

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE INGENIERÍA

SISTEMAS DE INFORMACIÓN



TRABAJO DE TITULACIÓN

DESARROLLO DE UN SISTEMA DE GESTIÓN DE CONSULTA PARA EL
CONTROL DEL SERVICIO DE MANTENIMIENTO VEHICULAR

CASO DE ESTUDIO: TECNIMECÁNICA IBARRA

AUTOR:

NOELIA SALOMÉ QUISHPE ZAMBRANO

TUTOR:

SUYANA FABIOLA ARCOS VILLAGÓMEZ

QUITO DM, 2023

DEDICATORIA

Con profundo cariño y gratitud, dedico el presente trabajo a mis padres, quienes han sido mi apoyo incondicional en cada paso que he dado. La confianza constante que depositan en mí y su incansable esfuerzo para brindarme la oportunidad de estudiar son los pilares que han guiado mi camino, siempre acompañados de sus valiosos consejos, los cuales me han permitido crecer como persona y enfrentar cada obstáculo con determinación.

A mis hermanas, fieles compañeras en incontables experiencias de la vida, compartiendo tanto los momentos más felices como los más difíciles. Su presencia ha añadido una razón adicional para esforzarme cada día, buscando ser un referente positivo a seguir.

A mi abuelito Diego, por ser un ejemplo vivo de que la perseverancia y dedicación pueden llevarnos a lograr grandes cosas.

Finalmente, a mi abuelita Lupe, quien siempre ha estado ahí para mí, dándome ánimos para seguir adelante y no rendirme.

AGRADECIMIENTO

Agradezco a la Pontificia Universidad Católica del Ecuador, a la Facultad de Ingeniería, y a mis profesores, quienes me han brindado los conocimientos necesarios para ser un profesional exitoso. Su dedicación, paciencia y apoyo han sido ejemplos inspiradores para sus alumnos.

A mi tutora de trabajo de titulación, la Ing. Suyana Arcos, por su valioso apoyo, conocimientos, preocupación, dedicación de tiempo y esfuerzo para alcanzar la culminación del presente trabajo.

A mis compañeros y amigos de carrera con quienes he compartido vivencias significativas dentro de mi etapa universitaria. Su compañía ha enriquecido mi trayectoria académica y personal, brindándome valiosas lecciones y momentos compartidos.

Finalmente, reitero el agradecimiento a mis padres, ya que su apoyo, orientación y sacrificios han sido la fuerza impulsora detrás de cada logro alcanzado.

RESUMEN

El presente trabajo de titulación se centra en el desarrollo de un aplicativo web de consulta enfocado a simplificar la búsqueda de información crucial para la gestión del taller TecniMecánica Ibarra. Está dirigido específicamente al administrador, dueño de la mecánica, quien tiene problemas con el manejo del sistema actual debido a su compleja navegación.

El propósito principal del aplicativo es facilitar el seguimiento de los mantenimientos, desglosando los servicios y repuestos utilizados. Además, de proporcionar un seguimiento detallado de las existencias de inventario, permitiendo al administrador consultar fácilmente qué artículos y servicios deben considerarse en un nuevo mantenimiento en función de mantenimientos anteriores.

Además de atender consultas, se ha desarrollado un módulo para automatizar la gestión de órdenes de trabajo, las cuales actualmente se registran de forma manual.

El trabajo se ha planificado utilizando la metodología Scrum, lo que ha facilitado un desarrollo efectivo y constante gracias a la entrega periódica de avances, asegurando que el aplicativo final cumpla con las funcionalidades planificadas.

Palabras clave: Consulta, Mantenimientos, Órdenes de Trabajo, Scrum.

Abstract

This project centers around the development of a website designed to simplify the retrieval of essential information for workshop management. It is specifically tailored for the workshop administrator, who is encountering challenges with the current system due to its complexity.

The primary objective of the website is to streamline the tracking of maintenance activities by detailing the services and spare parts utilized. Additionally, it offers a comprehensive inventory overview, enabling the administrator to easily identify items and services required for upcoming maintenance based on past records.

In addition to addressing inquiries, a module has been implemented to automate the handling of work orders, currently performed manually. This process holds significance, as the work order serves as documentation of the vehicle's inventory and the necessary services when it enters the workshop.

The project has been strategically planned to use the Scrum methodology, facilitating effective and consistent development through periodic delivery of progress updates. This ensures that the final website aligns with the initially planned functionalities.

Keywords: Consultation, Maintenance, Work Orders, Scrum.

ÍNDICE

DEDICATORIA	II
AGRADECIMIENTO	III
RESUMEN	IV
ÍNDICE DE FIGURAS.....	IX
ÍNDICE DE TABLAS	XIII
CAPÍTULO I: INTRODUCCIÓN	1
1.1 Justificación.....	1
1.2 Antecedentes	1
1.3 Planteamiento del problema	2
1.4 Objetivo General	2
1.5 Objetivos Específicos.....	3
1.6 Alcance.....	3
CAPÍTULO II: MARCO TEÓRICO	4
2.1 Bases teóricas del servicio técnico automotriz.....	4
2.1.1 Sector automotriz	4
2.1.2 Taller mecánico.....	4
2.1.3 Tipos de mantenimiento vehicular	5
2.1.4 Importancia del mantenimiento preventivo	5
2.1.5 Control de mantenimiento preventivo	6
2.2 Metodologías de desarrollo de software.....	6
2.2.1 Tipos de metodología de desarrollo de software	6
2.2.2 Metodologías ágiles	7
2.2.3 Scrum	11
2.3 Sistemas de gestión de base de datos	13
2.3.1 Tipos de gestores de bases de datos	14
2.3.2 Sistema de gestión de base de dato relacional	15
2.3.3 Oracle Database Express Edition 11g.....	16
2.4 Herramientas de administración de bases de datos	17
2.4.1 Toad for Oracle Base Edition	17

2.4.2	Oracle Enterprise Manager	18
2.5	Lenguajes de programación de alto nivel.....	18
2.5.1	PL/SQL	18
2.5.2	C#.....	19
2.5.3	JavaScript.....	20
2.5.4	TypeScript.....	20
2.6	Aplicación web.....	21
2.6.1	Frontend	Error! Bookmark not defined.
2.6.2	Backend.....	Error! Bookmark not defined.
2.7	Arquitectura de software	22
2.7.1	API REST	22
2.7.2	API RESTFUL.....	22
2.8	Herramientas de desarrollo web	23
2.8.1	Visual Studio 2022 Community Edition.....	23
2.8.2	Visual Studio Code	23
2.8.3	Framework	24
2.8.4	GitHub.....	25
CAPÍTULO III: ANÁLISIS Y DISEÑO DEL SISTEMA WEB		27
3.1	Análisis de requerimientos	27
3.1.1	Especificación funcional del sistema	27
3.1.2	Casos de uso del sistema.....	29
3.2	Configuración del proyecto.....	50
3.3	Product Backlog General del Producto	50
3.4	Sprint Planning.....	1
3.5	Diseño de la base de datos del sistema.....	1
3.5.1	Base de datos fuente.....	1
3.5.2	Diseño de la base de datos objetivo	2
3.5.3	Procedimientos PL/SQL	3
3.5.4	Modelos de base de datos	3
3.6	Diagrama de arquitectura del sistema	4
CAPÍTULO IV: DESARROLLO Y PRUEBAS DEL SISTEMA WEB		5
4.1	Sprint 1	5
4.1.1	Sprint Backlog 1	5
4.1.2	Historias de usuario.....	5

4.1.3	Pruebas unitarias Sprint 1	12
4.1.4	Sprint Review 1.....	13
4.1.5	Sprint Retrospective 1.....	13
4.2	Sprint 2.....	14
4.2.1	Sprint Backlog 2	14
4.2.2	Historias de usuario.....	15
4.2.3	Pruebas unitarias Sprint 2	34
4.2.4	Sprint Review 2.....	36
4.2.5	Sprint Retrospective 2.....	36
4.3	Sprint 3.....	37
4.3.1	Sprint Backlog 3	37
4.3.2	Historias de usuario.....	38
4.3.3	Pruebas unitarias Sprint 3	51
4.3.4	Sprint Review 3.....	52
4.3.5	Sprint Retrospective 3.....	53
4.4	Pruebas de Integración	54
4.5	Pruebas de Integridad de Datos	55
4.6	Plan de Carga de Datos para el Aplicativo de Consultas	60
CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES		62
ANEXOS		64
BIBLIOGRAFÍA		69

ÍNDICE DE FIGURAS

Figura 1	Planificación de la iteración	13
Figura 2	Diagrama de caso de uso a nivel general	29
Figura 3	Diagrama de caso de uso RF0: Ingreso al sistema	30
Figura 4	Diagrama de caso de uso RF1: Gestión de Clientes.....	31
Figura 5	Diagrama de caso de uso RF1.1.1: Consulta general de Clientes	31
Figura 6	Diagrama de caso de uso RF1.1.2: Consulta por parámetro de Clientes	32
Figura 7	Diagrama de caso de uso RF1.2: Ingreso de Clientes	33
Figura 8	Diagrama de caso de uso RF1.3: Modificación de Clientes.....	34
Figura 9	Diagrama de caso de uso RF1.4: Eliminación de Clientes.....	34
Figura 10	Diagrama de caso de uso RF2: Consultas de Ventas	35
Figura 11	Diagrama de caso de uso RF2.1: Consulta general de Ventas	36
Figura 12	Diagrama de caso de uso RF2.2: Consulta por parámetro de Ventas	36
Figura 13	Diagrama de caso de uso RF2.3.1: Reporte de Ventas por Cliente.....	37
Figura 14	Diagrama de caso de uso RF2.3.1: Reporte de Ventas General.....	38
Figura 15	Diagrama de caso de uso RF2.4: Modificación de Garantía de Ventas	39
Figura 16	Diagrama de caso de uso RF3: Consultas de Inventario	41
Figura 17	Diagrama de caso de uso RF3.1: Consulta general de Inventario.....	41
Figura 18	Diagrama de caso de uso RF3.2: Consulta por parámetro de Ventas	42
Figura 19	Diagrama de caso de uso RF3.3.1: Reporte de Existencia Artículos por Categoría	43
Figura 20	Diagrama de caso de uso RF3.3.2: Reporte General de Existencia Artículos	44
Figura 21	Diagrama de caso de uso RF4: Gestión de Órdenes de Trabajo	45
Figura 22	Diagrama de caso de uso RF4.1: Ingreso de Órdenes de Trabajo.....	45
Figura 23	Diagrama de caso de uso RF4.2: Modificación de Órdenes de Trabajo	46
Figura 24	Diagrama de caso de uso RF4.3: Eliminación de Órdenes de Trabajo	47
Figura 25	Diagrama de caso de uso RF4.4.1: Consulta general de Órdenes de Trabajo.....	48
Figura 26	Diagrama de caso de uso RF4.4.2: Consulta por parámetro de Órdenes de Trabajo	48
Figura 27	Diagrama de caso de uso RF5: Salida del sistema	49
Figura 28	Diagrama Entidad-Relación: Modelo lógico de base de datos.....	3
Figura 29	Diagrama Entidad-Relación: Modelo físico de base de datos.....	4

Figura 30	Diagrama de arquitectura del sistema	4
Figura 31	Script de base de datos	6
Figura 32	Esquema de base de datos del aplicativo.....	7
Figura 33	Restricción de usuarios a la base de datos.....	7
Figura 34	Procedimientos almacenados - ETL.....	8
Figura 35	Carga de datos a la base de datos destino.....	9
Figura 36	Generación de la cadena de conexión con Scaffolding.....	10
Figura 37	Modelos de contexto de la base de datos	10
Figura 38	Arquitectura modular - Backend	11
Figura 39	Levantamiento del servicio REST en IIS	11
Figura 40	Acceso al sistema	16
Figura 41	Cerrar sesión.....	16
Figura 42	Api Usuarios.....	17
Figura 43	Middleware Vue Router	18
Figura 44	Acceso seguro al sistema – Control de rutas	18
Figura 45	Pantalla principal - Clientes	19
Figura 46	Formulario de ingreso Cliente – Validación de campos	20
Figura 47	Formulario de ingreso Cliente.....	20
Figura 48	Api Clientes.....	21
Figura 49	Filtros para búsqueda de información global - Clientes.....	21
Figura 50	Búsqueda flexible - Clientes.....	22
Figura 51	Registro de Cliente	23
Figura 52	Consulta - Cliente registrado.....	23
Figura 53	Validación de ingreso – Cliente existente	23
Figura 54	Edición de Cliente	24
Figura 55	Consulta - Cliente actualizado.....	25
Figura 56	Eliminación de Cliente	26
Figura 57	Notificación – Cliente eliminado	26
Figura 58	Eliminación de cliente con registros en otras tablas.....	26
Figura 59	Notificación error de eliminación – Cliente.....	26
Figura 60	Pantalla principal - Ventas	28
Figura 61	Detalle de Venta – Servicios y Repuestos.....	28
Figura 62	Api Ventas.....	29
Figura 63	Filtros para búsqueda de información global - Ventas.....	30

Figura 64 Búsqueda flexible – Detalle Ventas	30
Figura 65 Búsqueda flexible – Detalle Ventas	30
Figura 66 Registro de garantía - Ventas	31
Figura 67 Lista retorno de vehículos por garantía	32
Figura 68 Ventana emergente: Parámetros para el Reporte de Ventas por Cliente.....	33
Figura 69 Reporte de Ventas por Cliente.....	33
Figura 70 Ventana emergente: Parámetros para el Reporte de General de Ventas	33
Figura 71 Reporte General de Ventas.....	34
Figura 72 Cliente sin registros de ventas en las fechas especificadas – Reporte Ventas por Cliente.....	34
Figura 73 Pantalla principal – Inventario	39
Figura 74 Detalle de Inventario – Kardex: Salidas.....	39
Figura 75 Detalle de Inventario – Kardex: Entradas	39
Figura 76 Filtros para búsqueda de información global – Artículos	41
Figura 77 Búsqueda flexible – Artículos	41
Figura 78 Búsqueda flexible – Detalle Kardex.....	41
Figura 79 Ventana emergente: Parámetros para el Reporte de Existencias por Categoría – Artículos.....	42
Figura 80 Reporte de Existencias por Categoría – Artículos.....	43
Figura 81 Ventana emergente: Parámetros para el Reporte General Existencias – Artículos	43
Figura 82 Reporte General de Existencias– Artículos.....	44
Figura 83 Pantalla principal – Órdenes de Trabajo	45
Figura 84 Formulario de ingreso Orden de Trabajo – Parte 1	45
Figura 85 Formulario de ingreso Orden de Trabajo – Parte 2	46
Figura 86 Api Órdenes de Trabajo	47
Figura 87 Filtros para búsqueda de información global – Órdenes de Trabajo.....	47
Figura 88 Registro de Orden de Trabajo – Inventariado de vehículo.....	48
Figura 89 Registro de Orden de Trabajo – Servicios requeridos.....	48
Figura 90 Registro de Orden de Trabajo – Daños y observaciones.....	49
Figura 91 Edición Orden de Trabajo	50
Figura 92 Consulta – Orden de Trabajo actualizada.....	50
Figura 93 Eliminación Orden de Trabajo	51
Figura 94 Notificación – Orden de Trabajo eliminada.....	51
Figura 95 Página Principal del Aplicativo.....	60

Figura 96 Proceso ETL Diario: Carga de Datos en el Sistema de Consultas	61
Figura 97 Configuración para la Ejecución Diaria del Proceso ETL de Carga de Datos	61
Figura 98 Supervisión del Proceso en Ejecución: Administrador de Tareas.....	61
Figura 99 Acceso al Sistema SACI por Periodo.....	64
Figura 100 Interfaz Gestión de Clientes - Sistema SACI.....	65
Figura 101 Interfaz Gestión de Artículos- Sistema SACI	65
Figura 102 Interfaz Gestión de Inventario - Sistema SACI.....	65
Figura 103 Interfaz Gestión de Ventas - Sistema SACI.....	66
Figura 104 Reportes Generales - Sistema SACI.....	66
Figura 105 Órdenes de Trabajo Manuales.....	67
Figura 106 Menú de navegación.....	67
Figura 107 Páginas de Navegación - Menú desplegable	68

ÍNDICE DE TABLAS

Tabla 1 Comparación Gestores de Bases de Datos Relacionales	16
Tabla 2 Product Backlog General	1
Tabla 3 Sprint Backlog 1	5
Tabla 4 Historia de usuario HU-1	5
Tabla 5 Historia de usuario HU-2	7
Tabla 6 Historia de usuario HU-3	8
Tabla 7 Historia de usuario HU-4	9
Tabla 8 Historia de usuario HU-5	10
Tabla 9 Pruebas unitarias – Sprint 1	12
Tabla 10 Reunión sprint 1	13
Tabla 11 Sprint Backlog 2	14
Tabla 12 Historia de usuario HU-6	15
Tabla 13 Historia de usuario HU-7	16
Tabla 14 Historia de usuario HU-8	18
Tabla 15 Historia de usuario HU-9	20
Tabla 16 Historia de usuario HU-10	22
Tabla 17 Historia de usuario HU-11	23
Tabla 18 Historia de usuario HU-12	25
Tabla 19 Historia de usuario HU-13	27
Tabla 20 Historia de usuario HU-14	28
Tabla 21 Historia de usuario HU-15	31
Tabla 22 Historia de usuario HU-16	32
Tabla 23 Pruebas unitarias – Sprint 2	34
Tabla 24 Reunión sprint 2	36
Tabla 25 Sprint Backlog 3	37
Tabla 26 Historia de usuario HU-17	38
Tabla 27 Historia de usuario HU-18	39
Tabla 28 Historia de usuario HU-19	41
Tabla 29 Historia de usuario HU-20	44
Tabla 30 Historia de usuario HU-21	46
Tabla 31 Historia de usuario HU-22	47

Tabla 32 Historia de usuario HU-23	49
Tabla 33 Historia de usuario HU-24	50
Tabla 34 Pruebas unitarias – Sprint 3	51
Tabla 35 Reunión sprint 3	52
Tabla 36 Caso de Prueba #1 PI - Orden de Trabajo.....	54
Tabla 37 Caso de Prueba #1 PIN - Ventas.....	56
Tabla 38 Caso de Prueba #2 PIN - Inventario	57

CAPÍTULO I: INTRODUCCIÓN

1.1 Justificación

A medida que la tecnología evoluciona, pequeñas y medianas empresas se enfrentan al reto de mantenerse competitivas en el mercado. Las empresas dedicadas al servicio de mantenimiento vehicular no son la excepción, por lo que se han visto en la necesidad de innovar e implementar sistemas de información que permitan mejorar sus procesos operativos y agregar valor a sus servicios con relación a la competencia.

No obstante, resulta de gran importancia que los sistemas de información implementados en los talleres mecánicos permitan un acceso ágil a la información relevante para los administradores, incluyendo el historial de mantenimiento de los vehículos, dado que esta característica es crucial para garantizar un servicio de calidad.

La necesidad de contar con un acceso eficiente a la información propuesta ha motivado la realización del presente proyecto. A través de este, se busca desarrollar un aplicativo web de consulta que facilite al administrador del taller mecánico la gestión de la información proporcionada por el sistema actual de manera efectiva y útil.

1.2 Antecedentes

La empresa Tecnicentro Ibarra comenzó sus actividades como un proyecto familiar hace 20 años, liderado por padre e hijos, con la principal meta de brindar todo tipo de servicios automotrices en la ciudad de Ibarra. Con el paso del tiempo, el dueño optó por reestructurar la empresa y dividirla entre sus hijos.

Esta reestructuración resultó en la formación de tres empresas independientes: una especializada en servicios de grúas, otra enfocada en alineación, balanceo y mecánica exprés, y finalmente, un taller mecánico dedicado al mantenimiento y reparación vehicular, fundado en 2018. Así surge TecniMecánica Ibarra.

1.3 Planteamiento del problema

Actualmente, el taller mecánico automotriz TecniMecánica Ibarra, empresa pequeña conformada por un jefe de taller, cuatro técnicos y una asistente administrativa, cuenta con un sistema contable llamado SACI para la gestión de clientes, inventario, y ventas del establecimiento. Este sistema fue desarrollado en Visual Basic 6.0 y Oracle Database 11g Express Edition.

Desde 2018, el personal de oficina ha manejado el sistema sin mayores complicaciones, debido al mantenimiento regular que se le ha dado. Sin embargo, surge una problemática relevante: El jefe de taller, quien es el responsable de dirigir y tomar decisiones sobre los servicios, presenta dificultades en el acceso a información relevante, como el historial de mantenimientos y la gestión de suministros del taller. Esto se debe a que la usabilidad del sistema presenta desafíos en términos de navegación, lo que resulta poco intuitivo y conlleva una pérdida de tiempo significativa en la gestión de actividades del taller.

En función de esta problemática se plantea la siguiente pregunta principal de investigación:

- ¿Cuál sería la forma más adecuada para integrar la información del sistema actual con un aplicativo web que se enfoque exclusivamente en la gestión de consultas?

Y las siguientes preguntas secundarias:

- ¿Cuál es la información más relevante y necesaria para ser consultada?
- ¿Cuáles son las principales dificultades que enfrenta el jefe de taller al consultar información en el sistema actual?

1.4 Objetivo General

Desarrollar una aplicación web que permita gestionar el acceso eficiente a la información relevante del taller mecánico, para un buen control y seguimiento de los servicios de mantenimiento vehicular y la gestión de recursos, permitiendo que el jefe de taller tome decisiones informadas en menor tiempo.

1.5 Objetivos Específicos

- 1 Identificar las necesidades del usuario en cuanto a la automatización de la búsqueda de información estratégica del taller mecánico.
- 2 Diseñar un prototipo que facilite al usuario a encontrar la información que necesita de manera rápida y precisa.
- 3 Desarrollar un aplicativo web de consultas basado en una plataforma intuitiva, efectiva, y segura.
- 4 Probar las funcionalidades del aplicativo de acuerdo con los requisitos y especificaciones definidos por el usuario.

1.6 Alcance

El presente proyecto contempla el análisis, diseño y desarrollo de un aplicativo web enfocado a la gestión de consultas para el control del servicio de mantenimiento vehicular en el taller TecniMecánica Ibarra.

Para ello, se llevará a cabo un análisis de las necesidades y requerimientos del taller mecánico, y se diseñará una solución enfocada en la gestión de consultas útiles y relevantes del servicio automotriz. Esta solución se integrará con la base de datos del sistema actual utilizado por el taller para acceder y extraer la información necesaria.

El proyecto no incluye la instalación e implementación del sistema en un entorno de producción.

CAPÍTULO II: MARCO TEÓRICO

2.1 Bases teóricas del servicio técnico automotriz

2.1.1 Sector automotriz

El sector automotriz engloba la industria que se enfoca en el diseño y fabricación de vehículos motorizados. Es reconocido como uno de los sectores más innovadores y globalizados, cuyo impacto económico en los países es notable, gracias a la diversidad de productos y a las mejoras en los procesos productivos que presenta (Sánchez, 2006).

En el Ecuador, el sector automotriz desempeña un papel fundamental en la economía del país, debido a que su desarrollo aporta ingresos fiscales para el estado a través de aranceles e impuestos y genera fuentes de empleo, tanto en los procesos de producción, como en las actividades comerciales y de servicio que lo rodean (Carrillo, 2009).

De acuerdo con el informe de la Asociación de Empresas Automotrices del Ecuador (AEADE), el mercado de vehículos del país experimentó una recuperación formidable en el año 2022. Con un crecimiento de más del 24% en comparación al año 2021, inclusive superando las cifras prepandémicas.

2.1.2 Taller mecánico

Pérez (2022), define al taller mecánico como un establecimiento especializado en la reparación de vehículos, que dispone de un equipo de profesionales capacitados y, en la mayoría de los casos, con experiencia previa en el campo. Cuenta con equipos, maquinaria y herramientas especializadas para llevar a cabo una variedad de servicios, desde mantenimiento y reparación, hasta la venta de repuestos (pág. 1).

