS 2023	
2024-02-27	Abril Lenardon	Desarrollo		Correccion errores en Reportes que generaban una nueva pagina en blanco, corrigiendo margenes	**4.00**		PPS 2023	
2024-02-28	Abril Lenardon	Investigacion		Investigacion y Visualizacion de otros tipos de Reportes ademas del ModeloBasico	**4.00**		PPS 2023	
2024-02-29	Abril Lenardon	Investigacion		Investigación sobre consultas SQL asociadas a reportes con cortes de control y agrupamiento según rubro, y su implementación, tomando como ejemplo la consulta Lista de existencias por rubro (PDF).	**4.00**		PPS 2023	
2024-03-01	Abril Lenardon	Capacitación Interna		Capacitacion interna con Lucas sobre Reporteria	**1.00**		PPS 2023	
2024-03-04	Abril Lenardon	Desarrollo	Reporteria #1929: listado de Policia	Analisis de las tablas en la BDD y Desarrollo de la consulta necesaria para la obtencion de datos	**4.00**	Backlog	PPS 2023	
2024-03-05	Abril Lenardon	Diseño	Reporteria #1929: listado de Policia	Diseño de la vista del reporte y cuestiones asociadas al mismo	**4.00**	Backlog	PPS 2023	
2024-03-06	Abril Lenardon	Desarrollo	Reporteria #1929: listado de Policia	Correccion de errores en el Query para optimizacion	**4.00**	Backlog	PPS 2023	
2024-03-07	Abril Lenardon	Desarrollo	Reporteria #1929: listado de Policia	Finalizacion diseño del reporte y asociacion de valores necesarios	**3.00**	Backlog	PPS 2023	

redmine.gya.com.ar/projects/pps-agosto-2021/time_entries?sort=spent_on 11/12

30/5/24, 20:09 Tiempo dedicado - Detalles - PPS 2023 - Redmine GYA

Fecha	Usuario	Actividad	Petición	Comentario	Horas	Estado	Proyecto	
2024-03-08	Abril Lenardon	Desarrollo	Reporteria #1929: listado de Policia	Finalizacion Query necesario con aplicacion de filtros	**4.00**	Backlog	PPS 2023	
2024-03-11	Abril Lenardon	Diseño	Reporteria #1929: listado de Policia	Corrección de agrupamiento en reporte y rediseño del mismo consultado a Nicolas Diaz	**4.00**	Backlog	PPS 2023	

(1-96/96)

redmine.gya.com.ar/projects/pps-agosto-2021/time_entries?sort=spent_on 12/12