Cabe destacar que un taller mecánico debe tener como prioridad la cuidadosa planificación de las actividades de trabajo y limpieza de las instalaciones. Debido a que estos puntos son de

vital importancia para garantizar un servicio de alta calidad y profesionalismo a los clientes (Bonilla, 2023).

2.1.3 Tipos de mantenimiento vehicular

“Los vehículos son máquinas complejas que requieren mantenimientos periódicos para preservar su vida útil o, en otras ocasiones, reparaciones, ya que inevitablemente sufren averías, ya sea por defectos, mal uso, accidentes o desgaste. Este tipo de problemas demandan expertos que se encarguen de realizar reparaciones profesionales y de calidad” (Pérez, 2022, pág. 1).

Mantenimiento preventivo

Revisiones periódicas y programadas aplicables al auto, las cuales permiten un mayor tiempo de vida útil. Dichas revisiones se rigen a las indicaciones del fabricante detalladas en el manual del vehículo, según el tipo de servicio y los intervalos de tiempo designados para las operaciones de mantenimiento (RECOPE, 2020).

Mantenimiento correctivo

RECOPE (2020), describe al mantenimiento correctivo como actividades no planificadas, lo que implica su ejecución, cuando las distintas partes del vehículo dejan de funcionar o comienzan a presentar fallos (pág. 2).

2.1.4 Importancia del mantenimiento preventivo

La importancia de realizar mantenimiento preventivo al vehículo recae en garantizar su óptimo funcionamiento. Normalmente es recomendable realizar revisiones cada 5000 km, con el fin de detectar y abordar tempranamente posibles problemas o desgastes que podrían afectar el funcionamiento del automóvil (Carreño, 2020).

Dicha práctica se realiza en partes principales del vehículo, las cuales experimentan una mayor exposición a desgastes derivados del uso cotidiano, como el sistema de frenos, el tren de transmisión, la suspensión motor, las pastillas de freno, los amortiguadores y fluidos del motor.

Esto contribuye a la seguridad de las personas al minimizar el riesgo de fallas inesperadas (Carreño, 2020).

2.1.5 Control de mantenimiento preventivo

Un establecimiento que preste servicios técnicos de mantenimiento vehicular de calidad se distingue por llevar una documentación organizada de los mantenimientos preventivos realizados a los vehículos de sus clientes, con el objetivo de mantener un adecuado control de estos y llevar un historial, tanto de los servicios efectuados, como de los repuestos utilizados.

2.2 Metodologías de desarrollo de software

Según Pressman (2010), una metodología de desarrollo de software es un marco de trabajo que provee un conjunto de principios, prácticas y herramientas para guiar el desarrollo de software. Su propósito es asistir al equipo de desarrollo en la organización de su trabajo, permitiendo una adecuada gestión de riesgos que garantice la calidad del producto resultante (pág. 44).

Las metodologías se componen de fases, las cuales pueden subdividirse en sub-fases. Estas etapas orientan al equipo de desarrollo en la selección de las técnicas más adecuadas en cada fase del proyecto, así como en la planificación, gestión, control y evaluación de este (Avison & Fitzgerald, 2003).

2.2.1 Tipos de metodología de desarrollo de software

Metodologías tradicionales

Modelo basado en procesos secuenciales predictivos, que tiene por objetivo la obtención de resultados sólidos dentro de un calendario predefinido de plazos extensos. Bajo este enfoque, se estipula que todos los requisitos deben ser definidos al inicio del proyecto, limitando la interacción con el cliente en cuanto a la aceptación de cambios en el producto de software (Cadavid, Martínez, & Vélez, 2013).

Sánchez N. C. (2023), resalta que las metodologías tradicionales demandan de un nivel de burocracia y documentación exhaustiva de procesos, lo cual en varias ocasiones ralentiza los

proyectos de software. Un ejemplo que se ajusta a este modelo es la metodología en cascada o waterfall, reconocida como la más emblemática en este sentido (pág. 23).

Metodologías Ágiles

Las metodologías ágiles surgen como respuesta a las limitaciones de las metodologías tradicionales. Debido a que estas últimas no se ajustan a los cambios constantes que suelen darse en las fases de desarrollo de los proyectos de software. Dicha problemática impulsó la creación de procedimientos que prioricen la participación del cliente, en lugar de enfocarse en procesos rígidos que no son suficientemente productivos (Sánchez N. C., 2023).

Amaro & Valverde (2007), describen a las metodologías ágiles como adaptativas y orientadas a personas. En virtud de que su eje de trabajo se centra en la entrega incremental de versiones del producto que pueden ser consumidas por el cliente, para de esta manera, lograr una evaluación continua de los requisitos y resultados del producto, posibilitando una respuesta ágil ante los cambios que puedan surgir. (pág. 13)

2.2.2 Metodologías ágiles

Ventajas de las Metodologías ágiles

Tal y como se señala en el estudio “Aplicación de metodologías ágiles para la evolución de productos software” (González, 2008), las metodologías ágiles resultan apropiadas en situaciones donde los requisitos emergen y evolucionan rápidamente. En base al presente estudio y el trabajo llevado a cabo por Atlassian (2022), es posible afirmar que dichas metodologías ofrecen varias ventajas:

- Adaptarse a los cambios: Es posible realizar modificaciones sin afectar a la planificación. Esto se debe a que los proyectos se dividen en ciclos cortos, que permiten entregar avances frecuentes al cliente.
- Mejorar la satisfacción del cliente: Las entregas continuas permiten al cliente conocer el progreso del proyecto y proporcionar retroalimentación.

- Disminuir los riesgos: Los ciclos cortos permiten identificar y corregir errores rápidamente, lo que disminuye la dimensión de los riesgos.
- Mejorar la comunicación: La comunicación directa y fluida entre el cliente y el equipo de desarrollo ayuda a evitar malentendidos y errores.
- Eliminar trabajos innecesarios: Solo se redactará la documentación que sea totalmente necesaria y que aporte valor. Esto optimiza el tiempo y los recursos.
- Mejorar continuamente: Las metodologías ágiles se basan en la mejora continua de los procesos y del equipo de desarrollo, garantizando que los proyectos se lleven a cabo de la manera más eficiente y efectiva posible.

Metodologías ágiles más destacadas

En el entorno empresarial actual, la adaptación ágil es clave para el éxito. Entre las metodologías más reconocidas en este contexto se encuentran Extreme Programming y Scrum.

Extreme Programming

Extreme Programming (XP) es una metodología ágil de desarrollo de software que prioriza la excelencia técnica y la entrega de software valioso para los usuarios finales. Se caracteriza por su enfoque iterativo e incremental, donde el desarrollo se divide en ciclos cortos llamados iteraciones. Esto permite al equipo adaptarse a requerimientos cambiantes y centrarse en desarrollar componentes específicos del software en cada iteración (Mendoza, 2020).

Roles de trabajo

De acuerdo con Wells (2009), XP define los siguientes roles:

- Cliente: Representante de los usuarios finales. Es responsable de definir los requisitos del software y proporcionar retroalimentación al equipo de desarrollo.
- Desarrollador: Responsable de desarrollar el software.
- Coach: Desempeña el papel de intermediario entre el cliente y el equipo, aborda desafíos, y garantiza el cumplimiento adecuado de la metodología.

- Tracker: Responsable de recopilar datos sobre el progreso del equipo y comunicarlos al cliente.

Prácticas

Según Cadavid, Martínez y Vélez (2013), XP se basa en cuatro prácticas clave para crear software de alta calidad:

- Programación en pareja: Dos desarrolladores colaboran en un ordenador, uno codifica y el otro revisa, mejorando así la calidad y comunicación.
- Pruebas unitarias: Los desarrolladores realizan pruebas para cada unidad de código para asegurar su correcto funcionamiento.
- Refactorización: Los desarrolladores mejoran el código existente para optimizar su claridad, eficiencia y mantenibilidad.
- Entregas frecuentes: Entrega regular de software en partes pequeñas para satisfacer al cliente y detectar errores rápidamente.

Extreme Programming ofrece ventajas notables, como la entrega frecuente y adaptable de software, alta calidad del código y colaboración continua con el cliente, lo que resulta beneficioso en proyectos pequeños, con plazos ajustados y una sólida experiencia en ingeniería de software. Sin embargo, para aprovechar al máximo estas ventajas, es esencial mantener una comunicación constante con el cliente y un compromiso firme del equipo.

Scrum

Scrum es un marco de trabajo versátil diseñado para gestionar proyectos en entornos de cambio constante. Su enfoque va más allá del desarrollo de software, ya que su objetivo principal es abordar problemas mediante un enfoque iterativo basado en el aprendizaje continuo. Esto se logra a través de "Sprints", periodos de trabajo definidos y cortos que promueven la adaptación ágil y la mejora constante (Schwaber & Sutherland, 2020).

Roles de trabajo

De acuerdo con Drumond (2022), Scrum establece tres roles distintivos en cada equipo de trabajo:

- **Product Owner:** Responsable de maximizar el valor del producto, definiendo el objetivo, gestionando el backlog del producto (lista de tareas) y tomando decisiones que deben ser respetadas por la organización.
- **Desarrolladores:** Responsables de planificar las tareas del Sprint, asegurando que se completen según los estándares y trabajar juntos para alcanzar el objetivo.
- **Scrum master:** Garantiza que el equipo comprenda y aplique Scrum, eliminando obstáculos y facilitando la colaboración con las partes interesadas.

Prácticas

Scrum se basa en tres artefactos formales para tomar decisiones importantes. Estos artefactos representan el trabajo que se ha realizado, lo que permite a los equipos tener transparencia, inspeccionar su progreso y adaptarse según sea necesario (Schwaber & Sutherland, 2020).

Los artefactos son:

- **Backlog del producto:** Lista dinámica que ordena las tareas para crear el producto por prioridad. Se actualiza constantemente y los elementos deben ser aprobados en la "Planificación de Sprint" antes de avanzar.
- **Backlog del sprint:** Lista de tareas en tiempo real para el Sprint, flexible y actualizable, que los desarrolladores deben completar para alcanzar el objetivo.
- **Incremento:** El trabajo completado por el equipo de Scrum durante un sprint.

Scrum ofrece ventajas que podrían ser más adecuadas que la metodología Extreme Programming. Esto se debe a que su principal fortaleza reside en la flexibilidad y adaptabilidad que simplifica la curva de aprendizaje y permite realizar ajustes de manera eficiente. En contraste, XP implica prácticas de ingeniería de software más rigurosas, lo que conlleva a un proceso menos flexible.

Cabe destacar también, que Scrum se enfoca en la obtención de resultados rápidos, lo cual es esencial en proyectos con plazos ajustados como el presente, permitiendo una retroalimentación temprana y la capacidad de efectuar correcciones a medida que avanza el proyecto.

2.2.3 Scrum

Tras un exhaustivo análisis de las metodologías propuestas, Scrum destaca como la elección más apropiada para el presente proyecto. Esto se debe a su capacidad para adaptarse a las necesidades específicas del mismo, su enfoque en la flexibilidad y la colaboración con el cliente, y su énfasis en la participación de todos los miembros del equipo desde el inicio. Esta combinación de factores genera valor, y promueve una retroalimentación constante que mejora la comunicación entre todas las partes involucradas.

Cabe acotar que, si bien XP es una metodología ágil viable, en el contexto del proyecto actual, su énfasis en sólidas prácticas de ingeniería de software y pruebas rigurosas podría llevar a una extensión en el cronograma del proyecto.

Definición de roles

Considerando los participantes del proyecto, se han definido los siguientes roles para la ejecución de las tareas:

- **Product Owner:** El jefe de taller asume el rol de Product Owner. Su responsabilidad es definir y priorizar las características del producto.
- **Desarrolladores:** El estudiante tomará el papel de desarrollador, encargado de la implementación de las funcionalidades del prototipo de software.
- **Scrum Master:** Dado que se trata de un proyecto de integración curricular en el que el estudiante está fuertemente involucrado, este también asumirá el rol de Scrum Master. Facilitará el proceso Scrum, superará obstáculos y asegurará el cumplimiento de la metodología.

- Stakeholders: El tutor guía y el jefe de taller constituirán un elemento fundamental de las partes interesadas, proporcionando dirección y retroalimentación durante todo el proyecto para lograr un desarrollo exitoso.

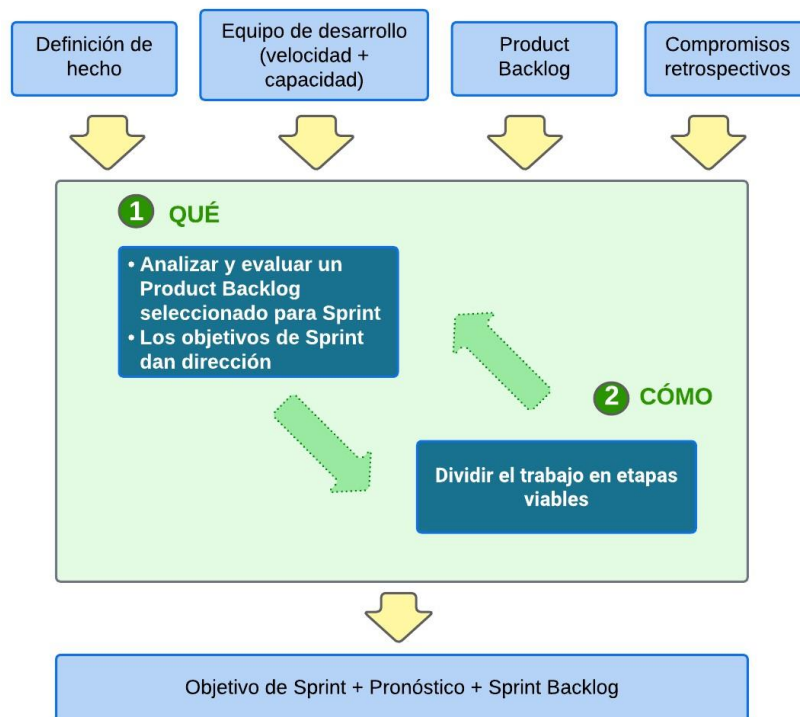
Planificación del trabajo

La planificación del presente proyecto seguirá el enfoque propuesto por Sutherland (2014), que define la planificación de iteraciones en términos del "Qué" y el "Cómo", como se muestra en la Figura 1, y se detalla a continuación:

- QUÉ: El product owner presenta la lista de requisitos priorizados y establece la meta del sprint. El desarrollador analiza la lista, aclara dudas, agrega condiciones y selecciona los requisitos más prioritarios para completar el sprint.
- CÓMO: El desarrollador planifica el sprint, define estrategias para cumplir con los objetivos, y crea el sprint backlog (lista de tareas), donde estima el esfuerzo requerido y se autoasigna tareas.

Al final de cada sprint, se entrega un incremento del prototipo para su revisión y verificación por parte del cliente y las partes interesadas. Este proceso se repite en cada sprint hasta que se complete el desarrollo del prototipo de software según los requisitos establecidos.

Figura 1 Planificación de la iteración



Nota: El grafico representa la planificación de las tareas a realizar en las iteraciones de SCRUM. Tomado de ProyectoSagiles.org (p. 1), por Albaladejo X, 2022.

2.3 Sistemas de gestión de base de datos

Tradicionalmente, la gestión de grandes cantidades de datos se llevaba a cabo mediante bases de datos o bancos de datos, conjunto de datos organizados pertenecientes a un mismo contexto. Esto implicaba el almacenamiento de datos en archivos, y el uso de aplicaciones informáticas con sistemas de índices para una gestión eficaz (Ripoll, 2007).

No obstante, con la creciente complejidad de estos bancos de datos, caracterizada por un aumento en la cantidad de archivos, usuarios, accesos simultáneos y registros, surgieron desafíos significativos para asegurar la consistencia, integridad y eficiencia en el procesamiento de datos (Ripoll, 2007).

Como solución a esta problemática, surgen los Sistemas de Gestión de Bases de Datos, al actuar como intermediarios entre los usuarios y los ficheros, encargándose de tareas complejas, como

el mantenimiento, comprobación y explotación de los datos. Permitiendo que el usuario pierda de vista cómo y dónde se almacenan los datos, al interactuar con ellos a través de un lenguaje conceptual sencillo dentro de un entorno organizado (Tsichritzis & Klug, 1978).

Quintas Ripoll (2007), describe el lenguaje conceptual de un sistema de gestión de bases de datos como la herramienta esencial para llevar a cabo las operaciones de mantenimiento y explotación de los datos (p. 5). Estas operaciones se pueden agrupar en tres clases:

- **Funciones de definición:** permiten crear, modificar o eliminar tablas, registros, campos, índices, etc. También permiten crear relaciones entre tablas y validar campos.
- **Funciones de manipulación:** permiten acceder a los datos mediante operaciones de búsqueda condicionada. Estas operaciones utilizan operadores lógicos para presentar, modificar o eliminar registros.
- **Funciones de administración:** permiten gestionar el acceso a la base de datos, realizar copias de seguridad y recuperación, migrar la base de datos a nuevas versiones, comunicarse con otros sistemas de gestión y crear interfaces gráficas de usuario.

2.3.1 Tipos de gestores de bases de datos

En función de cómo se maneja la información, los gestores de base de datos pueden clasificarse en:

Sistema de gestión de bases de datos relacionales

Un sistema gestor de base de datos relacional se caracteriza por organizar la información en fragmentos identificables, relacionados entre sí mediante tablas estructuradas, destacando por su robustez, resiliencia ante fallos y garantía de propiedades ACID: atomicidad, consistencia, integridad y durabilidad, las cuales aseguran la realización completa de operaciones, la integridad de los datos, su coherencia y persistencia confiable a lo largo del tiempo (Arregui, 2022).

Sistema de gestión de bases de datos no relacionales

Un sistema gestor de base de datos no relacional se caracteriza por manejar grandes volúmenes de datos, ofreciendo alta escalabilidad. A diferencia de las bases de datos relacionales, almacena información en documentos en lugar de tablas, y no garantiza las cualidades ACID. Estos gestores resultan útiles para gestionar datos no estructurados o cuando no se conoce previamente el tipo de datos que se almacenará, y generalmente carecen de identificadores para relacionar conjuntos de datos (Arregui, 2022).

2.3.2 Sistema de gestión de base de dato relacional

En el caso de estudio del taller mecánico, el sistema contable actual que maneja la empresa trabaja con un sistema gestor de base de datos relacional. Esta elección se fundamenta en la necesidad de asegurar las propiedades ACID en las transacciones que se realizan, especialmente en el proceso de facturación, donde se registran los servicios y repuestos necesarios para el vehículo.

Es relevante destacar que el historial de mantenimientos de los vehículos se gestiona en función de las facturas emitidas en cada mantenimiento. A partir de esta información, es posible seguir los registros de reparaciones y repuestos utilizados en un vehículo durante su permanencia en el taller mecánico.

En este contexto, se justifica la elección de un sistema gestor de base de datos relacional, debido a su capacidad para mantener los datos organizados y garantizar su integridad, facilitando la recuperación y consultas flexibles. Esto es esencial para el desarrollo del aplicativo web de consulta que se llevará a cabo, ya que este hará uso de una base de datos relacional que extraerá información de las tablas importantes de la base de datos actual de la empresa.

Sistema de gestión de base de datos manejado por la empresa

El sistema contable del taller mecánico utiliza Oracle Database XE debido a su sólido soporte para transacciones ACID y su adaptabilidad para sistemas de alto rendimiento y alta disponibilidad, en comparación con otros gestores, como se muestra en la Tabla 1.

Tabla 1 Comparación Gestores de Bases de Datos Relacionales

Parámetros	Oracle Database	MySQL	PostgreSQL	Microsoft SQL Server
Desarrollador	Oracle Corporation	Oracle Corporation	Comunidad de código abierto	Microsoft Corporation
Año de Lanzamiento	Década de 1970	1994	Década de 1980	Década de 1980
Escalabilidad	Muy alta	Alta	Alta	Alta
Lenguaje de Consulta	SQL, PL/SQL	SQL	SQL	SQL, T-SQL
Uso Común	Aplicaciones empresariales críticas que necesitan de alta disponibilidad y rendimiento	Aplicaciones web, comercio electrónico	sistemas de información geográfica, análisis de datos	Aplicaciones empresariales, sistemas de gestión de contenido
Soporte Transacciones ACID	Fuerte soporte, con énfasis en integridad y confiabilidad de los datos	Sólido soporte, adecuado para aplicaciones web	Soporte sólido, especialmente personalizable	Fuerte soporte, ideal para aplicaciones empresariales

Nota: Esta tabla muestra aspectos específicos de comparación de los gestores de bases de datos más populares actualmente.

2.3.3 Oracle Database Express Edition 11g

Oracle Database 11g Express Edition es una solución de gestión de bases de datos que se ha consolidado como una opción adecuada en TecniMecánica Ibarra debido a su capacidad para satisfacer las necesidades específicas del establecimiento.

TecniMecánica Ibarra, como una empresa pequeña, requiere de una base de datos de tamaño mediano y fácil de usar para operaciones cotidianas. Su capacidad para manejar hasta 11 GB de datos de usuario y permitir hasta 20 conexiones simultáneas la convierte en una solución

robusta para gestionar información relacionada con clientes, vehículos, inventario y servicios, entre otros aspectos fundamentales del negocio (Oracle, 2023).

Además de su capacidad técnica, Oracle Database XE 11g sobresale por su facilidad de administración. Su soporte para transacciones, seguridad de datos y alta disponibilidad garantiza la integridad y confidencialidad de la información crítica del taller. Esta solución también se beneficia de una robusta infraestructura y capacidad de expansión, lo que permite al taller mecánico desarrollar y mantener aplicaciones de manera eficiente y económica (Oracle, 2023).

2.4 Herramientas de administración de bases de datos

Rodríguez (2023), señala que las herramientas de administración de bases de datos son tecnologías diseñadas para simplificar la gestión de bases de datos. Proporcionan interfaces de usuario que facilitan a los administradores y desarrolladores crear, modificar y supervisar la estructura de la base de datos. Además de gestionar la autenticación y autorización de usuarios, permitiendo un control preciso de los permisos de acceso y mejorando el rendimiento del sistema (p. 1).

2.4.1 Toad for Oracle Base Edition

Toad for Oracle ha destacado como la herramienta comercial de gestión y desarrollo de Oracle número uno durante más de dos décadas, gracias a su capacidad para simplificar el flujo de trabajo de los administradores de bases de datos (Toad, 2023).

Ofrece una amplia gama de características, que incluyen la creación de programas de base de datos con soporte para scripts PL/SQL y SQL, comparación de esquemas, gestión de usuarios, auditoría, carga de datos, y generación de informes. Su interfaz de usuario intuitiva y fácil de usar garantiza una administración eficaz de bases de datos (Toad for Oracle, 2020).

Esta potente herramienta es utilizada por la persona que realiza mantenimientos al sistema de TecniMecánica Ibarra, para una amplia variedad de tareas de gestión de bases de datos, que

incluye actividades como crear, modificar, eliminar y consultar objetos de las bases de datos, gestionar usuarios y permisos, y realizar copias de seguridad periódicas y restauraciones. Lo subraya su versatilidad y valor en el ámbito empresarial del presente caso de estudio.

2.4.2 Oracle Enterprise Manager

Oracle Enterprise Manager es una plataforma de gestión de Oracle robusta que ofrece una solución integral para administrar todas las implementaciones de Oracle en un solo lugar. Este software simplifica la gestión de los componentes de Oracle al proporcionar una serie de herramientas para la supervisión y administración eficiente de componentes clave, como la administración de esquemas y objetos de base de datos, seguridad de usuarios, y gestión de almacenamiento, que incluye tablespaces (espacios de almacenamiento lógico), datafiles (archivos físicos de datos) y rollback segments (segmentos para deshacer transacciones). Además, brinda un entorno propicio para la programación en PL/SQL (ITD, 2022).

Para empresas como TecniMecánica Ibarra, que operan con Oracle, Oracle Enterprise Manager se convierte en una herramienta esencial para administrar las bases de datos que impulsan el funcionamiento del negocio, y, por ende, se trabaja con ella (Vanting, 2020).

2.5 Lenguajes de programación de alto nivel

UNAM (2023), define a los lenguajes de programación como herramientas que permiten desarrollar programas mediante algoritmos, actuando como intermediario entre humano y máquina. Estos lenguajes usan instrucciones y símbolos similares al lenguaje humano, los cuales otorgan significado a sus componentes y expresiones. Mediante programas especiales compiladores e intérpretes, el código se traduce a lenguaje de máquina, facilitando la comunicación humano-computadora (p. 1).

2.5.1 PL/SQL

PL/SQL es un lenguaje de procedimiento desarrollado por Oracle, que potencia las capacidades de SQL. Ya que además de permitir la ejecución de sentencias SQL, brinda la

flexibilidad de programar funciones, triggers, procedimientos almacenados y scripts en bloques de código (UNIR, 2021).

La estructura de un bloque PL/SQL consta de tres partes fundamentales: la declaración, donde se definen constantes y variables con la palabra "DECLARE"; la ejecución, que incluye las instrucciones a llevar a cabo y comienza con "BEGIN" y finaliza con "END"; y el apartado de excepciones, encargado de manejar errores durante el proceso, usando la palabra "EXCEPTION" (Junta de Andalucía, 2023).

PL/SQL es una tecnología clave para el desarrollo de este proyecto, ya que se requiere extraer información de las tablas principales de la base de datos del sistema contable de TecniMecánica Ibarra. Para lo cual, es necesario programar procedimientos almacenados que alimenten la nueva base de datos que se propondrá para el aplicativo web.

2.5.2 C#

C# es un lenguaje de programación moderno, innovador y de código abierto desarrollado por Microsoft, es reconocido por su enfoque orientado a objetos y su capacidad multiplataforma, lo que lo hace adaptable a diversos sistemas operativos. Su versatilidad se evidencia en el desarrollo de aplicaciones de escritorio, web, y móviles, siendo compatible con una amplia gama de tecnologías y marcos de trabajo, lo que lo convierte en una elección preferida por los desarrolladores (Microsoft, 2023).

Además, C# es ampliamente reconocido como el lenguaje más popular para el desarrollo de aplicaciones Backend utilizando el framework .NET de Microsoft. Esta combinación ofrece una sólida plataforma para crear aplicaciones empresariales, servicios web, API y otros sistemas de Backend robustos y escalables (Campaña, 2022).

La elección de C# en el presente proyecto simplificará la gestión de datos al adoptar una perspectiva orientada a objetos y permitirá la creación de servicios web que faciliten el acceso y la presentación de datos en la interfaz gráfica del aplicativo.

2.5.3 JavaScript

JavaScript es un lenguaje de programación de secuencia de comandos interpretado, es decir, que se ejecuta en el navegador del cliente para que las características específicas de la página web sean funcionales. Es un lenguaje orientado a objetos, lo que significa que los datos y las funciones se agrupan en entidades independientes que contienen propiedades y métodos (Menéndez, 2022).

De acuerdo con el sitio web de Oracle (2022), JavaScript es ampliamente utilizado para crear y controlar contenido dinámico de un sitio web. El script indica al sitio web qué hacer, lo que permite mejorar una página HTML añadiendo interacción del usuario, como animaciones, ayudas de navegación, mostrar u ocultar texto, deslizar imágenes, formularios interactivos y demás (p. 1).

El presente proyecto requiere el uso de JavaScript para implementar características interactivas y dinámicas en el aplicativo web a desarrollar, puesto que es un lenguaje de programación versátil, compatible con todos los navegadores web modernos, y cuenta con una gran cantidad de marcos y bibliotecas que facilitan el desarrollo de aplicaciones web de manera eficiente y sencilla.

2.5.4 TypeScript

TypeScript es un lenguaje de programación fuertemente tipado que se basa en JavaScript, y le brinda características adicionales para escribir código con menos errores, mayor simplicidad, coherencia y facilidad de prueba, resultando en un código más limpio y sólido (TypeScript, 2023).

Lo que significa que TypeScript extiende a JavaScript al permitir la especificación opcional de tipos de datos para variables y propiedades, brindando la posibilidad de emplear herramientas de desarrollo avanzadas que permitan detectar y analizar errores de código con el fin de garantizar la calidad de este (Chacón, 2021).

Una vez analizados JavaScript y TypeScript, se considera apropiado utilizar ambos lenguajes de programación para manejar la interacción y lógica del aplicativo, ya que su combinación garantiza calidad y mantenibilidad de código al introducir tipado estático. Lo cual será beneficioso para el mantenimiento de una aplicación que se espera que perdure y evolucione con el tiempo.

2.6 Aplicación web

Las aplicaciones web son programas informáticos que los usuarios pueden utilizar accediendo a un servidor web a través de internet o de una intranet mediante un navegador web. Son ampliamente utilizadas debido a su capacidad para ofrecer un acceso rápido y sencillo a la información sin depender del sistema operativo e instalaciones previas (ICTEA, 2023).

Las aplicaciones web se basan en una arquitectura cliente-servidor. El lado del servidor se encarga de almacenar y recuperar la información, mientras que el lado del cliente muestra la información al usuario. Esto posibilita que las aplicaciones web sean dinámicas, permitiendo a los usuarios interactuar con ellas a través de formularios, sistemas de gestión de contenido, consultas, etc. (Maluenda, 2020).

2.6.1 Frontend

Reyes (2022), define al Frontend como la capa que se encarga de la parte visual o interfaz del aplicativo web, es decir, lo que el usuario ve y con lo que interactúa. Su función principal es asegurar que todas las acciones del aplicativo web, como hacer clic en botones para acceder a información, funcionen adecuadamente para garantizar la satisfacción del usuario en el manejo de la aplicación (p. 1).

2.6.2 Backend

El Backend es la capa que se encarga de la lógica de la aplicación, se ocupa de la lógica del programa y gestiona la conexión con el servidor para procesar las solicitudes de datos hacia la base de datos. Esta capa es invisible para el usuario, pero es fundamental para el

funcionamiento de la aplicación, ya que se encarga de la recuperación y almacenamiento de los datos, autenticación de usuarios y garantía de la seguridad del sitio (AWS, 2022).

Dado que el proyecto abarca la creación de un aplicativo web, se debe contemplar el desarrollo de ambas capas. El Frontend se centrará en la creación de una interfaz de usuario atractiva que facilite a los usuarios el acceso a la información que se requiera consultar, mientras que el Backend desempeñará un papel crucial al encargarse de recuperar datos de la base de datos de manera eficiente y segura.

2.7 Arquitectura de software

2.7.1 API REST

API REST (Representational State Transfer), es un estilo arquitectónico diseñado para crear servicios web. Se fundamenta en principios clave, destacando el concepto de recursos como conjuntos de datos que se representan de manera única mediante URIs (Identificadores Uniformes de Recursos) y métodos HTTP que posibilitan la creación, eliminación, actualización y obtención de información, con el objetivo de establecer una comunicación uniforme y eficaz entre sistemas, proporcionando una base sólida para el intercambio de datos (Fielding, 2000).

2.7.2 API RESTFUL

API RESTFUL hace referencia a la implementación específica de REST para exponer recursos y capacidades de una aplicación a través de una interfaz web. Esta interfaz de programación cumple los principios de diseño REST y es de uso común en el desarrollo web para permitir una comunicación efectiva entre sistemas (Fielding, 2000).

En base a estos conceptos, es fundamental implementar la comunicación entre las capa de Frontend y Backend a través de una API RESTFUL basada en solicitudes y respuestas HTTP estándar, con el fin de facilitar la integración e interoperabilidad entre los componentes de la

aplicación, permitiendo que el Frontend y el Backend operen de manera independiente, mientras mantienen una comunicación eficaz a través de una interfaz estandarizada.

2.8 Herramientas de desarrollo web

Las herramientas de desarrollo web son programas informáticos diseñados para asistir a los desarrolladores a crear, depurar, y mantener aplicaciones web. Estas herramientas incluyen entornos de desarrollo integrados, editores de código y frameworks, que proporcionan estructuras y bibliotecas que permiten agilizar y simplificar el proceso de desarrollo de software. Por lo cual, son esenciales para mejorar la productividad, y aumentar la eficiencia en la construcción de aplicativos web (Meneses, 2023).

2.8.1 Visual Studio 2022 Community Edition

Visual Studio Community Edition es un entorno de desarrollo completo de Microsoft que simplifica la escritura, edición, depuración y compilación de código, al proporcionar herramientas como autocompletado de código, resaltado de sintaxis y depuración visual, lo que facilita el desarrollo de software y mejora la eficiencia de los desarrolladores (Microsoft, 2022). Para el desarrollo del Backend del proyecto, resulta de gran utilidad utilizar esta herramienta, debido a que ofrece un sólido soporte integrado para el lenguaje de programación C# y el marco de desarrollo .NET. proporcionando funcionalidades como autocompletado automático de código, acceso a extensiones NuGet y plantillas predefinidas que establecen una estructura y configuración inicial de aplicaciones web, lo que resulta en un ahorro significativo de tiempo en la construcción de este tipo de aplicativos (Microsoft, 2022).

2.8.2 Visual Studio Code

Visual Studio Code, desarrollado por Microsoft, es un editor de código ligero y extensible que admite más de 100 lenguajes de programación, incluidos JavaScript, y TypeScript. Es una herramienta popular entre los desarrolladores web debido a su interfaz gráfica intuitiva y numerosas características, como el resaltado de sintaxis, autocompletado, depuración, control

de versiones e integración con diversas plataformas, además de una amplia variedad de extensiones para personalizar y mejorar la experiencia de programación (Rifi, 2023).

Visual Studio Code provee un entorno de trabajo óptimo para el desarrollo del Frontend del proyecto, debido a la variedad de extensiones que ofrece para adaptar el entorno de desarrollo de acuerdo con las necesidades del proyecto. Lo cual será provechoso para mejorar la experiencia de desarrollo. Además, destaca como elección adecuada gracias a su capacidad de integración con APIs, lo que asegura una comunicación efectiva con el Backend y una interacción fluida con los datos.

2.8.3 Framework

Según Sepúlved (2023), un framework se define como una aplicación o conjunto de módulos que permiten el desarrollo ágil de aplicativos web mediante la contribución de librerías y funcionalidades ya creadas. Lo que permite que el desarrollador se centre en la resolución de la problemática en lugar de la implementación de funciones que ya han sido previamente desarrolladas (p. 1).

Vue.js

Vue.js es un framework de JavaScript que se distingue por su enfoque simplificado y accesible en comparación con otros frameworks más consolidados como angular y react, lo que lo convierte en una opción adecuada para desarrolladores que buscan una solución moderna y sencilla de implementar. Su capacidad para simplificar la organización del código al combinar HTML, CSS y JavaScript en archivos de componentes, fortalecer la modularidad con componentes reutilizables contribuyen a una estructura coherente y sintaxis declarativa, consolidando su potencial como herramienta esencial en el porvenir del desarrollo web (Párraga, 2023).

Además, incorpora un potente sistema de efectos visuales que realza el diseño de la página, y una variedad de bibliotecas que mejoran la funcionalidad del aplicativo. Todas estas

características justifican la elección de esta herramienta para el proyecto, ya que se cuenta con un tiempo limitado y se requiere de una herramienta versátil que ofrezca funcionalidades modernas y sencillas para construir una aplicación web atractiva y funcional que cumpla con los requisitos del cliente (Párraga, 2023).

ASP.NET Core Web API

ASP.NET Core Web API es un marco de trabajo de Microsoft diseñado para crear servicios web HTTP (API) basados en tecnología .NET, siguiendo el estilo arquitectónico REST. Entre sus características destacan la facilidad de serialización de datos en formatos JSON y XML, un enrutamiento flexible, y la capacidad de garantizar la seguridad mediante autenticación y autorización. Debido a lo mencionado sobresale como un herramienta ideal para crear servicios web escalables y seguros (Microsoft, 2023).

La presente tecnología resulta apta para el proyecto debido a que permite integrar Oracle Database en soluciones ASP.NET Core para construir aplicaciones rest. Esto es posible mediante la configuración de la conexión de Oracle, a través de ingeniería inversa, la cual se basa en el uso del comando Scaffold-DbContext, y herramientas como Oracle dbtools para generar clases de entidad y contexto basadas en un esquema de base de datos existente (Entity Framework Tutorial, 2023).

2.8.4 GitHub

GitHub es una plataforma en línea basada en un sistema de control de versiones distribuido llamado Git, permite a los desarrolladores gestionar los cambios en el código fuente de proyectos a lo largo del tiempo, además de ofrecer herramientas para colaborar en equipo de manera efectiva. Git es escalable y permite trabajar en ramas separadas para nuevas funcionalidades, que luego se fusionan con la rama principal (Rifi, 2023).

El proyecto demanda del uso de esta herramienta para mantener un registro de los cambios que surjan durante la fase de desarrollo del aplicativo web, con el fin de mejorar la gestión de versiones y simplificar la corrección de errores cuando sea necesario.

3.1 Análisis de requerimientos

Durante el proceso de levantamiento de requerimientos funcionales del aplicativo a web a desarrollar, se llevaron a cabo reuniones con el jefe de taller de TecniMecánica Ibarra para comprender a detalle las funcionalidades que formarán parte del aplicativo.

Se determinó que el jefe del taller no es usuario experto tecnológico, por lo que no domina completamente el sistema actual de la empresa, que demanda de una compleja navegación entre varias pantallas. Esto lo obliga a depender del personal de oficina y a dedicar una cantidad significativa de tiempo para acceder a información vital de la empresa.

3.1.1 Especificación funcional del sistema

Se requiere desarrollar una aplicación de consulta que facilite al administrador el acceso al historial de mantenimientos, con el fin de que este pueda coordinar las actividades del taller mecánico de manera más eficiente. Esta información se obtendrá a partir de los datos de ventas, que incluyen los servicios y repuestos utilizados en todos los mantenimientos realizados.

Además, es fundamental tener acceso a información sobre el inventario. Ya que, en numerosas ocasiones, el jefe de taller presenta desafíos al intentar obtener datos actualizados sobre el stock y movimientos de entrada - salida de productos.

También se ha identificado que las órdenes de trabajo, que incluyen el inventariado de vehículos al ingresar al taller y servicios requeridos, se gestionan de forma manual. Para automatizar este proceso, se propone utilizar la aplicación a desarrollar para llevar a cabo su gestión.

Como requisito final, el jefe de taller señaló que el sistema actual carece de un registro de las garantías otorgadas a los vehículos que regresan al taller debido a la persistencia de fallos. Implementar este registro sería beneficioso para evaluar, al final del período, la cantidad de

garantías proporcionadas y así establecer estrategias para mejorar el servicio. Por lo tanto, se agregará una sección específica de registro de garantía en las ventas consultadas, con el propósito de indicar si se otorgó o no garantía en los servicios realizados.

En base a lo mencionado, las funcionalidades principales del producto son las siguientes:

RF0: Ingresar al sistema

RF1: Gestionar Clientes

RF1.1 Consultar Clientes

RF1.1.1 Consultar Clientes por Parámetro.

RF1.1.2 Consultar Clientes General.

RF1.2 Ingresar Clientes

RF1.3 Modificar Clientes

RF1.4 Eliminar Clientes

RF2: Consultar Ventas

RF2.1 Consultar Ventas por Parámetro

RF2.2 Consultar Ventas General

RF2.3 Generar Reportes de Ventas

RF2.3.1 Generar Reporte de Ventas por Cliente

RF2.3.2 Generar Reporte de Ventas General

RF2.4 Modificar Garantía

RF3: Consultar Inventario

RF3.1 Consultar Inventario por Parámetro.

RF3.2 Consultar Inventario General.

RF3.3 Generar Reportes de Existencias

RF3.3.1 Generar Reporte de Existencias por Categoría

RF3.3.2 Generar Reporte de Existencias General

RF4: Gestionar Órdenes de Trabajo

RF4.1 Ingresar Órdenes de Trabajo

RF4.2 Modificar Órdenes de Trabajo

RF4.3 Eliminar Órdenes de Trabajo

RF4.4 Consultar Órdenes de Trabajo

RF4.4.1 Consultar Clientes por Parámetro.

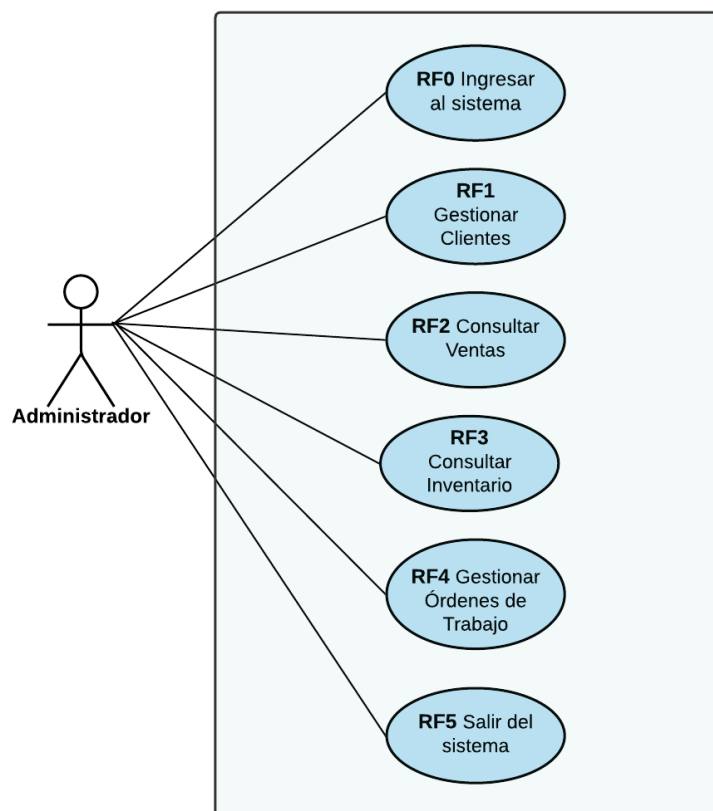
RF4.4.2 Consultar Clientes General.

RF5: Salir del sistema

3.1.2 Casos de uso del sistema

Caso de uso a nivel General

Figura 2 Diagrama de caso de uso a nivel general



Casos de uso a detalle:

RF0 Ingresar al sistema

Figura 3 Diagrama de caso de uso RF0: Ingreso al sistema



Descripción: El sistema posibilitará el acceso al jefe de taller mecánico.

Actor: Administrador.

Flujo Principal:

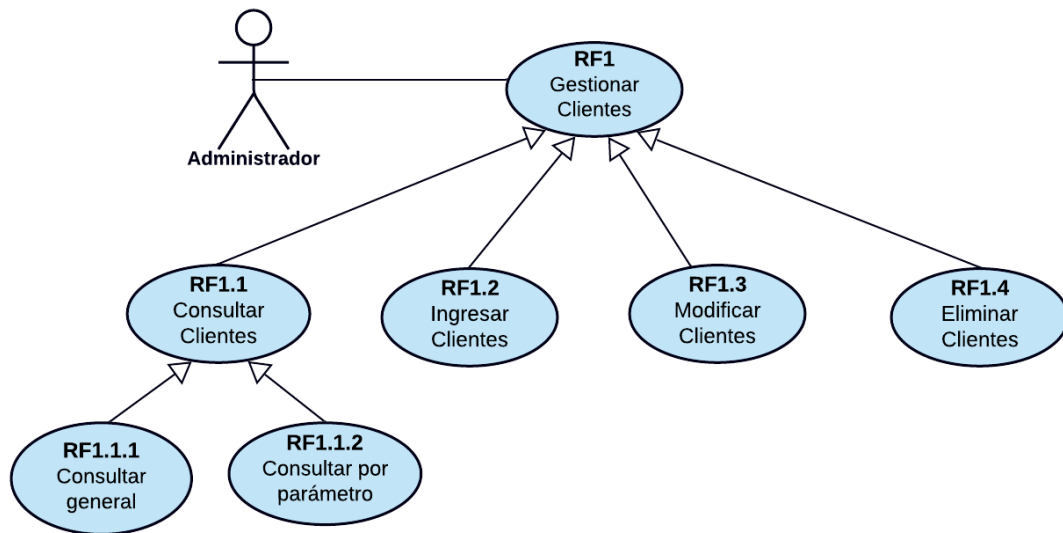
1. El actor carga la página en su navegador.
2. El sistema presenta la interfaz de inicio de sesión.
3. El actor ingresa su nombre de usuario y contraseña.
4. El actor presiona el botón ingresar.
5. El sistema valida en la base de datos la existencia del usuario. (E1)
6. El sistema carga la ventana del menú principal.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

Casos de uso Siguiete Nivel: RF1 Gestionar Clientes

Figura 4 Diagrama de caso de uso RF1: Gestión de Clientes



Casos de uso a detalle:

RF1.1.1 Consultar general

Figura 5 Diagrama de caso de uso RF1.1.1: Consulta general de Clientes



Descripción: El sistema permitirá consultar clientes de todos los periodos.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar clientes del menú principal.
2. El sistema carga la información de los clientes del periodo por defecto 2023. (E1)
3. El sistema presenta la información de todos los clientes del periodo por defecto 2023.
4. El actor selecciona el período de su preferencia.
5. El sistema carga la información de los clientes del periodo seleccionado. (E1)
6. El sistema presenta la información de todos los clientes del periodo seleccionado.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF1.1.2 Consultar por parámetro

Figura 6 Diagrama de caso de uso RF1.1.2: Consulta por parámetro de Clientes



Descripción: El sistema permitirá consultar clientes de todos los períodos utilizando parámetros específicos.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar clientes del menú principal.
2. El sistema carga la información de los clientes del periodo por defecto 2023. (E1)
3. El sistema presenta la información de todos los clientes del periodo por defecto 2023.
4. El actor selecciona el período de su preferencia y, si es necesario, ingresa código, nombre, celular o dirección en la caja de consulta.
5. El sistema carga el/los cliente(s) que coinciden con los parámetros de búsqueda. (E1)
6. El sistema presenta la información de el/los cliente(s) que coinciden con los parámetros de búsqueda.

Flujo Alternativo:

5. Si no existe, ver CU RF1.2 Ingresar Clientes.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF1.2 Ingresar Clientes

Figura 7 Diagrama de caso de uso RF1.2: Ingreso de Clientes



Descripción: El sistema permitirá el ingreso de clientes.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar clientes del menú principal.
2. El sistema carga la información de los clientes. (E1)
3. El sistema presenta la ventana de clientes.
4. El actor presiona el botón nuevo.
5. El actor ingresa los datos del cliente.
6. El actor presiona el botón guardar.
7. El sistema valida en la base de datos el código del cliente. (E1)
8. El sistema almacena en la base de datos la información del cliente. (E1)

Flujo Alternativo:

7. Si existe, ver CU RF1.2 Modificar Clientes, o CU RF1.3 Eliminar Clientes.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF1.3 Modificar Clientes

Figura 8 Diagrama de caso de uso RF1.3: Modificación de Clientes



Descripción: El sistema permitirá la modificación de información de clientes.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar clientes del menú principal.
2. El sistema carga la información de los clientes. (E1)
3. El sistema presenta la ventana de clientes.
4. El actor selecciona el cliente que desea modificar.
5. El actor modifica los campos de nombre, dirección, email o teléfono.
6. El actor presiona el botón guardar.
7. El sistema almacena en la base de datos la información modificada del cliente.
(E1)

Flujo Alternativo:

4. Si no existe, ver CU RF1.1 Ingresar Clientes.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF1.4 Eliminar clientes

Figura 9 Diagrama de caso de uso RF1.4: Eliminación de Clientes



Descripción: El sistema permitirá la eliminación de clientes.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar clientes del menú principal.
2. El sistema carga la información de los clientes. (E1)
3. El sistema presenta la ventana de clientes.
4. El actor selecciona el cliente que desea eliminar y presiona el botón correspondiente.
5. El sistema valida dependencias de registros.
6. El sistema elimina de la base de datos el registro del cliente. (E1)

Flujo Alternativo:

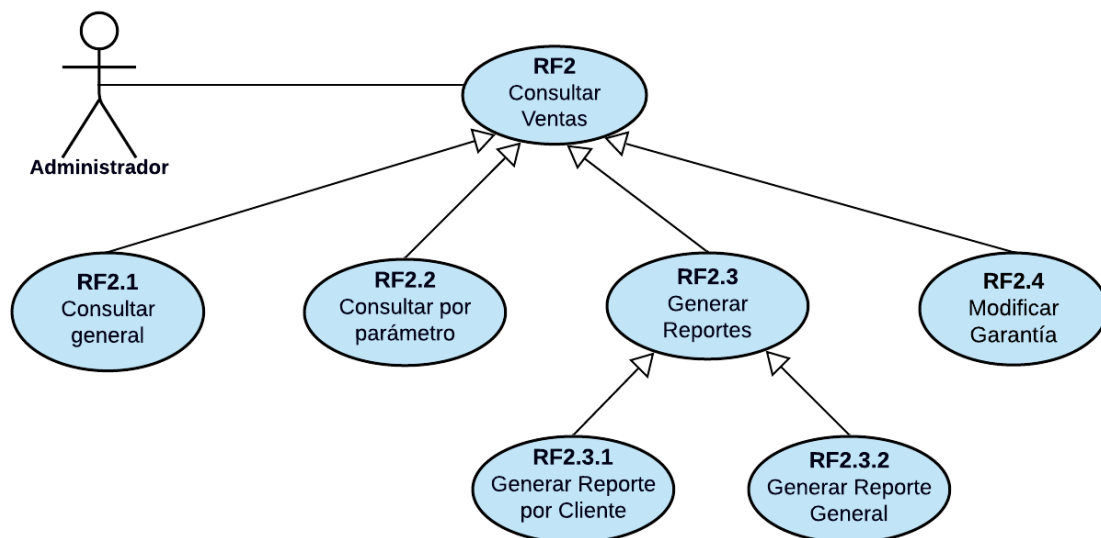
6. Si el cliente tiene dependencias con otras tablas, no es posible eliminarlo y se muestra un mensaje de advertencia

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

Casos de uso Siguiete Nivel: RF2 Consultar Ventas

Figura 10 Diagrama de caso de uso RF2: Consultas de Ventas



Casos de uso a detalle:

RF2.1 Consultar general – Ventas

Figura 11 Diagrama de caso de uso RF2.1: Consulta general de Ventas



Descripción: El sistema permitirá consultar ventas de todos los periodos.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar ventas del menú principal.
2. El sistema carga la información de las ventas del periodo por defecto 2023. (E1)
3. El sistema presenta la información de todas las ventas del periodo por defecto 2023.
4. El actor selecciona el período de su preferencia.
5. El sistema carga la información de las ventas del periodo seleccionado. (E1)
6. El sistema presenta la información de todas las ventas del periodo seleccionado.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF2.2 Consultar por parámetro – Ventas

Figura 12 Diagrama de caso de uso RF2.2: Consulta por parámetro de Ventas



Descripción: El sistema permitirá consultar ventas de todos los períodos utilizando parámetros específicos.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar ventas del menú principal.
2. El sistema carga la información de las ventas del periodo por defecto 2023. (E1)
3. El sistema presenta la información de todas las ventas del periodo por defecto 2023.
4. El actor selecciona el período de su preferencia y, si es necesario, ingresa número de factura, placa del automóvil, código del cliente o fecha en la caja de consulta.
5. El sistema carga el/las factura(s) de venta que coincidan con los parámetros de búsqueda. (E1)
6. El actor presiona el botón de lupa para visualizar la información de una factura en específico.
7. El sistema carga el detalle de la factura seleccionada. (E1)
8. El sistema presenta la información de la factura seleccionada.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF2.3.1 Generar Reporte de Ventas por Cliente

Figura 13 Diagrama de caso de uso RF2.3.1: Reporte de Ventas por Cliente



Descripción: El sistema permitirá generar reportes de ventas por cliente, permitiendo especificar un rango de fechas como parámetro para incluir únicamente las ventas realizadas en ese intervalo de tiempo.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar ventas del menú principal.
2. El sistema carga la información de las ventas del periodo por defecto 2023. (E1)
3. El sistema presenta la información de todas las ventas del periodo por defecto 2023.
4. El actor presiona el botón reportes de la ventana principal de Ventas.
5. El sistema presenta una ventana emergente para seleccionar la fecha inicio, fecha fin, y el código del cliente.
6. El actor presiona el botón generar reporte.
7. El sistema genera y descarga automáticamente en formato PDF el reporte de ventas por cliente correspondiente al intervalo de fechas especificado.

Flujo Alternativo:

7. Si el cliente no registra ventas en el intervalo de fechas dadas, se notificará al usuario a través un alerta emergente.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF2.3.2 Generar Reporte de Ventas General

Figura 14 Diagrama de caso de uso RF2.3.1: Reporte de Ventas General



Descripción: El sistema permitirá generar un reporte general de ventas, permitiendo especificar un rango de fechas como parámetro para incluir únicamente las ventas realizadas en ese intervalo de tiempo.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar ventas del menú principal.
2. El sistema carga la información de las ventas del periodo por defecto 2023. (E1)
3. El sistema presenta la información de todas las ventas del periodo por defecto 2023.
4. El actor presiona el botón reportes de la ventana principal de Ventas.
5. El sistema presenta una ventana emergente para seleccionar la fecha inicio, fecha fin. En la caja de selección del cliente, se establece por defecto el valor "General".
6. El actor presiona el botón generar reporte.
7. El sistema genera y descarga automáticamente en formato PDF el reporte de ventas general correspondiente al intervalo de fechas especificado.

Flujo Alternativo:

7. Si no existen ventas en el intervalo de fechas dadas, se notificará al usuario a través un alerta emergente.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF2.4 Modificar Garantía de Venta

Figura 15 Diagrama de caso de uso RF2.4: Modificación de Garantía de Ventas



Descripción: El sistema permitirá la actualización de ventas, centrándose en la incorporación del manejo de garantías, estableciendo por defecto el valor "No".

Actor: Administrador.

Flujo Principal:

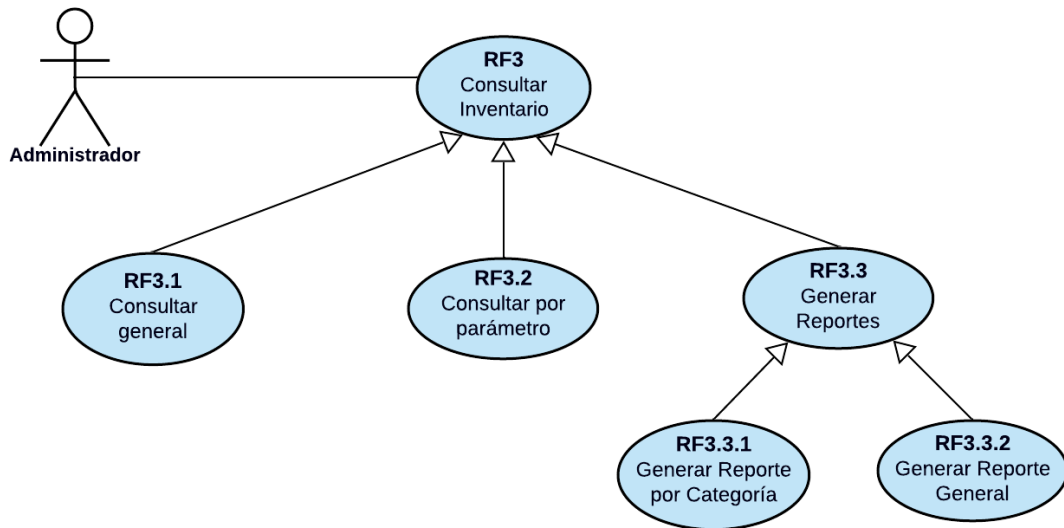
1. El actor selecciona la opción administrar ventas del menú principal.
2. El sistema carga la información de las ventas del periodo por defecto 2023. (E1)
3. El sistema presenta la información de todas las ventas del periodo por defecto 2023.
4. El actor selecciona el período de su preferencia y, si es necesario, ingresa número de factura, placa del automóvil, código del cliente o fecha en la caja de consulta.
5. El sistema carga el/las factura(s) de venta que coincidan con los parámetros de búsqueda. (E1)
6. El actor presiona el botón de lupa para visualizar la información de una factura en específico.
7. El sistema carga el detalle de la factura seleccionada. (E1)
8. El sistema presenta la información de la factura seleccionada.
9. El actor presiona el botón garantía correspondiente a la ventana de detalle de la factura seleccionada.
10. El actor ingresa información sobre la persistencia del fallo en la caja de texto y selecciona la opción "Sí".
11. El sistema almacena en la base de datos las modificaciones correspondientes al manejo de garantía en la venta.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

Casos de uso Siguiete Nivel: RF3 Consultar Inventario

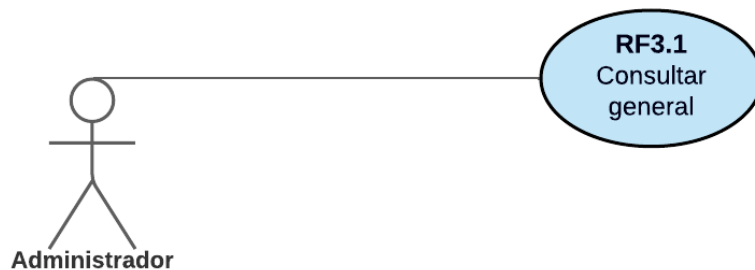
Figura 16 Diagrama de caso de uso RF3: Consultas de Inventario



Casos de uso a detalle:

RF3.1 Consultar general – Inventario

Figura 17 Diagrama de caso de uso RF3.1: Consulta general de Inventario



Descripción: El sistema permitirá consultar inventario de todos los periodos.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar inventario del menú principal.
2. El sistema carga la información de los productos del periodo por defecto 2023. (E1)
3. El sistema presenta la información de todos los productos del periodo por defecto 2023.
4. El actor selecciona el período de su preferencia.
5. El sistema carga la información de los productos del periodo seleccionado. (E1)
6. El sistema presenta la información de todos los productos del periodo seleccionado.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF3.2 Consultar por parámetro – Inventario

Figura 18 Diagrama de caso de uso RF3.2: Consulta por parámetro de Inventario



Descripción: El sistema permitirá consultar inventario de todos los períodos utilizando parámetros específicos.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar inventario del menú principal.
2. El sistema carga la información de los productos del periodo por defecto 2023. (E1)
3. El sistema presenta la información de todos los productos del periodo por defecto 2023.
4. El actor selecciona el período de su preferencia y, si es necesario, ingresa el código del producto o la descripción en la caja de consulta.
5. El sistema carga el/los producto(s) que coincidan con los parámetros de búsqueda. (E1)
6. El actor presiona el botón de lupa para visualizar la información de un producto en específico.
7. El sistema carga el kardex del producto seleccionado. (E1)
8. El sistema presenta la información de kardex del producto seleccionado.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF3.3.1 Generar Reporte de Existencias por Categoría

Figura 19 Diagrama de caso de uso RF3.3.1: Reporte de Existencia Artículos por Categoría



Descripción: El sistema permitirá generar un reporte de existencias (stock) de artículos por categoría, posibilitando la especificación de una fecha de corte para incluir únicamente las existencias registradas hasta ese momento.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar inventario del menú principal.
2. El sistema carga la información de los productos del periodo por defecto 2023. (E1)
3. El sistema presenta la información de todos los productos del periodo por defecto 2023.
4. El actor presiona el botón reportes de la ventana principal de Inventario.
5. El sistema presenta una ventana emergente para seleccionar la fecha de corte y categoría de los artículos.
6. El actor presiona el botón generar reporte.
7. El sistema genera y descarga automáticamente en formato PDF el reporte de existencias hasta la fecha de corte especificada de los artículos pertenecientes a la categoría seleccionada.

Flujo Alternativo:

7. En caso de que no se encuentren existencias registradas para los artículos en la fecha de corte indicada , se notificará al usuario a través un alerta emergente.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF3.3.2 Generar Reporte de Existencias General

Figura 20 Diagrama de caso de uso RF3.3.2: Reporte General de Existencia Artículos



Descripción: El sistema permitirá generar un reporte de existencias (stock) de artículos, posibilitando la especificación de una fecha de corte para incluir únicamente las existencias registradas hasta ese momento.

Actor: Administrador.

Flujo Principal:

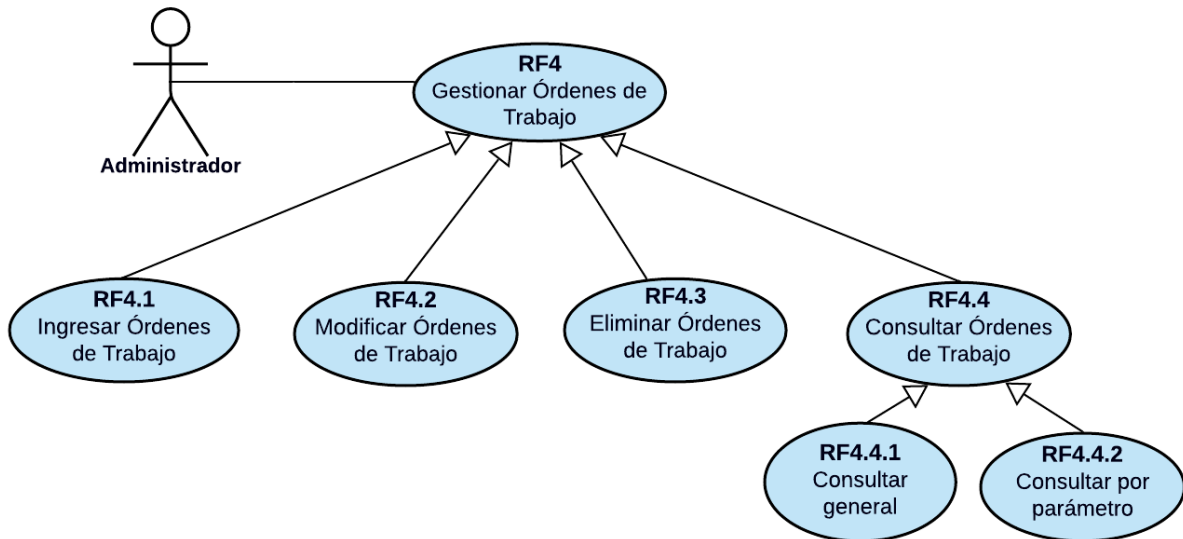
1. El actor selecciona la opción administrar inventario del menú principal.
2. El sistema carga la información de los productos del periodo por defecto 2023. (E1)
3. El sistema presenta la información de todos los productos del periodo por defecto 2023.
4. El actor presiona el botón reportes de la ventana principal de Inventario.
5. El sistema presenta una ventana emergente para seleccionar la fecha de corte. En la caja de selección de la categoría, se establece por defecto el valor "General".
6. El actor presiona el botón generar reporte.
7. El sistema genera y descarga automáticamente en formato PDF el reporte general de existencias hasta la fecha de corte especificada.

Flujo Alternativo:

8. En caso de que no se encuentren existencias registradas para los artículos en la fecha de corte indicada , se notificará al usuario a través un alerta emergente.

Casos de uso Siguiete Nivel: RF4 Gestionar Órdenes de trabajo

Figura 21 Diagrama de caso de uso RF4: Gestión de Órdenes de Trabajo



Casos de uso a detalle:

RF4.1 Ingresar Órdenes de Trabajo

Figura 22 Diagrama de caso de uso RF4.1: Ingreso de Órdenes de Trabajo



Descripción: El sistema permitirá el ingreso de órdenes de trabajo.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar órdenes de trabajo del menú principal.
2. El sistema carga la información de las órdenes de trabajo. (E1)

3. El sistema presenta la ventana de órdenes de trabajo.
4. El actor presiona el botón nuevo.
5. El actor ingresa los datos del cliente.
6. El sistema valida en la base de datos el código del cliente. (E1)
7. El actor ingresa los datos de la orden de trabajo.
8. El actor presiona el botón guardar.
9. El sistema almacena en la base de datos la información de la orden de trabajo. (E1)

Flujo Alternativo:

7. Si no existe, ver CU RF1.1 Crear Clientes.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF4.2 Modificar Órdenes de Trabajo

Figura 23 Diagrama de caso de uso RF4.2: Modificación de Órdenes de Trabajo



Descripción: El sistema permitirá la modificación de órdenes de trabajo.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar órdenes de trabajo del menú principal.
2. El sistema carga la información de las órdenes de trabajo. (E1)
3. El sistema presenta la ventana de órdenes de trabajo.
4. El actor selecciona la orden de trabajo que desea modificar.
5. El actor modifica las casillas del inventariado del vehículo.

6. El actor presiona el botón guardar.
7. El sistema almacena en la base de datos la información modificada de la orden de trabajo. (E1)

Flujo Alternativo:

4. Si no existe, ver CU RF4.1 Ingresar Órdenes de trabajo.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF4.3 Eliminar Órdenes de Trabajo

Figura 24 Diagrama de caso de uso RF4.3: Eliminación de Órdenes de Trabajo



Descripción: El sistema permitirá la eliminación de órdenes de trabajo.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar órdenes de trabajo del menú principal.
2. El sistema carga la información de las órdenes de trabajo. (E1)
3. El sistema presenta la ventana de órdenes de trabajo.
4. El actor selecciona la orden de trabajo que desea eliminar y presiona el botón correspondiente.
5. El sistema elimina de la base de datos el registro de la orden de trabajo. (E1)

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF4.4.1 Consultar general - Órdenes de Trabajo

Figura 25 Diagrama de caso de uso RF4.4.1: Consulta general de Órdenes de Trabajo



Descripción: El sistema permitirá consultar órdenes de trabajo del periodo actual.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar órdenes de trabajo del menú principal.
2. El sistema carga la información de las órdenes de trabajo del periodo actual. (E1)
3. El sistema presenta la información de todas las órdenes de trabajo del periodo actual.

Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

RF4.4.2 Consultar por parámetro - Órdenes de Trabajo

Figura 26 Diagrama de caso de uso RF4.4.2: Consulta por parámetro de Órdenes de Trabajo



Descripción: El sistema permitirá consultar órdenes de trabajo del periodo actual utilizando parámetros específicos.

Actor: Administrador.

Flujo Principal:

1. El actor selecciona la opción administrar órdenes de trabajo del menú principal.
2. El sistema carga la información de los productos del periodo actual. (E1)
3. El sistema presenta la información de todos los productos del periodo actual.
4. El actor ingresa el número de la orden de trabajo, código del cliente, o placa en la caja de consulta.
5. El sistema carga la/las orden(es) de trabajo que coincidan con los parámetros de búsqueda (E1).
6. El actor presiona el botón de lupa para visualizar la información de una orden de trabajo en específico.
7. El sistema carga el detalle de la orden de trabajo seleccionada. (E1)
8. El sistema presenta la información de la orden de trabajo seleccionada.

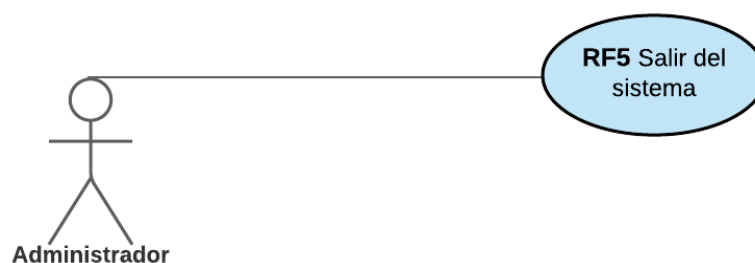
Excepciones:

E1: problemas con la base de datos; comunicarse con el encargado de sistemas.

Casos de uso a detalle:

RF5 Salir del sistema

Figura 27 Diagrama de caso de uso RF5: Salida del sistema



Descripción: El sistema posibilitará el cierre de sesión al jefe de taller mecánico.

Actor: Administrador.

Flujo Principal:

1. El actor presiona el botón de apagado en la esquina inferior derecha de la pantalla del sistema.

2. El sistema finaliza la sesión del usuario.
3. El sistema presenta la ventana de inicio de sesión.

3.2 Configuración del proyecto

Para el desarrollo del sistema web propuesto, conforme a lo establecido en el marco teórico, se emplearán los siguientes recursos:

Recursos Técnicos:

- Backend: .NET Core Web Api con C# para la creación de un api restful conectada a una base de datos Oracle.
- Frontend: Vue.js con JavaScript y TypeScript.
- Navegador web moderno: Microsoft Edge.

Metodología de Trabajo:

- Metodología Scrum y eventos Sprints.
- Comunicación abierta con el cliente (jefe de taller) y pruebas continuas.

3.3 Product Backlog General del Producto

El backlog general del producto, presentado en la Tabla 2, detalla la lista priorizada de funcionalidades que han sido previamente acordadas con el product owner para la finalización del aplicativo.

Tabla 2 *Product Backlog General*

ID Historia	Enunciado de la historia	Estado	Prioridad	Esfuerzo
HU-01	Implementación del modelo de base de datos propuesto.	Propuesto	Alta	Alto
HU-02	Programación de procedimientos almacenados para Extracción y Transformación de datos de la bases de datos fuente.	Propuesto	Alta	Alto
HU-03	Carga de Información en la Nueva Base de Datos - Ejecución de Procedimientos Almacenados.	Propuesto	Alta	Alto
HU-04	Generación de Cadena de Conexión de Base de Datos Oracle en .NET Core	Propuesto	Alta	Medio
HU-05	Desarrollo de API RESTful con Web API	Propuesto	Alta	Medio
HU-06	Diseño de la interfaz de inicio y cierre de sesión	Propuesto	Alta	Medio
HU-07	Funcionalidad de inicio y cierre de sesión	Propuesto	Alta	Medio
HU-08	Diseño de la interfaz de usuario Cliente	Propuesto	Alta	Medio
HU-09	Consulta de Clientes con filtros personalizados	Propuesto	Alta	Alto
HU-10	Registro de Clientes	Propuesto	Media	Alto
HU-11	Modificación de Clientes	Propuesto	Media	Alto
HU-12	Eliminación de Clientes	Propuesto	Media	Alto

HU-13	Diseño de la interfaz de usuario Ventas	Propuesto	Alta	Alto
HU-14	Consulta de ventas con filtros personalizados	Propuesto	Alta	Alto
HU-15	Modificación de Ventas para manejo de garantía	Propuesto	Alta	Medio
HU-16	Generación de reportes de Venta	Propuesto	Alta	Alto
HU-17	Diseño de la interfaz de usuario Inventario	Propuesto	Alta	Alto
HU-18	Consulta de inventario con filtros personalizados	Propuesto	Alta	Alto
HU-19	Generación de reportes de Existencias	Propuesto	Alta	Alto
HU-20	Diseño de la interfaz de usuario Órdenes de trabajo	Propuesto	Media	Alto
HU-21	Consulta de Órdenes de trabajo	Propuesto	Media	Alto
HU-22	Registro de Órdenes de trabajo	Propuesto	Media	Alto
HU-23	Modificación de Órdenes de trabajo	Propuesto	Media	Alto
HU-24	Eliminación de Órdenes de trabajo	Propuesto	Media	Alto

Nota: Esta tabla presenta la lista dinámica priorizada de las tareas a realizar para el desarrollo del producto de software.

3.4 Sprint Planning

El proyecto se ejecutará en tres sprints, considerando los 24 requerimientos totales. Estos sprints se planificarán de la siguiente manera:

- **Sprint 1:** Comprende 5 historias de usuario enfocadas en la implementación del modelo de base de datos, que incluye la extracción, transformación y carga de datos de la base de datos fuente en la nueva base de datos. También llevará a cabo la creación del servicio web.
- **Sprint 2:** Comprende 11 historias de usuario enfocadas en los módulos de usuario, clientes y ventas, bases esenciales de la aplicación, ya que el historial de mantenimientos se encuentra en el detalle de las ventas.
- **Sprint 3:** Comprende 8 historias de usuario enfocadas en el módulo de inventario para consultas, además de la gestión de órdenes de trabajo, siendo esta una funcionalidad adicional del aplicativo.

3.5 Diseño de la base de datos del sistema

3.5.1 Base de datos fuente

La base de datos que maneja el sistema contable del taller mecánico implementa el uso de esquemas en Oracle, como una estructura fundamental para gestionar su información de manera eficiente y organizada. Cada uno de estos esquemas representa un periodo anual específico en la vida de la empresa. En total, se han configurado 6 esquemas, desde el año 2018 hasta el presente 2023, uno para cada año contable.

La base de datos que respalda esta solución posee un conjunto robusto de 330 tablas que gestionan datos relacionados con ventas, compras, clientes, servicios, inventario, bancos, empleados, impuestos, obligaciones fiscales y otros aspectos fundamentales. Esta estructura de tablas proporciona una plataforma sólida para la gestión de información empresarial, lo que facilita la administración de transacciones.

3.5.2 Diseño de la base de datos objetivo

Dado que se requiere desarrollar un aplicativo de consulta que satisfaga los requerimientos establecidos por el jefe del taller mecánico, es fundamental identificar las tablas más importantes de la base de datos fuente con el fin obtener una base de datos simplificada que facilite la consulta de asignación de servicios y repuestos de los mantenimientos de los vehículos, así como la consulta del inventario. Además, de incorporar nuevas tablas que permitan la gestión de órdenes de trabajo.

Para ello se ha realizado un análisis exhaustivo de todas tablas de la base de datos fuente, identificando las más relevantes, que se enumeran a continuación:

1. VEN_MAECLIENTE: Maestro de clientes.
2. INV_MAEARTICULO: Maestro de artículos.
3. VEN_ENCFAC: Cabecera de factura de venta.
4. VEN_DETFAC: Detalle de factura de venta.
5. VEN_VHCSPCF: Especificaciones de vehículos en factura de venta.
6. INV_TRNKARDEX: Manejo de kardex de suministros.
7. VEN_VEHMARMOD: Maestro de Marcas de vehículos.
8. VEN_MAECOLOR: Maestro de Colores de vehículos.

Asimismo, se ha contemplado la adición de las siguientes tablas para el manejo de órdenes de trabajo.

1. SERVICIOS: Catálogo de servicios.
2. INV_VEHICULO: Catálogo de inventario de vehículo.
3. CAB_ORDEN: Cabecera orden de trabajo.
4. DET_ORD_SERVICIOS: Detalle de servicios de orden de trabajo.
5. DET_ORD_INVENTARIO: Detalle de inventario de orden de trabajo.

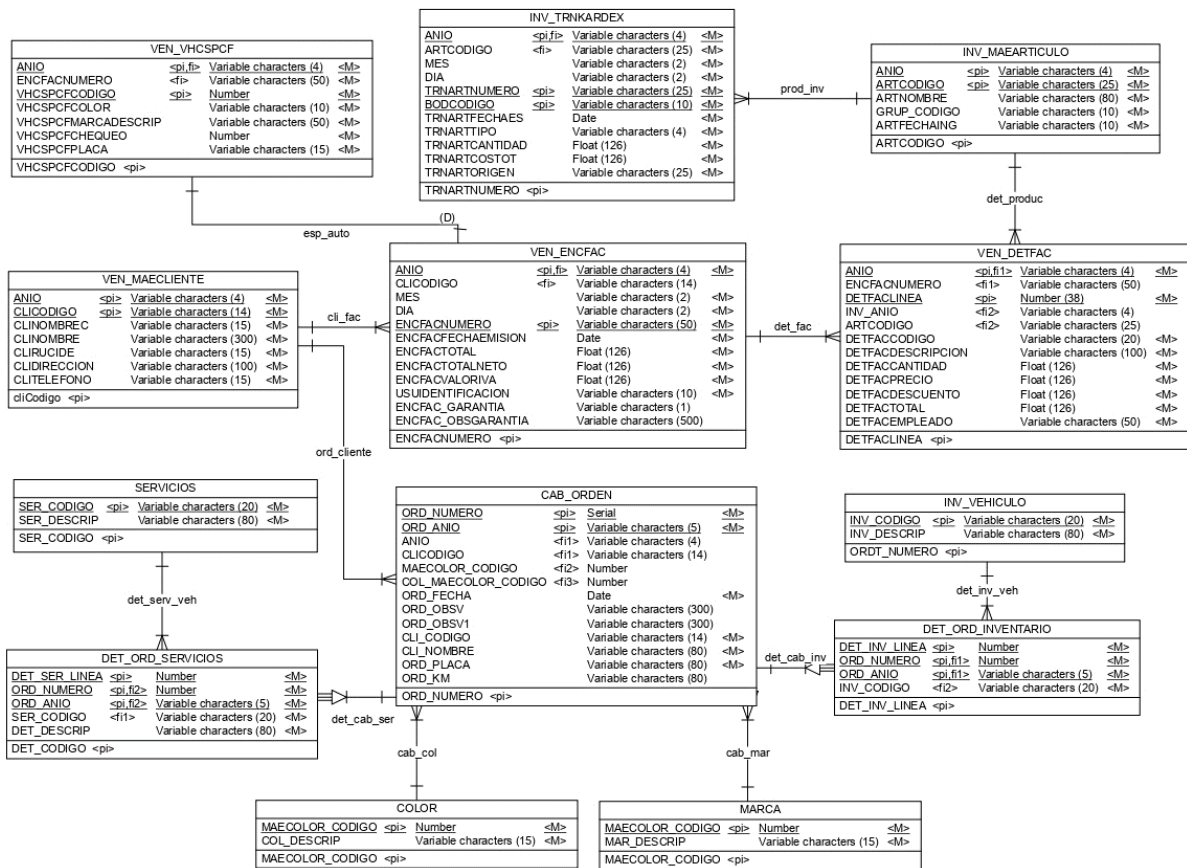
3.5.3 Procedimientos PL/SQL

La carga de información a la base de datos objetivo requiere el desarrollo de procedimientos PL/SQL en Oracle para realizar la extracción, transformación y carga (ETL) de datos provenientes de todos los periodos/esquemas que maneja la empresa en la base de datos fuente. Estos procedimientos cumplirán con los requisitos de consulta del cliente, ya que se encargarán de la extracción eficiente de datos brutos, transformación de datos en un formato coherente y carga en un entorno donde sean comprensibles y útiles para el usuario. La implementación de estos procedimientos garantizará la integridad y consistencia de los datos en el aplicativo web.

3.5.4 Modelos de base de datos

Modelo lógico de base de datos

Figura 28 Diagrama Entidad-Relación: Modelo lógico de base de datos



Modelo físico de base de datos

CAPÍTULO IV: DESARROLLO Y PRUEBAS DEL SISTEMA WEB

4.1 Sprint 1

4.1.1 Sprint Backlog 1

En la Tabla 3 se detallan las tareas a abordar en el primer sprint del desarrollo del aplicativo.

Estos elementos han sido seleccionados del Product Backlog general.

Tabla 3 *Sprint Backlog 1*

SPRINT BACKLOG 1				
ID Historia	Historia	Estado	Prioridad	Esfuerzo
HU-01	Implementación del modelo de base de datos propuesto	En Desarrollo	Alta	Alto
HU-02	Programación de procedimientos almacenados para Extracción y Transformación de datos de las bases de datos fuente	En Desarrollo	Alta	Alto
HU-03	Carga de Información en la Nueva Base de Datos - Ejecución de Procedimientos Almacenados	En Desarrollo	Alta	Alto
HU-04	Generación de Cadena de Conexión de Base de Datos Oracle en .NET Core	En Desarrollo	Alta	Medio
HU-05	Desarrollo de API RESTful con Web API	En Desarrollo	Alta	Medio

4.1.2 Historias de usuario

Tabla 4 *Historia de usuario HU-1*

Historia de Usuario			
Código:	HU-01	Usuario: Administrador	
Nombre de Historia:	Implementación del modelo de base de datos propuesto		
Prioridad:	Alta	Esfuerzo en desarrollo:	Alto
Esfuerzo estimado:	3 días	Sprint asignado:	1

Descripción:

La implementación del modelo de base de datos propuesto en la fase de diseño abarca las siguientes subtareas:

- Establecer un nuevo esquema de base de datos diseñado exclusivamente para albergar la información del aplicativo. Definir un script SQL que facilite la creación de las tablas y objetos necesarios en la base de datos.
- Optimizar el rendimiento de la base de datos mediante la creación de índices apropiados que garanticen la eficacia en la ejecución de consultas.

Observaciones:

Restringir el acceso a la base de datos exclusivamente a usuarios con rol de administrador.

Resultados HU-01

Figura 31 Script de base de datos

```
ScriptBD.sql x
C: > Users > Noelia Quishpe > Desktop > TESIS > ScriptBD.sql
41 /*=====*/
42 /* Table: CAB_ORDEN */
43 /*=====*/
44 create table CAB_ORDEN
45 (
46     ORD_NUMERO          NUMBER(6)          not null,
47     ORD_ANIO            VARCHAR2(5)        not null,
48     ORD_FECHA           DATE                not null,
49     ORD_OBSV            VARCHAR2(300),
50     ORD_OBSV1           VARCHAR2(300),
51     CLI_CODIGO          VARCHAR2(14)       not null,
52     CLI_NOMBRE          VARCHAR2(80)       not null,
53     ORD_PLACA           VARCHAR2(80)       not null,
54     ORD_KM              VARCHAR2(80),
55     VEHARMOD_CODIGO    VARCHAR2(5),
56     MAECOLOR_CODIGO    VARCHAR2(10),
57     constraint PK_CAB_ORDEN primary key (ORD_NUMERO, ORD_ANIO)
58 );
59
60 /*=====*/
61 /* Table: DET_ORD_INVENTARIO */
62 /*=====*/
63 create table DET_ORD_INVENTARIO
64 (
65     DET_INV_LINEA       NUMBER              not null,
66     ORD_NUMERO          NUMBER              not null,
67     ORD_ANIO            VARCHAR2(5)        not null,
68     INV_CODIGO          VARCHAR2(20)       not null,
69     constraint PK_DET_ORD_INVENTARIO primary key (ORD_NUMERO, DET_INV_LINEA, ORD_ANIO)
70 );
71
```

Figura 32 Esquema de base de datos del aplicativo

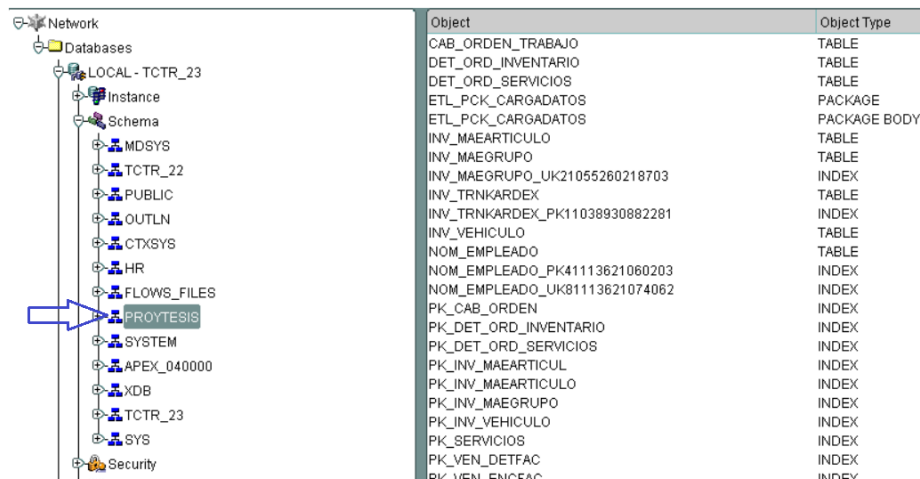


Figura 33 Restricción de usuarios a la base de datos

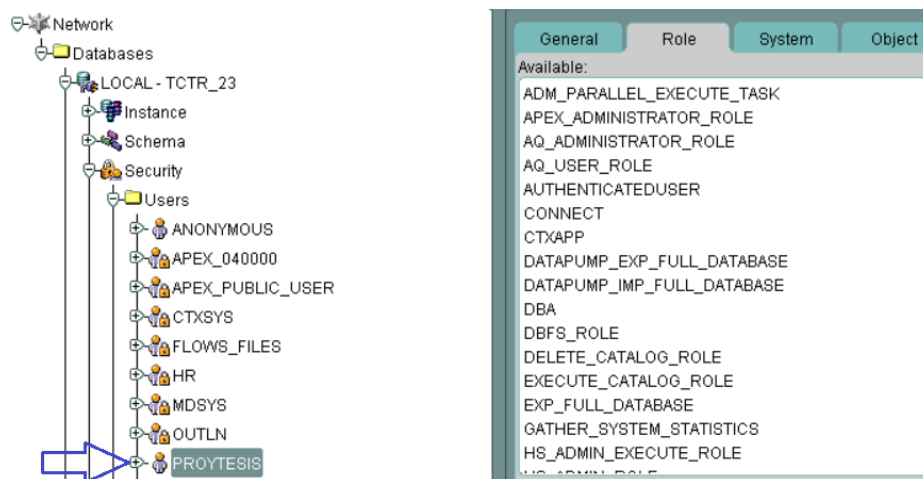


Tabla 5 Historia de usuario HU-2

Historia de Usuario			
Código:	HU-02	Usuario:	Administrador
Nombre de Historia:	Programación de procedimientos almacenados para Extracción y Transformación de datos de la bases de datos fuente.		
Prioridad:	Alta	Esfuerzo en desarrollo:	Alto
Esfuerzo estimado:	3 días	Sprint asignado:	1
Descripción:	La programación de procedimientos almacenados que trabajen con procesos ETL abarca las siguientes subareas:		

- Extracción de datos desde la base de datos fuente. Captura de información por periodo/esquema de las tablas definidas en la fase de diseño.
- Modificación, estructuración, y limpieza de los datos para cumplir con requisitos de consulta.
- Prueba de carga de los datos transformados en una tabla de prueba para verificar el funcionamiento de los procedimientos programados.

Observaciones:

Cargar datos correspondientes a los periodos 2018, 2019, 2020, 2021, 2022 y 2023.

Separar los campos de fecha en año, mes y día para facilitar las consultas posteriores.

Resultados HU-02

Figura 34 Procedimientos almacenados - ETL

```

Edit Package Body : PROYTESIS.ETL_PCK_CARGADATOS - TCTR_23@LOCAL

General
Name: ETL_PCK_CARGADATOS
Schema: PROYTEBIB

Source
Methods
procedure etl_sp_cargaClientesXPeriodo( v_anio varchar2, v_esquema varchar2);
procedure etl_sp_cargaClientes( '2018' 'TCTR_18');
procedure etl_sp_cargaKardexXPeriodo( v_anio varchar2, v_esquema varchar2);
procedure etl_sp_cargaKardex( '2018' 'TCTR_18');
procedure etl_sp_cargaArticuloXPeriodo( v_anio varchar2, v_esquema varchar2);

procedure etl_sp_cargaEncFac
as
begin
  ETL_PCK_CARGADATOS.etl_sp_cargaEncFacXPeriodo('2018','TCTR_18');
  ETL_PCK_CARGADATOS.etl_sp_cargaEncFacXPeriodo('2019','TCTR_19');
  ETL_PCK_CARGADATOS.etl_sp_cargaEncFacXPeriodo('2020','TCTR_20');
  ETL_PCK_CARGADATOS.etl_sp_cargaEncFacXPeriodo('2021','TCTR_21');
  ETL_PCK_CARGADATOS.etl_sp_cargaEncFacXPeriodo('2022','TCTR_22');
  ETL_PCK_CARGADATOS.etl_sp_cargaEncFacXPeriodo('2023','TCTR_23');
  update VEN_ENCFAC
  set mes = to_char(to_date(TO_CHAR(ENCFAC_FECHAEMISION,'dd/mm/yyyy'),'dd/mm/yyyy'),'mm'),
      dia = to_char(to_date(TO_CHAR(ENCFAC_FECHAEMISION,'dd/mm/yyyy'),'dd/mm/yyyy'),'dd');
end etl_sp_cargaEncFac;

procedure etl_sp_cargaDetFacXPeriodo(v_anio varchar2,v_esquema varchar2)

```

Tabla 6 Historia de usuario HU-3

Historia de Usuario			
Código:	HU-03	Usuario: Administrador	
Nombre de Historia:	Carga de Información en la Nueva Base de Datos - Ejecución de Procedimientos Almacenados.		
Prioridad:	Alta	Esfuerzo en desarrollo:	Alto

Esfuerzo estimado:	1 día	Sprint asignado:	1
Descripción:			
Una vez comprobado el funcionamiento de los procedimientos almacenados, se procede a la carga de datos por periodo a la base de datos destino, procurando la gestión de posibles errores que puedan ocurrir durante la ejecución del procedimiento.			
Observaciones:			
Implementar el manejo de excepciones para evitar interrupciones inesperadas en la ejecución de los procedimientos, asegurando la integridad y consistencia de la base de datos.			

Resultados HU-03

Figura 35 Carga de datos a la base de datos destino

```

Carga Datos
1 DECLARE
2     v_anio VARCHAR2(4) := '2023';
3     v_esquema VARCHAR2(30) := 'TCTR_23';
4 BEGIN
5     ETL_PCK_CARGADATOS.etl_sp_cargaClientesXPeriodo(v_anio, v_esquema);
6     ETL_PCK_CARGADATOS.etl_sp_cargaKardexXPeriodo(v_anio, v_esquema);
7     ETL_PCK_CARGADATOS.etl_sp_cargaArticuloXPeriodo(v_anio, v_esquema);
8     ETL_PCK_CARGADATOS.etl_sp_cargaEncFacXPeriodo(v_anio, v_esquema);
9     ETL_PCK_CARGADATOS.etl_sp_cargaDetFacXPeriodo(v_anio, v_esquema);
10    ETL_PCK_CARGADATOS.etl_sp_cargaEspVehXPeriodo(v_anio, v_esquema);
11    ETL_PCK_CARGADATOS.etl_sp_cargaCompania;
12    ETL_PCK_CARGADATOS.etl_sp_cargaEmpleadoXPeriodo(v_anio, v_esquema);
13    ETL_PCK_CARGADATOS.etl_sp_cargaDetFacEmpXPeriodo(v_anio, v_esquema);
14    ETL_PCK_CARGADATOS.etl_sp_cargaGruArtXPeriodo(v_anio, v_esquema);
15    COMMIT; -- Confirmar los cambios en la base de datos
16 EXCEPTION
17     WHEN OTHERS THEN
18         ROLLBACK; -- En caso de error, deshacer los cambios
19         RAISE; -- Re-lanzar la excepción para que sea manejada en niveles superiores
20 END;
21 /

```

Tabla 7 Historia de usuario HU-4

Historia de Usuario			
Código:	HU-04	Usuario:	Administrador
Nombre de Historia:	Generación de Cadena de Conexión de Base de Datos Oracle en .NET Core.		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	1
Descripción:			
La generación de la cadena de conexión de base de datos abarca las siguientes subtareas:			
<ul style="list-style-type: none"> - Instalación de herramientas Oracle Entity Framework Core. 			

- Uso del comando scaffold para automatizar la creación de clases de modelo basadas en las tablas de la base de datos existente, y la generación del contexto de base de datos que representa la conexión y la estructura de base de datos.

Observaciones:

Revisar y configurar las clases generadas, asegurando una definición correcta de las claves foráneas en el modelo de datos.

Resultados HU-04

Figura 36 Generación de la cadena de conexión con Scaffolding

```
Package Manager Console
Package source: All Default project: BackProyTesis
PM>
PM> Scaffold-DbContext "User Id=PROYTESIS;Password=PROYTESIS;Data Source=LAPTOP-D9JHB33B;"
Oracle.EntityFrameworkCore -Tables INV_MAEARTICULO, INV_TRNKARDEX, SEG_MAEUSUARIO,
VEN_MAECLIENTE, VEN_DETFACT, VEN_ENCFAC, VEN_VHCSPCF, SEG_MAECOMPANIA, NOM_EMPLEADO,
VEN_DETFACEMP -outputDir Models -Context contexto -Force
```

Figura 37 Modelos de contexto de la base de datos

```
modelBuilder.Entity<SegMaeusuario>(entity =>
{
    entity.HasKey(e => e.Usidentificacion).HasName("SEG_MAEUSUARIO_PK");
    entity.ToTable("SEG_MAEUSUARIO");
    entity.Property(e => e.Usidentificacion)
        .HasMaxLength(10)
        .IsUnicode(false)
        .HasColumnName("USIDENTIFICACION");
    entity.Property(e => e.BodCodigo)
        .HasMaxLength(10)
        .IsUnicode(false)
        .HasColumnName("BOD_CODIGO");
    entity.Property(e => e.Comcodigo)
        .HasMaxLength(2)
        .IsUnicode(false)
        .HasColumnName("COMCODIGO");
    entity.Property(e => e.Usuapellido)
        .HasMaxLength(15);
});
```

Tabla 8 Historia de usuario HU-5

Historia de Usuario			
Código:	HU-05	Usuario: Administrador	
Nombre de Historia:	Desarrollo de API RESTful con Web API		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	3 días	Sprint asignado:	1
Descripción:	El desarrollo del servicio RESTful abarca las siguientes subtareas:		

- Implementación modular: la capa de datos interactúa con la base de datos, la capa de lógica de negocio maneja reglas específicas, y la capa de controladores gestiona las solicitudes y respuestas del usuario.
- Desarrollo de métodos en los controladores para manejar diferentes tipos de solicitudes HTTP (GET, POST, PUT, DELETE). En función de lo que se necesite.
- Despliegue de la API en un servidor local para pruebas.

Resultados HU-05

Figura 38 Arquitectura modular - Backend

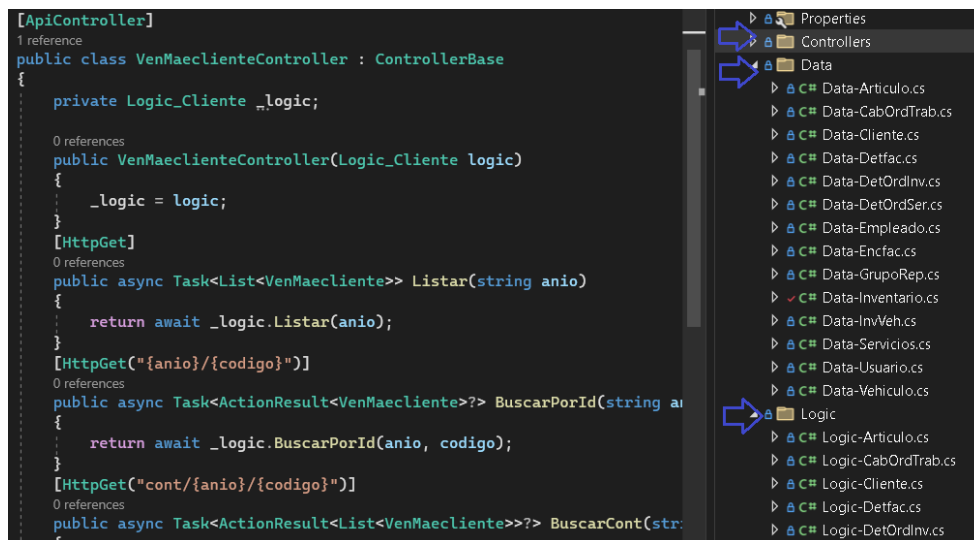
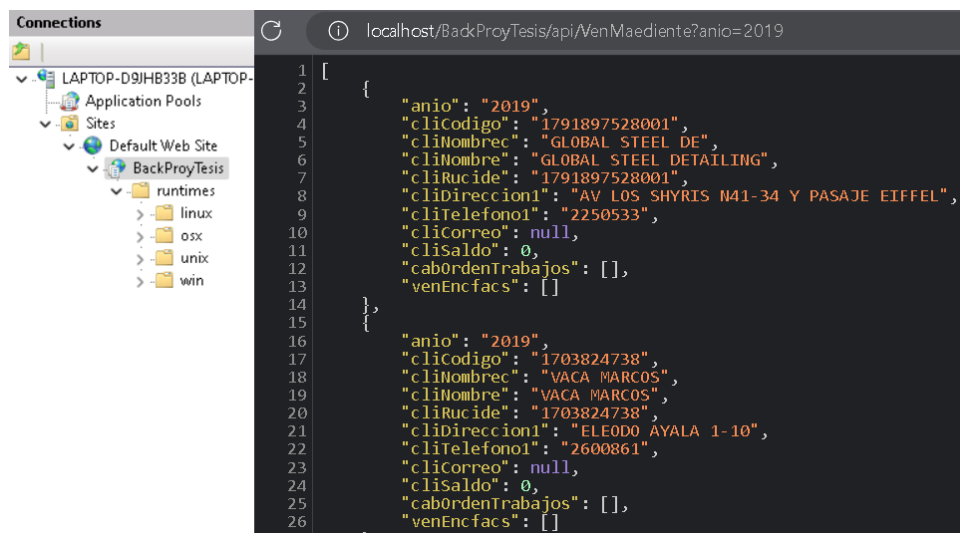


Figura 39 Levantamiento del servicio REST en IIS



4.1.3 Pruebas unitarias Sprint 1

Se realizarán pruebas de funcionalidad a lo largo del desarrollo del aplicativo para detectar errores de manera temprana y facilitar su solución, asegurando así que todas las partes de la aplicación se integren de manera gradual.

Tabla 9 Pruebas unitarias – Sprint 1

Caso de prueba	Resultados esperados	Resultados reales	Estado
Verificar la implementación del modelo de base de datos en la herramienta gráfica Toad.	La estructura de la base de datos en Toad coincide con el modelo de base de datos, e incluye todas las tablas, índices y relaciones.	En TOAD, se logró visualizar todas las tablas, relaciones e índices definidos en el modelo de base de datos.	Exitoso
Verificar que los procesos ETL aseguren la carga precisa y coherente de los datos en un entorno de prueba.	La ejecución de los procedimientos resulta en la carga completa de los datos, respetando las restricciones definidas.	La base de datos de prueba cargó toda la información tal como se planteó en los procedimientos almacenados.	Exitoso
Validar la carga de datos en la base de datos propuesta mediante la ejecución de consultas complejas.	Las consultas deben ejecutarse sin errores y devolver resultados coherentes.	Posterior a la carga de información en la base de datos, se realizaron consultas de datos específicas que validaron que la información se encuentre completa y concisa.	Exitoso
Validar que los modelos de clase generados en el entorno .NET cumplan con la estructura de la base de datos propuesta.	Las propiedades de las clases coinciden con las columnas de las tablas, incluyendo tipos de datos y restricciones.	.NET Core generó modelos de clases acorde a las propiedades de las tablas de la base de datos.	Exitoso
Evaluar la funcionalidad de los métodos (GET, POST, PUT, DELETE) implementados en el API RESTful.	Los métodos (GET, POST, PUT, DELETE) de la API RESTful cumplen con sus respectivas operaciones y proporcionan respuestas HTTP adecuadas.	Posterior a probar los métodos e implementar pequeñas correcciones, el API RESTful cumple efectivamente con todas sus operaciones.	Exitoso

4.1.4 Sprint Review 1

Tabla 10 Reunión sprint 1

Reunión del sprint 1	
Propósito	Validar el trabajo realizado en el primer sprint con un usuario experto en el manejo del sistema actual del taller mecánico. La elección del stakeholder se debe a su experiencia en informática, necesaria para la comprensión del sprint.
Asistentes	Stakeholder: Azucena Zambrano Desarrollador: Noelia Quishpe
Duración	1 hora
Entradas	Servicio RESTful
Salidas	Retroalimentación
Agenda	<ol style="list-style-type: none">1. Presentación del avance de desarrollo Sprint 1.2. Evaluación del Product Backlog General y Sprint Backlog 1:<ul style="list-style-type: none">- Validación de lo cumplido según lo detallado en las Historias de Usuario.3. Retroalimentación de la parte interesada:<ul style="list-style-type: none">- Se especificó y aclaró de mejor manera qué información debe presentarse en las consultas relacionadas con las entidades de artículos y facturas de venta.

4.1.5 Sprint Retrospective 1

¿Qué salió bien?

- Las pruebas unitarias del primer sprint resultaron exitosas, lo que permite seguir adelante con el proyecto según lo planificado.
- El desarrollo del Backend del aplicativo transcurrió sin mayores inconvenientes, gracias a que la fase de diseño definió claramente la idea de lo que se quería construir.

¿Qué salió mal?

- La falta de conocimiento sobre cómo especificar el acceso a esquemas específicos de la base de datos con el comando de scaffolding llevó una extensa investigación y pruebas de acierto y error para generar la conexión a la base de datos.
- La generación automática de modelos de clase no estableció correctamente las relaciones entre las clases de la base de datos. Lo que implicó un análisis adicional para definir de manera correcta las mismas, y evitar errores en el desarrollo del API.
- Swagger, la herramienta predeterminada para administrar APIs en .NET Core, tuvo problemas para mostrar la información. Para solucionarlo temporalmente, se utilizó IIS Express y luego se decidió publicar el servicio local en IIS.

Propósito de mejora

- Modificar los métodos de consulta para que se devuelva únicamente la información especificada por la parte interesada, limitando a lo estrictamente necesario.
- Mejorar la eficiencia de las consultas, optimizando el uso de memoria, CPU y otros recursos del sistema para garantizar un rendimiento óptimo.

4.2 Sprint 2

4.2.1 Sprint Backlog 2

En la Tabla 10 se detallan las tareas a cumplir en el segundo sprint del desarrollo del aplicativo. Estos elementos han sido seleccionados del Product Backlog general y enfocados de manera más efectiva gracias a las lecciones aprendidas del anterior sprint.

Tabla 11 *Sprint Backlog 2*

SPRINT BACKLOG 2				
ID Historia	Historia	Estado	Prioridad	Esfuerzo
HU-06	Diseño de la interfaz de inicio y cierre de sesión	En Desarrollo	Alta	Medio

HU-07	Funcionalidad de inicio y cierre de sesión	En Desarrollo	Alta	Alto
HU-08	Diseño de la interfaz Cliente	En Desarrollo	Alta	Medio
HU-09	Consulta de Clientes con filtros personalizados	En Desarrollo	Alta	Alto
HU-10	Registro de Clientes	En Desarrollo	Media	Alto
HU-11	Modificación de Clientes	En Desarrollo	Media	Alto
HU-12	Eliminación de Clientes	En Desarrollo	Media	Alto
HU-13	Diseño de la interfaz de usuario Ventas	En Desarrollo	Alta	Alto
HU-14	Consulta de ventas con filtros personalizados	En Desarrollo	Alta	Alto
HU-15	Modificación de Ventas para manejo de garantía	En Desarrollo	Alta	Medio
HU-16	Generación de reportes de Venta	En Desarrollo	Alta	Alto

4.2.2 Historias de usuario

Tabla 12 Historia de usuario HU-6

Historia de Usuario			
Código:	HU-06	Usuario: Administrador	
Nombre de Historia:	Diseño de la interfaz de inicio y cierre de sesión		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	1 día	Sprint asignado:	2
Descripción:	<p>El diseño de la interfaz de inicio y cierre de sesión debe tomar en cuenta los siguientes aspectos:</p> <ul style="list-style-type: none"> - Proporcionar un campo claro y etiquetado para que el administrador ingrese su nombre de usuario. 		

- Ofrecer un campo de contraseña seguro que permita ocultar los caracteres para proteger la información confidencial del administrador.
- Agregar un botón agradable al usuario para iniciar sesión, que contenga un texto como "Iniciar Sesión" o "Ingresar".
- Incluir un botón que represente la acción "Cerrar Sesión" dentro de la sesión para que los usuarios puedan salir del sistema de manera sencilla.

Observaciones:

Diseñar mensajes de error claros y comprensibles si se ingresan credenciales incorrectas.

Resultados HU-06

Figura 40 Acceso al sistema



Figura 41 Cerrar sesión



Tabla 13 Historia de usuario HU-7

Historia de Usuario			
Código:	HU-07	Usuario: Administrador	
Nombre de Historia:	Funcionalidad de inicio y cierre de sesión		
Prioridad:	Alta	Esfuerzo en desarrollo:	Alto

Esfuerzo estimado:	2 días	Sprint asignado:	2
Descripción:			
El desarrollo de la funcionalidad de inicio y cierre de sesión debe tomar en cuenta los siguientes aspectos:			
<ul style="list-style-type: none"> - Crear una clase de servicio en el Frontend para consumir el API de usuarios. - Emplear middleware en Vue Router como práctica de seguridad para supervisar la navegación autorizada de los usuarios por el aplicativo: <ul style="list-style-type: none"> a) Controlar la autenticación del usuario antes de permitir el acceso a las rutas. b) Si un usuario no autenticado intenta acceder a cualquier ruta que no sea la página de inicio de sesión, se le redirige automáticamente a la página de inicio de sesión para iniciar sesión antes de acceder a otras secciones de la aplicación. 			
Observaciones:			
Acceso exclusivo al sistema para usuarios con privilegios de administrador.			

Resultados HU-07

Figura 42 *Api Usuarios*

```

Login.vue JS UsuarioService.js U
src > service > JS UsuarioService.js > UsuarioService
1 import axios from 'axios';
2 export default class UsuarioService {
3
4     baseUrl = 'http://localhost/BackProyTesis/api/';
5
6
7     async getUsuAdministrador(usu, token) {
8         try {
9             // Realiza la llamada a la API
10            const response = await axios.get(`${this.baseUrl}SegMaeeusuario`, {
11                params: { usu: usu, token: token }
12            });
13
14            // Devuelve los datos de administrador obtenidos de la API
15            return response.data;
16        } catch (error) {
17            console.error('Error al obtener usuario', error);
18            return []; // Devuelve un array vacío en caso de error
19        }
20    }
21

```

Figura 43 *Middleware Vue Router*

```

197     path: '/auth/access',
198     name: 'accessDenied',
199     component: () => import('@/views/pages/auth/Access.vue')
200   },
201   {
202     path: '/auth/error',
203     name: 'error',
204     component: () => import('@/views/pages/auth/Error.vue')
205   }
206 ]
207 });
208
209 // Guardia de ruta global: se ejecuta antes de cada cambio de ruta en la aplicación.
210 router.beforeEach((to, from, next) => {
211   // Redirige a la página de inicio de sesión si el usuario no está autenticado.
212   if (to.path !== '/auth/login' && !isAuthenticated()) {
213     next('/auth/login');
214   } else {
215     // Permite el acceso a la ruta solicitada si el usuario está autenticado o intenta acceder a la página de inicio de sesión.
216     next();
217   }
218 });

```

Figura 44 *Acceso seguro al sistema – Control de rutas*

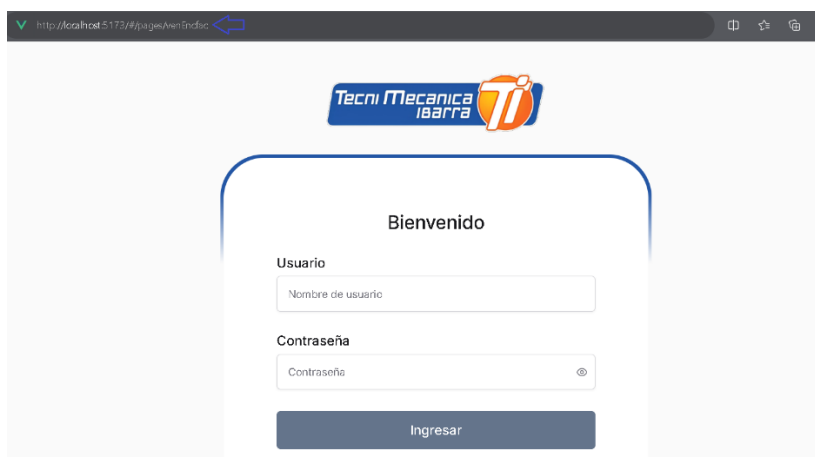


Tabla 14 *Historia de usuario HU-8*

Historia de Usuario			
Código:	HU-08	Usuario: Administrador	
Nombre de Historia:	Diseño de la interfaz Cliente		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	2
Descripción:			
El diseño de la interfaz del módulo de clientes debe tomar en cuenta los siguientes aspectos:			
<ul style="list-style-type: none"> - Página detallada que muestre la información clave de los clientes. - Herramientas de búsqueda y filtros para encontrar fácilmente información de los clientes de todos los periodos. - Formulario de registro para nuevos clientes. 			

- Funcionalidades para actualizar la información del cliente.
- Al hacer clic en el botón eliminar cliente, la interfaz debe mostrar una ventana emergente de confirmación para asegurarse de que el administrador realmente desea proceder con la eliminación.

Observaciones:

Diseñar mensajes de error claros y comprensibles para el administrador si no se puede eliminar un cliente debido a que está relacionado con otra tabla en la base de datos.

Resultados HU-08

Figura 45 Pantalla principal - Clientes

The screenshot shows a web application interface for managing clients. At the top left is the logo for 'Tecní Mecánica Ibarra'. The main header area contains a green '+ Nuevo' button, a 'PERIODO:' dropdown menu currently set to '2023', and a grey 'Exportar' button. Below this is a section titled 'Administrar Clientes' with a search bar labeled 'Buscar...'. The main content is a table with the following data:

CÓDIGO	NOMBRE ↑↓	TELÉFONO	CORREO	DIRECCIÓN	
1717903999001	ARMIJO CARLOS	0967493434	diego.quishpe.mecanica.automotriz@gmail.com	SAN LORENZO	
0400561650001	CADENA HERNAN	0987593345	hcadena60@yahoo.com	JAUN DE DIOS NAVAS 1-51	
1001468774	CALDERON MARCELO	0983930489	edgarcalderson365@gmail.com	ALPACHACA	
1060020370001	GAD Parroquial rural de Imantag	2570130	diego.quishpe.mecanica.automotriz@gmail.com	Eloy alfarero y Amazonas	

Figura 46 Formulario de ingreso Cliente – Validación de campos

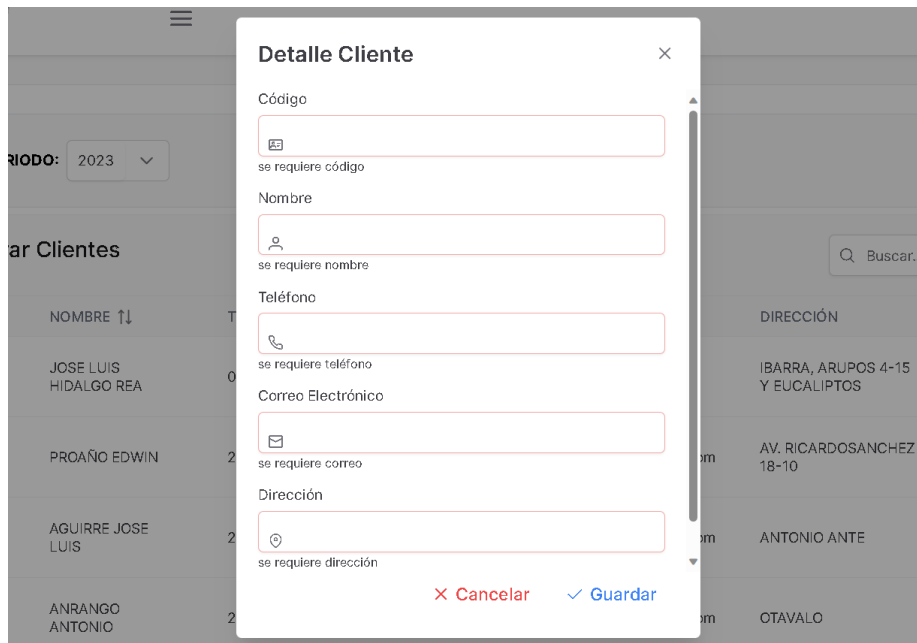


Figura 47 Formulario de ingreso Cliente

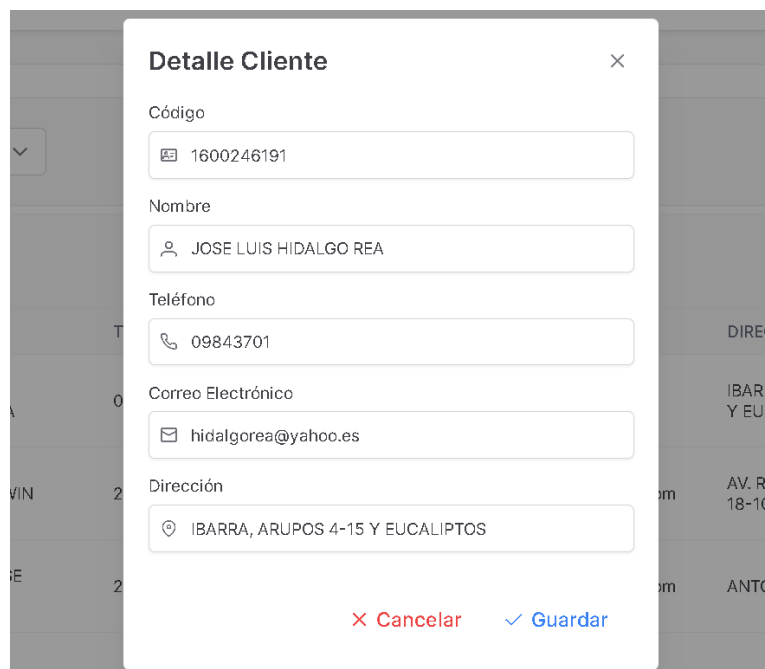


Tabla 15 Historia de usuario HU-9

Historia de Usuario			
Código:	HU-09	Usuario: Administrador	
Nombre de Historia:	Consulta de Clientes con filtros personalizados		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	2

Descripción:

El desarrollo de consultas de Clientes con filtros personalizados debe tomar en cuenta los siguientes aspectos:

- Crear una clase de servicio en el Frontend para consumir la API de clientes.
- Mostrar en la pantalla principal del módulo clientes una consulta general de clientes del período predeterminado 2023.
- Permitir búsqueda flexible de clientes mediante un menú desplegable para periodos y caja de texto global que permita filtrar cualquier propiedad. Mostrar clientes que cumplan con estos criterios.
- Presentar información relevante y esencial para el administrador, evitando datos innecesarios y garantizando una interfaz limpia.

Resultados HU-09

Figura 48 Api Clientes

```
JS ClienteService.js X
src > service > JS ClienteService.js > ClienteService > updateCliente
1 import axios from 'axios';
2 export default class ClienteService {
3
4   baseUrl = 'http://localhost/BackProyTesis/api/';
5
6
7   async getClientes(year) {
8     try {
9       // Realiza la llamada a la API para obtener los clientes del año específico
10      const response = await axios.get(`${this.baseUrl}ven%aecliente`, {
11        params: { año: year }
12      });
13
14      // Devuelve los datos de clientes obtenidos de la API
15      return response.data;
16    } catch (error) {
17      console.error('Error al obtener los clientes:', error);
18      return []; // Devuelve un array vacío en caso de error
19    }
20  }
21
22   async createCliente(cliente) {
23     try {
24       // Realiza la llamada a la API para crear un nuevo cliente
25       const response = await axios.post(`${this.baseUrl}ven%aecliente`, cliente);
26
27       // Devuelve la respuesta de la API en caso de éxito

```

Figura 49 Filtros para búsqueda de información global - Clientes

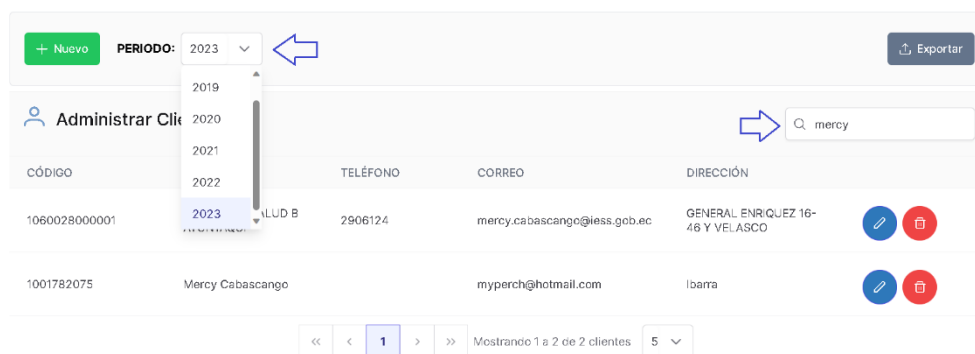


Figura 50 *Búsqueda flexible - Clientes*



Tabla 16 *Historia de usuario HU-10*

Historia de Usuario			
Código:	HU-10	Usuario: Administrador	
Nombre de Historia:	Registro de Clientes		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	2
Descripción:	<p>El aplicativo permitirá el registro de clientes con la siguiente información: nombre, cédula de identidad, dirección domiciliaria y número de celular.</p> <p>El registro de clientes debe tener las siguientes características:</p> <ul style="list-style-type: none"> - El campo de cédula debe permitir únicamente la entrada de caracteres numéricos. - En el campo del número de celular, se permiten solo valores numéricos y se restringe a un máximo de 10 dígitos para asegurar un formato correcto. 		
Observaciones:	<p>Cumplir con las siguientes restricciones para asegurar la integridad en la base de datos:</p> <ul style="list-style-type: none"> - No permitir el registro de usuarios que ya están registrados en la base de datos de clientes. Para ello, se implementará un mecanismo de verificación de registro que impida que un usuario se registre con una cédula que ya este registrada. - Notificar al usuario si existen este tipo de problemas. 		

Resultados HU-10

Figura 51 Registro de Cliente

Detalle Cliente

Código: 1725807597

Nombre: Jessica Andrade

Teléfono: 099675943229

Correo Electrónico: danyp09@hotmail.com

Dirección: Ibarra

Cancelar Guardar

Figura 52 Consulta - Cliente registrado

Administrar Clientes

PERIODO: 2023

Exportar

Q Jessica A

CÓDIGO	NOMBRE	TÉLFONO	CORREO	DIRECCIÓN
1725807597	Jessica Andrade	099675943229	danyp09@hotmail.com	Ibarra

Mostrando 1 a 1 de 1 clientes

Figura 53 Validación de ingreso – Cliente existente

Detalle Cliente

Código: 1002140646

Nombre: Noelia Quishpe

Teléfono: 099675943229

Correo Electrónico: nsquishpe@puce.edu.ec

Dirección: Ibarra

Error
El cliente ya está registrado

Exportar

Q noeli

DIRECCIÓN

Ibarra

Tabla 17 Historia de usuario HU-11

Historia de Usuario			
Código:	HU-11	Usuario: Administrador	
Nombre de Historia:	Modificación de Clientes		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio

Esfuerzo estimado:	2 días	Sprint asignado:	2
Descripción:			
El aplicativo permitirá actualizar la información de clientes, incluyendo su nombre, dirección domiciliaria y número de celular			
La actualización de un registro de cliente debe permitir:			
<ul style="list-style-type: none"> - Acceder al registro y editar la información existente, como nombre, dirección domiciliaria y número de celular, a través del icono de lápiz asociado a cada registro específico. - Guardar cambios únicamente si estos cumplen con el formato establecido. 			
Observaciones:			
Notificar al usuario de manera clara si los datos ingresados no corresponden al formato que se permite.			

Resultados HU-11

Figura 54 *Edición de Cliente*

The screenshot shows a mobile application interface for editing a client's details. The form is titled "Detalle Cliente" and includes the following fields:

- Código:** 1725807597
- Nombre:** Jessica Andrade
- Teléfono:** 099675943229
- Correo Electrónico:** jsandrade@hotmail.com (with a blue arrow icon pointing left)
- Dirección:** Ibarra

At the bottom of the form, there are two buttons: "Cancelar" (with a red 'x' icon) and "Guardar" (with a blue checkmark icon).

Figura 55 Consulta - Cliente actualizado

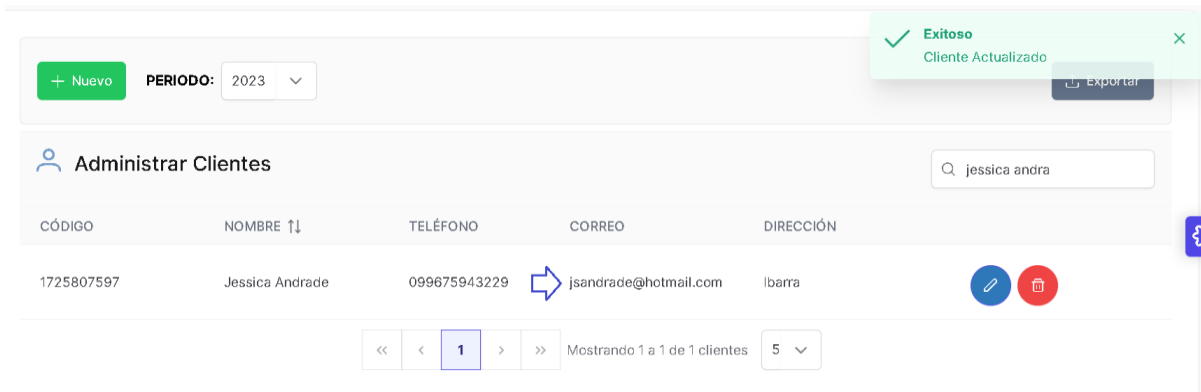


Tabla 18 Historia de usuario HU-12

Historia de Usuario			
Código:	HU-12	Usuario: Administrador	
Nombre de Historia:	Eliminación de Clientes		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	2
Descripción:			
La aplicación permitirá eliminar registros de clientes.			
Previo a la eliminación de un registro de cliente se deberá verificar los siguientes aspectos :			
<ul style="list-style-type: none"> - Validar si el registro está asociado con otras entidades. - Si el registro está relacionado con otras entidades, no podrá ser eliminado. Aparecerá un mensaje informando que el cliente no se puede eliminar. 			
Observaciones:			
Implementar un proceso de confirmación para evitar eliminaciones accidentales por parte de los usuarios.			

Resultados HU-12

Figura 56 Eliminación de Cliente



Figura 57 Notificación – Cliente eliminado

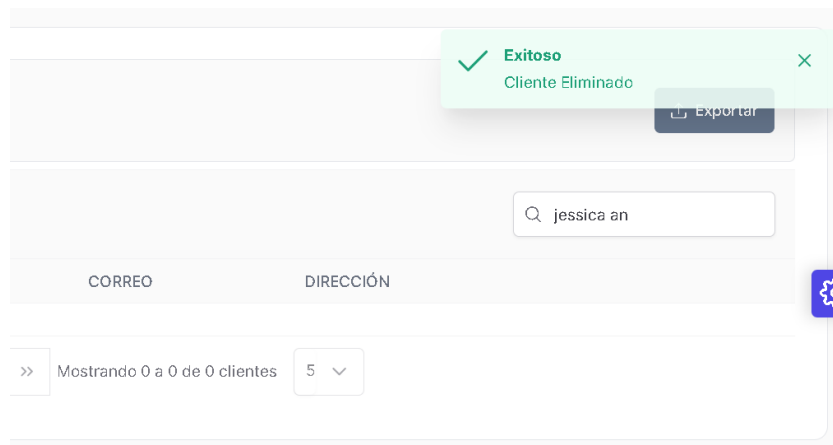


Figura 58 Eliminación de cliente con registros en otras tablas

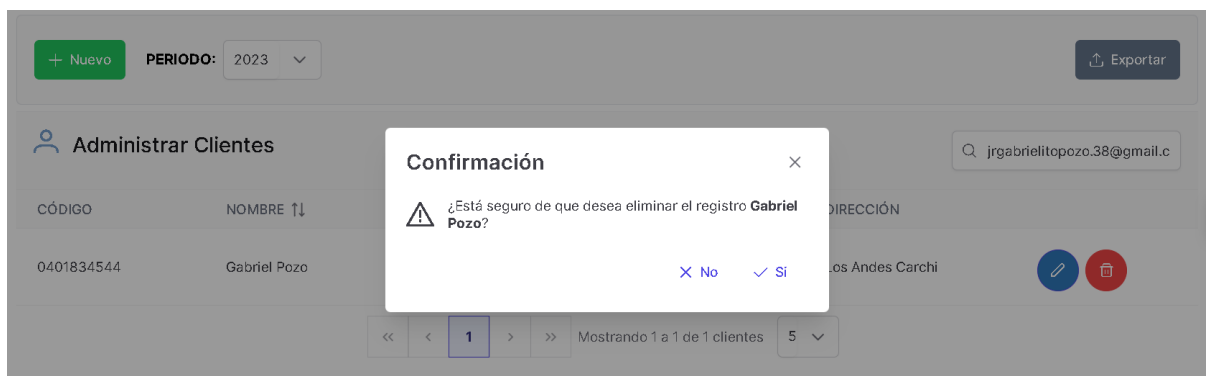


Figura 59 Notificación error de eliminación – Cliente

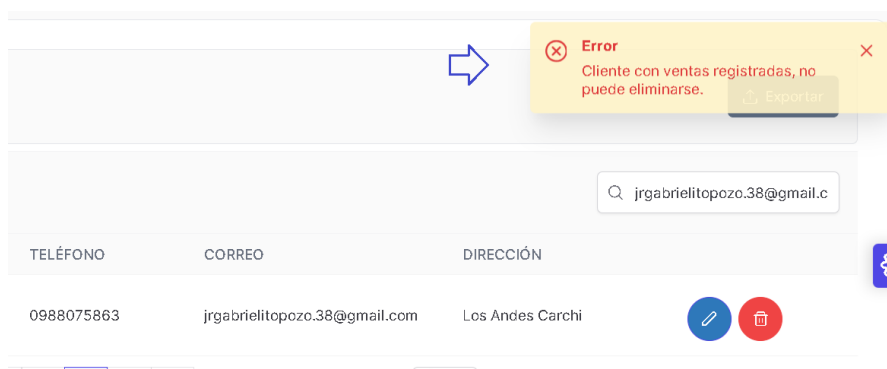


Tabla 19 *Historia de usuario HU-13*

Historia de Usuario			
Código:	HU-13	Usuario: Administrador	
Nombre de Historia:	Diseño de la interfaz de usuario Ventas		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	2
Descripción:			
El diseño de la interfaz del módulo de ventas debe tomar en cuenta los siguientes aspectos:			
<ul style="list-style-type: none"> - Página detallada que muestre la información clave y general de las ventas. - Herramientas de búsqueda y filtros para encontrar fácilmente información general de las ventas de todos los periodos. - Sección para el manejo de reportes. 			
Para visualizar un mantenimiento específico, detallado en la factura de venta, se requiere del diseño de una interfaz que cumpla con las siguientes características:			
<ul style="list-style-type: none"> - Agregar un ícono de lupa en la pantalla general de ventas que permita dirigirse a la pantalla correspondiente del registro de venta deseado. - Presentar información detallada del vehículo , y datos del cliente en secciones separadas. - Mostrar información sobre los servicios realizados y los repuestos utilizados en dicho mantenimiento en una tabla organizada. - Habilitar una caja de texto que permita filtrar cualquier campo del detalle, ya sea servicio o repuesto, para encontrar la información específica de forma rápida. 			

Resultados HU-13

Figura 60 Pantalla principal - Ventas

NÚMERO	NOMBRE	CÓDIGO	PLACA	FECHA	TOTAL	VISUALIZAR
P00000224	ALEXIS OSPINA	0801910514	ICE-0369	2/28/2023	\$160.00	[Icono]
P00000225	ZACARIAS LUIS	1001396215	HBB-2521	2/28/2023	\$59.00	[Icono]
P00000226	Andzia Mora	1726970799	ICR-0084	2/28/2023	\$33.50	[Icono]
P00000227	NAVARRETE ANDRES	1003207949	PDF-7563	3/14/2023	\$38.00	[Icono]

Figura 61 Detalle de Venta – Servicios y Repuestos

TIPO	CÓDIGO	DESCRIPCIÓN	CANTIDAD	PRECIO	%DESCUENTO	TOTAL	TEC
A	219	WIX WL10466A PH6607 Filtro de Aceite	1	\$3.15	9.84	\$2.70	
A	BT-1271	Origs para pasadores de mordazas (10x12 mm)	8	\$0.47	10.64	\$3.28	
A	BT-1272	Fritec Juego de pastillas delanteras	1	\$50.97	9.75	\$46.00	
S	G01	Cambio aceite y filtro de motor.	1	\$19.94	9.73	\$18.00	JU/
NA		Desmontaje-monatje portamordazas y pasadores, reemplazo de	1	\$0.00	0	\$0.00	
NA		origns para solucionar ruido en ruedas delanteras.	1	\$0.00	0	\$0.00	
NA		Cambio pastillas de freno delanteras	1	\$0.00	0	\$0.00	

Total Neto: 69.98
Valor IVA: 6.24
TOTAL: 76.22

Tabla 20 Historia de usuario HU-14

Historia de Usuario			
Código:	HU-14	Usuario: Administrador	
Nombre de Historia:	Consulta de ventas con filtros personalizados		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	2
Descripción:			

El desarrollo de consultas de Ventas con filtros personalizados debe tomar en cuenta los siguientes aspectos:

- Crear una clase de servicio en el Frontend para consumir la API de ventas.

Para consulta general de ventas.

- Mostrar en la pantalla principal del módulo ventas una consulta general de ventas en el período predeterminado 2023.
- Permitir la búsqueda flexible de ventas mediante un menú desplegable para la selección de periodos o caja de texto que permita filtrar cualquier propiedad. Mostrar las ventas que cumplan con estos criterios.
- Presentar información relevante de ventas para el administrador, evitando datos innecesarios y garantizando una interfaz limpia.

Para consulta específica de venta/mantenimiento.

- Mostrar en la pantalla de detalle de venta, información simplificada relacionada con el vehículo, cliente, servicios y mantenimientos.
- Permitir la búsqueda flexible de servicios o repuestos del mantenimiento mediante un filtro global que acepte cualquier valor de los campos del detalle de la venta.

Resultados HU-14

Figura 62 *Api Ventas*

```
src > service > JS VenEncfacService.js > ClienteService
1 import axios from 'axios';
2 export default class ClienteService {
3
4   baseUrl = 'http://localhost/BackProyTesis/api/';
5
6   async modeloAuto(year) {
7     try {
8       // Realiza la llamada a la API para obtener los datos de VenEncfac
9       const venEncfacResponse = await axios.get(`${this.baseUrl}venEncfac`, {
10         params: { anio: year }
11       });
12
13       const venEncfacData = venEncfacResponse.data;
14
15       if (venEncfacData.length === 0) {
16         return [];
17       }
18
19       // Crea un array de promesas para las llamadas a venEncfac en paralelo
20       const promises = venEncfacData.map((venEncfac) =>
21         axios.get(`${this.baseUrl}venEncfac/fac/${year}/${venEncfac.encfacNumero}`)
22           .then((response) => {
23             const venEncfacData = response.data;
24             venEncfac.vhcspcPlaca = venEncfacData.vhcspcPlaca;
25             return venEncfac;
26           })
27       );

```

Figura 63 Filtros para búsqueda de información global - Ventas

PERIODO: 2023

Adi /ventas

Q Mercy Ca

NÚMERO	ABRE	CÓDIGO	PLACA	FECHA	TOTAL	VISUALIZAR
P00000742	Cabascango	1001782075	IBE-4042	7/1/2023	\$76.22	🔍
P00000321	Mercy Cabascango	1001782075	IBE-4042	3/24/2023	\$26.00	🔍
P00000080	Mercy Cabascango	1001782075	IBE-4042	1/21/2023	\$109.00	🔍

Mostrando 1 a 3 de 3 Ventas 5

Figura 64 Búsqueda flexible – Detalle Ventas

← Garantía

Factura: P00000742
Fecha: 7/1/2023
Usuario: nsquishpe

PROPIETARIO
Mercy Cabascango - 1001782075

VEHÍCULO
MAZDA IBE-4042 VINO - 76058 km

Detalle

Q Camb

TIPO	CÓDIGO	DESCRIPCIÓN	CANTIDAD	PRECIO	%DESCUENTO	TOTAL	TEC
S	G01	Cambio aceite y filtro de motor.	1	\$19.94	9.73	\$18.00	JU/
NA		Cambio pastillas de freno delanteras	1	\$0.00	0	\$0.00	

Figura 65 Búsqueda flexible – Detalle Ventas

← Garantía

Factura: P00000742
Fecha: 7/1/2023
Usuario: nsquishpe

PROPIETARIO
Mercy Cabascango - 1001782075

VEHÍCULO
MAZDA IBE-4042 VINO - 76058 km

Detalle

Q BT-1272

TIPO	CÓDIGO	DESCRIPCIÓN	CANTIDAD	PRECIO	%DESCUENTO	TOTAL	TEC
A	BT-1272	Fritec Juego de pastillas delanteras	1	\$50.97	9.75	\$46.00	

Mostrando 1 a 1 de 1 líneas 8

Tabla 21 Historia de usuario HU-15

Historia de Usuario			
Código:	HU-15	Usuario: Administrador	
Nombre de Historia:	Modificación de Ventas para manejo de garantía		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	1 días	Sprint asignado:	2
Descripción:			
El aplicativo permitirá la actualización de la información de las ventas para registrar si un vehículo ha regresado o no al taller mecánico por garantía de servicios.			
La actualización de un registro de venta debe permitir:			
<ul style="list-style-type: none"> - Acceder a la ventana de garantía, especificar observaciones de persistencia de fallos y marcar 'Sí'. - Guardar cambios en la base de datos. 			
Además, incluir en la ventana principal del aplicativo un espacio para la lista de vehículos que han solicitado garantía, facilitando así un acceso rápido.			

Resultados HU-14

Figura 66 Registro de garantía - Ventas

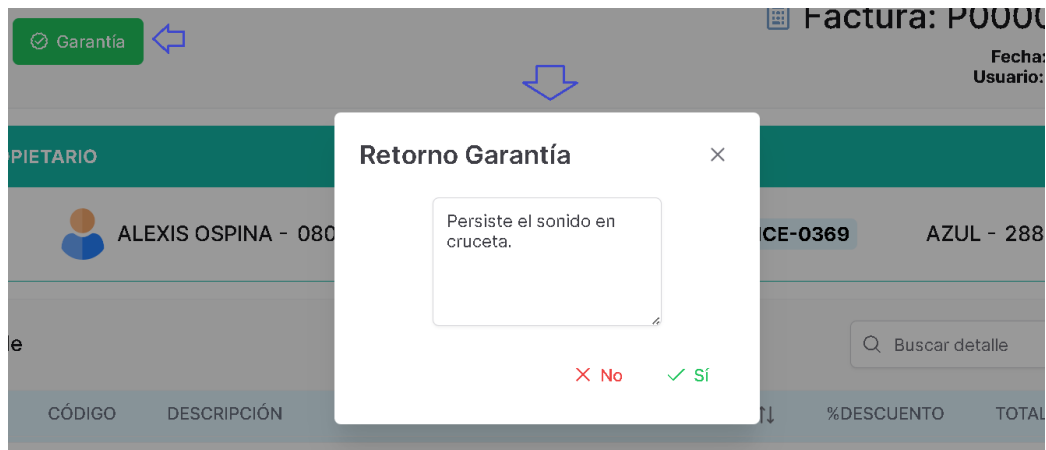


Figura 67 Lista retorno de vehículos por garantía

Número ↑↓	Placa	Observación	Ver
P00000224	ICE-0369	Persiste el sonido en cruceta.	Q
P00000225	HBB-2521	Ruido en cilindro de frenos.	Q
P00000742	IBE-4042	Continúa sonido ruedas delanteras.	Q
P00000354	ICR-0084	Ruido en la dirección.	Q

Tabla 22 Historia de usuario HU-16

Historia de Usuario			
Código:	HU-16	Usuario:	Administrador
Nombre de Historia:	Generación de reportes de Venta		
Prioridad:	Alta	Esfuerzo en desarrollo:	Alto
Esfuerzo estimado:	2 días	Sprint asignado:	1
Descripción:	<p>El aplicativo permitirá la generación de dos tipos de reportes de ventas: uno específico por cliente y otro general, ambos basados en un rango de fechas especificado. El desarrollo del apartado de Reportes comprende las siguiente subtareas:</p> <ul style="list-style-type: none"> - Codificación de la estructura del reporte. - Según el tipo de informe, capturar los parámetros indicados y mediante la librería jsPDF, crear y descargar el informe en formato PDF para su visualización. 		
Observaciones:	<p>Podría darse el escenario en el cual, al intentar generar un reporte de ventas, no se encuentren facturas registradas en las fechas seleccionadas. En tal situación, se notificará al usuario mediante una alerta sobre la imposibilidad de generar el reporte.</p>		

Resultados HU-14

Figura 68 Ventana emergente: Parámetros para el Reporte de Ventas por Cliente

REPORTE FACTURAS EMITIDAS ×

Fecha Inicio:

Fecha Fin:

Cliente:

Mercy Cabascango - 1001782075

Generar Reporte

Figura 69 Reporte de Ventas por Cliente

11/12/2023, 12:02:07 AM

FACTURAS EMITIDAS
Desde: 01/01/2023 Hasta: 12/08/2023

Factura Nro.	Cliente	Código	Fecha	Estado	Subtotal	Iva	Total
P00000742	Mercy Cabascango	1001782075	7/1/2023	PAGADO	69.98	6.24	76.22
P00000321	Mercy Cabascango	1001782075	3/24/2023	PAGADO	25.89	0.11	26
P00000080	Mercy Cabascango	1001782075	1/21/2023	PAGADO	100.69	8.31	109
Total:							211.22

Figura 70 Ventana emergente: Parámetros para el Reporte de General de Ventas

REPORTE FACTURAS EMITIDAS ×

Fecha Inicio:

Fecha Fin:

Cliente:

Generar Reporte

Figura 71 Reporte General de Ventas

Factura Nro.	Cliente	Código	Fecha	Estado	Subtotal	Iva	Total
P00000224	ALEXIS OSPINA	0801910514	2/28/2023	PAGADO	148.6	11.4	160
P00000225	ZACARIAS LUIS	1001396215	2/28/2023	PAGADO	55.4	3.6	59
002002000001393	Andzia Mora	1726970799	2/28/2023	PAGADO	30.23	3.27	33.5
002002000001408	NAVARRETE ANDRES	1003207949	3/14/2023	PAGADO	34.22	3.78	38
P00000228	CONSUMIDOR FINAL	999999999999	2/28/2023	PAGADO	138.26	6.74	145
P00000229	ZACARIAS LUIS	1001396215	3/1/2023	PAGADO	3.28	0.21	3.49
P00000230	RUBIO GONZALO	1001933462	3/1/2023	PAGADO	23	0	23
P00000231	EDISON BELTRAN	1001225034	3/2/2023	PAGADO	374.47	25.53	400
P00000232	ANDRES ZABALA	1003437330	3/2/2023	PAGADO	52.03	5.97	58
P00000234	Edum Vaca Cevallos	1001630225	3/3/2023	PAGADO	7	0	7
P00000235	Cangas Ortiz Aura Elisa	0401464151001	4/25/2023	PAGADO	249.83	28.78	278.61
P00000236	JOSE TAMAYO	1757089238	3/3/2023	PAGADO	19.47	0.53	20

Figura 72 Cliente sin registros de ventas en las fechas especificadas – Reporte Ventas por Cliente.

4.2.3 Pruebas unitarias Sprint 2

Se realizarán pruebas de funcionalidad a lo largo del desarrollo del aplicativo para detectar errores de manera temprana y facilitar su solución, asegurando así que todas las partes de la aplicación se integren de manera gradual.

Tabla 23 Pruebas unitarias – Sprint 2

Caso de prueba	Resultados esperados	Resultados reales	Estado
Verificar inicio y cierre de sesión en el aplicativo con un usuario de prueba.	El sistema valida las credenciales y redirige al usuario a la página principal. Tras cerrar sesión, lo redirige a la página de inicio.	Inicio de sesión y cierre de sesión completados con éxito, confirmando la autenticación del usuario y su salida segura del sistema.	Exitoso

Validar consulta de clientes con filtros personalizados (Período 2018 y Número de Cédula 1004695654).	El sistema presenta la información del cliente que coincide con los parámetros especificados	El sistema presentó la información del cliente que se ajusta los criterios definidos	Exitoso
Validar el registro de un cliente de prueba	El sistema registra el cliente en la base de datos.	Registro satisfactorio del cliente de prueba en la base de datos.	Exitoso
Evaluar actualización del cliente de prueba.	El sistema actualiza la información del cliente en la base de datos.	Actualización satisfactoria del cliente de prueba en la base de datos.	Exitoso
Evaluar eliminación del cliente de prueba.	El sistema valida la existencia del registro en otras tablas; al no encontrarlo, lo elimina.	Validación y eliminación satisfactoria del cliente de prueba en la base de datos.	Exitoso
Validar consulta de venta con filtros personalizados (Período 2022 y Número de Factura P00000225).	El sistema presenta la información del mantenimiento que coincide con los parámetros especificados	El sistema presentó la información del mantenimiento, que se ajusta los criterios definidos.	Exitoso
Verificación de la actualización de ventas para el manejo de garantía.	El sistema almacena la información de la garantía y al consultar la venta resalta el estado afirmativo de la garantía junto con la observación en la caja de texto correspondiente.	El sistema registró y presentó la información de la garantía conforme a los datos actualizados en las ventas.	Exitoso
Validar la generación correcta de reportes de ventas, asegurando la coherencia de los datos según los parámetros establecidos.	El sistema genera un reporte consistente acorde a las ventas que cumplen con los parámetros especificados.	Generación satisfactoria del reporte. Lista las ventas íntegras esperadas.	Exitoso

4.2.4 Sprint Review 2

Tabla 24 Reunión sprint 2

Reunión del sprint 2	
Propósito	Validar el desarrollo del segundo sprint con el administrador del taller mecánico, quien es el principal interesado en el aplicativo debido a sus dificultades con el manejo del sistema actual.
	Stakeholder: Diego Quishpe
Asistentes	Desarrollador: Noelia Quishpe
Duración	1 hora
Entradas	Funcionalidad módulo clientes y ventas.
Salidas	Retroalimentación
Agenda	<ol style="list-style-type: none">1. Presentación del avance de desarrollo Sprint 2.2. Evaluación del Product Backlog General y Sprint Backlog 2:<ul style="list-style-type: none">- Validación de lo cumplido según lo detallado en las Historias de Usuario.3. Retroalimentación de la parte interesada:<ul style="list-style-type: none">- Se especificó y aclaró de mejor manera como organizar la información en pantalla y los colores a usar en las interfaces para alinearse a la colorimetría del negocio.

4.2.5 Sprint Retrospective 2

¿Qué salió bien?

- Las pruebas unitarias del segundo sprint resultaron exitosas, lo que permite seguir adelante con el proyecto según lo planificado.
- El módulo de clientes se desarrolló sin mayores complicaciones, proporcionando todas las funcionalidades necesarias detalladas en los requerimientos.
- Las interfaces de acceso al sistema, clientes y ventas fueron diseñadas según las especificaciones del cliente, asegurando cumplimiento de sus requisitos.

¿Qué salió mal?

- La falta de experiencia en la manipulación de componentes Vue HTML retrasó ligeramente el diseño de las interfaces.

Propósito de mejora

- Mejorar las interfaces de usuario de acuerdo con las especificaciones del cliente, las cuales se fundamentan en la distribución de información y la paleta de colores utilizada en el aplicativo.

4.3 Sprint 3

4.3.1 Sprint Backlog 3

En la Tabla 22 se detallan las tareas a cumplir en el último sprint del desarrollo del aplicativo. Estos elementos han sido seleccionados del Product Backlog general y de la retrospectiva de los sprints anteriores. La culminación de este sprint, conforme a lo planificado, da como resultado un prototipo funcional completo del aplicativo.

Tabla 25 *Sprint Backlog 3*

SPRINT BACKLOG 3				
ID Historia	Historia	Estado	Prioridad	Esfuerzo
HU-17	Diseño de la interfaz de usuario Inventario	En Desarrollo	Alta	Alto
HU-18	Consulta de inventario con filtros personalizados	En Desarrollo	Alta	Alto
HU-19	Generación de reportes de Existencias	En Desarrollo	Alta	Alto
HU-20	Diseño de la interfaz de usuario Órdenes de trabajo	En Desarrollo	Media	Alto
HU-21	Consulta de Órdenes de trabajo	En Desarrollo	Media	Alto
HU-22	Registro de Órdenes de trabajo	En Desarrollo	Media	Alto
HU-23	Modificación de Órdenes de trabajo	En Desarrollo	Media	Alto

HU-24	Eliminación de Órdenes de trabajo	En Desarrollo	Media	Alto
--------------	-----------------------------------	---------------	-------	------

4.3.2 Historias de usuario

Tabla 26 *Historia de usuario HU-17*

Historia de Usuario			
Código:	HU-17	Usuario: Administrador	
Nombre de Historia:	Diseño de la interfaz de usuario Inventario		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	3
Descripción:			
El diseño de la interfaz del módulo de inventario debe tomar en cuenta los siguientes aspectos:			
<ul style="list-style-type: none"> - Página detallada que muestre la información clave y general de los artículos. - Herramientas de búsqueda y filtros para encontrar fácilmente información general del inventario de todos los periodos. 			
Para visualizar el kardex de un producto en específico, se requiere del diseño de una interfaz que cumpla con las siguientes características:			
<ul style="list-style-type: none"> - Agregar un ícono de lupa en la pantalla general de inventario que permita dirigirse a la pantalla de kardex del producto seleccionado. - Mostrar información detallada del kardex, incluyendo entradas, salidas, costos de compra y venta, entre otros aspectos relevantes. - Habilitar filtro por entradas y salidas. - Habilitar una caja de texto que permita filtrar cualquier campo del detalle de kardex, ya sea información de entrada o salida, para encontrar la información específica de forma rápida. 			

Resultados HU-17

Figura 73 Pantalla principal – Inventario

CÓDIGO	DESCRIPCIÓN	CATEGORÍA	PRECIO	STOCK	STATUS	VISUALIZAR
300	FANFARO SCT 8702 75W90 Fully Synthetic 1 Lt)	ACEITES	\$12.54	0	AGOTADO	[Icono]
301	FANFARO SCT 8615 ATF Dexron VI Fully Synthetic (1 Lt)	ACEITES	\$10.07	2	LIMITADO	[Icono]
573	TOTAL QUARTZ 7000 10W30 Semi Synthetic (Gal)	ACEITES	\$27.64	10	DISPONIBLE	[Icono]
574	TOTAL QUARTZ 7000 10W30 Semi Synthetic (Qt)	ACEITES	\$7.59	9	DISPONIBLE	[Icono]

Figura 74 Detalle de Inventario – Kardex: Salidas

Stock: 10 unidades

P. Compra: \$ 12.074 **P.V.P:** \$ 27.642

FECHA	CONCEPTO	BODEGA	TIPO DOC	# DOC	EGRESOS	COSTO
6/17/2023	SA	B001	FAC	P00000676	1	\$10.78
6/19/2023	SA	B001	FAC	P00000684	1	\$10.78
6/20/2023	SA	B001	FAC	P00000689	1	\$10.78
6/27/2023	SA	B001	FAC	P00000709	1	\$10.78

Figura 75 Detalle de Inventario – Kardex: Entradas

Stock: 10 unidades

P. Compra: \$ 12.074 **P.V.P:** \$ 27.642

FECHA	CONCEPTO	BODEGA	TIPO DOC	# DOC	INGRESOS	COSTO
6/17/2023	EN	B001	COM	000023339	18	\$194.04

Mostrando 1 a 1 de 1 líneas Kardex

Tabla 27 Historia de usuario HU-18

Historia de Usuario	
Código:	HU-18 Usuario: Administrador

Nombre de Historia:	Consulta de inventario con filtros personalizados		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	3
Descripción:			
El desarrollo de consultas de Inventario con filtros personalizados debe tomar en cuenta los siguientes aspectos:			
<ul style="list-style-type: none"> - Crear una clase de servicio en el Frontend para consumir la API de artículos y kardex. 			
Para consulta general de inventario.			
<ul style="list-style-type: none"> - Mostrar en la pantalla principal del módulo inventario una consulta general de productos en el período predeterminado 2023. - Habilitar un filtro por categoría de productos. - Permitir la búsqueda flexible de ventas mediante un menú desplegable para la selección de periodos o caja de texto que permita filtrar cualquier campo de la información de artículos. Mostrar los artículos que cumplan con estos criterios. - Presentar información relevante de inventario para el administrador, evitando datos innecesarios y garantizando una interfaz limpia. 			
Para consulta específica de kardex de articulo.			
<ul style="list-style-type: none"> - Mostrar en la pantalla información del artículo, como el stock disponible, costos de compra y venta, fecha de ingreso, etc. - Filtrar información del kardex por entradas y salidas, permitiendo una visualización más ordenada de los datos. - Permitir la búsqueda flexible de entradas o salidas mediante un filtro global que acepte cualquier valor de la información del detalle de kardex. 			

Resultados HU-18

Figura 76 Filtros para búsqueda de información global – Artículos

PERIODO: 2023 CATEGORÍAS: ACEITES

Administrar Inventario

Q fanfar

CÓDIGO	DESCRIPCIÓN	CATEGORÍA	PRECIO	STOCK	STATUS	VISUALIZAR
510	FANFARO SCT 10W30 SPX Semi Synthetic (1 Lt)			0	AGOTADO	🔍
511	FANFARO 5W30 Nissan Full Synthetic (4 Lt)	ACEITES	\$47.35	1	LIMITADO	🔍
513	FANFARO 5W40 (4 Lt)	ACEITES	\$46.54	0	AGOTADO	🔍
270	FANFARO SCT 75W80 GL-4 Fully Synthetic (1 Lt)	ACEITES	\$13.60	5	DISPONIBLE	🔍

Figura 77 Búsqueda flexible – Artículos

PERIODO: 2022 CATEGORÍAS: ACTUADORES

Administrar Inventario

Q Lubricante

CÓDIGO	DESCRIPCIÓN	CATEGORÍA	PRECIO	STOCK	STATUS	VISUALIZAR
481	TEK BOND 69957367255 Spray Lubricante Penetrante (300ml)	ACTUADORES	\$2.96	2	LIMITADO	🔍
215	Lubricante para Bandas Vela	ACTUADORES	\$1.24	0	AGOTADO	🔍
178	Penetrante Lubricante WILITA (600ml)	ACTUADORES	\$6.65	3	DISPONIBLE	🔍

Figura 78 Búsqueda flexible – Detalle Kardex

ENTRADAS SALIDAS TOTAL QUARTZ 7000 10W30 Semi Synthetic (Gal) - 573

Stock: 10 unidades

P. Compra: \$ 12.074 P.V.P: \$ 27.642

KARDEX Q 6/29/2023

FECHA	CONCEPTO	BODEGA	TIPO DOC	# DOC	EGRESOS	COSTO
6/29/2023	SA	B001	FAC	P00000730	1	\$10.78
6/29/2023	SA	B001	FAC	P00000698	1	\$10.78

Mostrando 1 a 2 de 2 líneas Kardex 5

Tabla 28 Historia de usuario HU-19

Historia de Usuario	
Código:	HU-19 Usuario: Administrador

Nombre de Historia:	Generación de reportes de Venta		
Prioridad:	Alta	Esfuerzo en desarrollo:	Alto
Esfuerzo estimado:	2 días	Sprint asignado:	3
Descripción:	<p>El aplicativo permitirá la generación de dos tipos de reportes de existencia de artículos: uno específico por categoría y otro general, ambos basados en una fecha de corte especificada. El desarrollo del apartado de Reportes comprende las siguiente subtareas:</p> <ul style="list-style-type: none"> - Codificación de la estructura del reporte. - Según el tipo de informe, capturar los parámetros indicados y mediante la librería jsPDF, crear y descargar el informe en formato PDF para su visualización. 		
Observaciones:	<p>Podría darse el escenario en el cual, al intentar generar un reporte de existencias, no se encuentren existencias registradas en la fecha de corte establecida. En tal situación, se notificará al usuario mediante una alerta sobre la imposibilidad de generar el reporte.</p>		

Resultados HU-19

Figura 79 Ventana emergente: *Parámetros para el Reporte de Existencias por Categoría – Artículos*

The image shows a modal window titled "REPORTE EXISTENCIA B001" with a close button (X) in the top right corner. Below the title, there are two main input fields:

- Fecha Corte:** A date input field containing "01/26/2023" and a calendar icon to its right.
- Grupo:** A dropdown menu currently showing "ADITIVOS" with a blue arrow icon to its right. The dropdown list is open, showing the following options:
 - ACEITES
 - ACTUADORES
 - ADITIVOS (highlighted in light blue)
 - AMORTIGUADORES
 - BANDAS DE DISTRIBUCION

Figura 80 Reporte de Existencias por Categoría – Artículos

rs/Noella%20Quishpe/Downloads/Report-ADITIVOS-26_01_2023-8001.pdf

aw | Read aloud | Ask Copilot | 1 of 1

Tecní Mecánica Ibarra

EXISTENCIA - PRECIOS

Fecha de Corte: 26/01/2023

ADITIVOS

Código	Nombre	Existencia	Neto	PVP
66	CYCLO Limpiador de Frenos y Piezas 14Oz	1	4.46	4.9952
67	CYCLO Spray Bornes de Batería (11 Oz)	1	9.38	10.5056
110	LOCTITE 37531 Compuesto Adherente Acero Liquido (28 g)	3.25	8.04	9.0048
116	MANNOL 6105 Cockpit Reiniger (220 ml)	8	4.46	4.9952
285	MANNOL 9091 Aditivo Molibden Additive (350 ml)	1	4.69	5.2528
521	MANNOL 9202 Aditivo Limpiador de Catalizador (450 ml)	6	7.2	8.064
125	MANNOL 9829 Aditivo Ceramo Ester Metálico 300ml	1	21.88	24.5056
126	MANNOL 9873 Limpiador Cuerpo De Aceleración/Mariposa (400ml)	1	6.92	7.7504

Figura 81 Ventana emergente: Parámetros para el Reporte General Existencias – Artículos

REPORTE EXISTENCIA B001 ×

Fecha Corte: 01/26/2023

Grupo: General ▾

Generar Reporte

Figura 82 Reporte General de Existencias– Artículos

EXISTENCIA - PRECIOS
Fecha de Corte: 26/01/2023

ACEITES

Código	Nombre	Existencia	Neto	PVP
305	CHEMPIOIL 5W30 Aceite Full Syntetico (4 Lt)	0.27	30.56	34.2272
293	CHEMPIOIL 9501 10W40 Semy Synthetic (4 Litros)	0.97	24.11	27.0032
511	FANFARO 5W30 Nissan Full Synthetic (4 Lt)	1.62	38.16	42.7392
513	FANFARO 5W40 (4 Lt)	4	32.14	35.9968
510	FANFARO SCT 10W30 SPX Semi Syntetic (1 Lt)	13	6.25	7
292	FANFARO SCT 10W30 SPX Semi Syntetic (4 Litros)	1.14	22.32	24.9984
271	FANFARO SCT 15W40 (10 Lt)	5	56.25	63
294	FANFARO SCT 6506 20W50 Synthetic Blend (4 Litros)	2.7	25	28
270	FANFARO SCT 75W80 GL-4 Fully Syntetic (1 Lt)	16.5	9.82	10.9984
301	FANFARO SCT 8615 ATF Dexron VI Fully Synthetic (1 Lt)	2.25	12.05	13.496

ACTUADORES

Código	Nombre	Existencia	Neto	PVP
308	CYCLO C2100 Tratamiento Humo (355 ml)	1	7.14	7.9968
307	LOCTITE 7647 Limpiador Contactos Electronico (220 ml)	1	3.57	3.9984
178	Penetrante Lubricante WILITA (600ml)	3.4	6.25	7
493	SENFINECO Elevador de Octanaje	3	4.69	5.2528
481	TEK BOND 69957367255 Spray Lubricante Penetrante (300ml)	1.75	3.57	3.9984
509	WILITA W-800 Limpiador Carburador	5.05	4.46	4.9952
306	WUNSCHER Limpiador Carburador e Inyector (650 ml)	2.2	4.46	4.9952

Tabla 29 Historia de usuario HU-20

Historia de Usuario			
Código:	HU-20	Usuario:	Administrador
Nombre de Historia:	Diseño de la interfaz de usuario Órdenes de trabajo		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	3
Descripción:	<p>El diseño de la interfaz del módulo órdenes de trabajo debe tomar en cuenta los siguientes aspectos:</p> <ul style="list-style-type: none"> - Página detallada que muestre la información clave de las órdenes de trabajo. - Herramientas de búsqueda y filtros para encontrar fácilmente información de las órdenes de trabajo del periodo actual. - Formulario de registro para nuevas órdenes de trabajo. - Funcionalidades para actualizar la información de la orden. 		

- Al hacer clic en el botón eliminar orden de trabajo, esta se elimina sin problemas, ya que no está vinculada a registros en otras tablas.

Observaciones:

Hacer uso de componentes de Vue.js para un diseño atractivo y amigable para el usuario.

Resultados HU-20

Figura 83 Pantalla principal – Órdenes de Trabajo

NÚMERO	NOMBRE	CÓDIGO	FECHA	PLACA	MARCA	
1	ZAMBRANO ENRIQUE	2365985298	11/12/2023	AAC-2345	FIAT	
2	Luis Adrián Chiliqueña Jaramillo	1001459138	11/12/2023	AAC-0123	LAND ROVER	
3	DE LA CRUZ FERNANDO	1001620762	11/13/2023	PCU-2343	JEEP	
4	PADILLA JOSE	1001464393	11/13/2023	PDF-2345	VOLVO	
5	PAOLA TERAN	1003429253	11/13/2023	MBC-250	CHEVROLET	

Figura 84 Formulario de ingreso Orden de Trabajo – Parte 1

Nro. Orden de trabajo: 7

Fecha: 11/12/2023

Cliente:

Tanque de Gasolina:

Placa: AAC-0123

Color:

Marca:

Kilometraje:

INVENTARIO VEHÍCULO

INTERIORES	EXTERIORES	ACCESORIOS
<input type="checkbox"/> Alarma <input type="text"/> <input type="checkbox"/> Perillas <input type="text"/> <input type="checkbox"/> Luz Check <input type="text"/> <input type="checkbox"/> Gata-Llave-Palanca <input type="checkbox"/> Llaves Vehículo <input type="checkbox"/> Espejo Int. <input type="checkbox"/> Cenicero <input type="checkbox"/> Llanta Emergencia <input type="checkbox"/> Pito	<input type="checkbox"/> Nro. Placas <input type="text"/> <input type="checkbox"/> Nro. Plumas <input type="text"/> <input type="checkbox"/> Nro. Espejos Exteriores <input type="text"/> <input type="checkbox"/> Nro. Tapacubos <input type="text"/> <input type="checkbox"/> Seguro de Ruedas <input type="checkbox"/> Antena <input type="checkbox"/> Emblemas <input type="checkbox"/> Tapa Combustible <input type="checkbox"/> Tapa Cubre Motor	<input type="checkbox"/> Memory USB

Figura 85 Formulario de ingreso Orden de Trabajo – Parte 2

Tabla 30 Historia de usuario HU-21

Historia de Usuario			
Código:	HU-21	Usuario: Administrador	
Nombre de Historia:	Consulta de Órdenes de trabajo		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	1
Descripción:	<p>El desarrollo de consultas de órdenes de trabajo con filtros personalizados debe tomar en cuenta los siguientes aspectos:</p> <ul style="list-style-type: none"> - Crear una clase de servicio en el Frontend para consumir la API de órdenes de trabajo. - Mostrar en la pantalla principal del módulo órdenes de trabajo una consulta general de clientes del período predeterminado 2023. - Permitir búsqueda flexible de órdenes de trabajo mediante una caja de texto que permita filtrar cualquier campo. Mostrar órdenes de trabajo que cumplan con dichos criterios. - Presentar información relevante y esencial para el administrador, evitando datos innecesarios y garantizando una interfaz limpia. 		

Resultados HU-21

Figura 86 Api Órdenes de Trabajo

```

src > service > JS OrdTrabCabService.js > OrdTrabCabService
1 import axios from 'axios';
2 export default class OrdTrabCabService {
3
4   baseUrl = 'http://localhost/BackProyTesis/api/';
5
6
7   async getOrdenes(year) {
8     try {
9       // Realiza las llamadas a la API para obtener órdenes de trabajo y marcas en paralelo
10      const [ordenesResponse, marcasResponse] = await Promise.all([
11        axios.get(`${this.baseUrl}CabOrdenTrabajo`, { params: { anio: year } }),
12        axios.get(`${this.baseUrl}venVehmarmod`);
13      ]);
14
15      // Obtiene los datos de órdenes de trabajo y marcas
16      const ordenes = ordenesResponse.data;
17      const marcas = marcasResponse.data;
18
19      // Mapea los códigos de marca a los nombres de las marcas en los objetos de órdenes de trabajo
20      const ordenesConNombresDeMarcas = ordenes.map(orden => {
21        const marca = marcas.find(marca => marca.vehmarmodCodigo === orden.vehmarmodCodigo);
22        orden.vehmarmodNombre = marca ? marca.vehmarmodNombre : null;
23        return orden;
24      });
25
26      // Devuelve los datos de órdenes de trabajo con nombres de marcas añadidos
27      return ordenesConNombresDeMarcas;

```

Figura 87 Filtros para búsqueda de información global – Órdenes de Trabajo

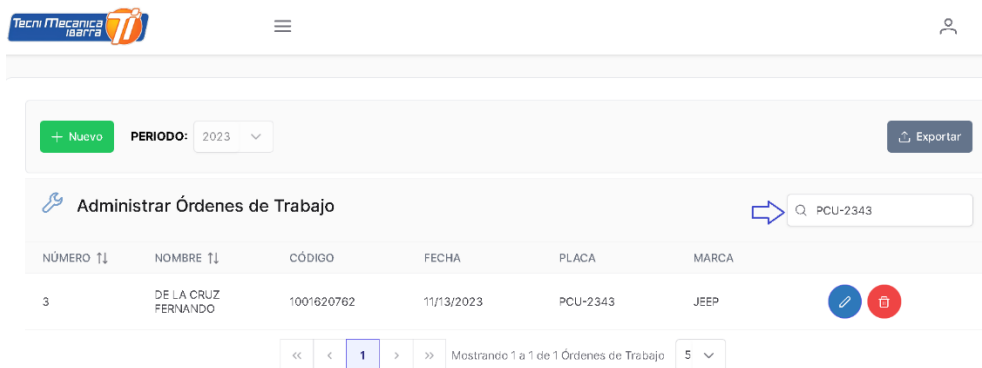


Tabla 31 Historia de usuario HU-22

Historia de Usuario			
Código:	HU-22	Usuario: Administrador	
Nombre de Historia:	Registro de Órdenes de trabajo		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	3
Descripción:	<p>El aplicativo permitirá el registro de órdenes de trabajo, incluyendo detalles del cliente, fecha, observaciones, número de placa del vehículo, kilometraje, marca y color del vehículo. Además, de permitir la selección de los servicios requeridos y el inventario completo del vehículo.</p> <p>El registro de clientes debe tener las siguientes características:</p>		

- Verificar la existencia del cliente en la base de datos. Si el cliente no está registrado, se requiere realizar su registro inicial antes de proceder.
- Sistema de marcado de casillas que permita realizar un inventariado detallado del vehículo y seleccionar los servicios específicos requeridos por el cliente.

Resultados HU-22

Figura 88 Registro de Orden de Trabajo – Inventariado de vehículo

Fecha: 10/29/2023

Cliente: JOSE LUIS HIDALGO REA - 1600246191

Tanque de Gasolina: 50%

Placa: AAC-0123

Color: VIOLETA

Marca: KIA

Kilometraje: 23,456

INVENTARIO VEHÍCULO

INTERIORES	EXTERIORES	ACCESORIOS
<input checked="" type="checkbox"/> Alarma 13332 <input type="checkbox"/> Perillas <input type="checkbox"/> Luz Check <input checked="" type="checkbox"/> Gata-Llave-Palanca <input checked="" type="checkbox"/> Llaves Vehículo <input checked="" type="checkbox"/> Espejo Int.	<input checked="" type="checkbox"/> Nro. Placas 2 <input type="checkbox"/> Nro. Plumas <input type="checkbox"/> Nro. Espejos Exteriores <input type="checkbox"/> Nro. Tapacubos <input type="checkbox"/> Seguro de Ruedas <input type="checkbox"/> Antena	<input type="checkbox"/> Memory USB

Figura 89 Registro de Orden de Trabajo – Servicios requeridos

SERVICIOS

Aceite Motor
 Aceite Caja
 Aceite Transfer
 Aceite Diferencial Posterior
 Aceite Diferencial Delantero

Aceite

- Aceite Motor
- Aceite Caja
- Aceite Transfer
- Aceite Diferencial Posterior
- Aceite Diferencial Delantero

Figura 90 Registro de Orden de Trabajo – Daños y observaciones

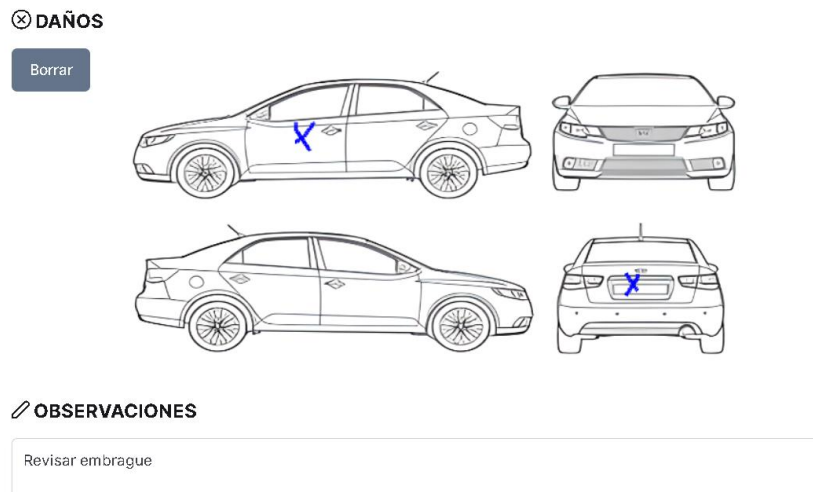


Tabla 32 Historia de usuario HU-23

Historia de Usuario			
Código:	HU-23	Usuario: Administrador	
Nombre de Historia:	Modificación de Órdenes de trabajo		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	3
Descripción:			
El aplicativo permitirá actualizar la información del inventario de vehículo y los servicios registrados en las órdenes de trabajo.			
La actualización de un registro de orden debe permitir:			
<ul style="list-style-type: none"> - Acceder al registro y editar la información de inventario de vehículo y servicios, a través del icono de lápiz asociado a cada registro específico. - Guardar cambios en la base de datos. 			

Resultados HU-23

Figura 91 Edición Orden de Trabajo

Figura 92 Consulta – Orden de Trabajo actualizada

NÚMERO	NOMBRE	CÓDIGO	FECHA	PLACA	MARCA
6	JOSE LUIS HIDALGO REA	1600246191	10/29/2023	AAC-0123	VOLVO

Tabla 33 Historia de usuario HU-24

Historia de Usuario			
Código:	HU-24	Usuario: Administrador	
Nombre de Historia:	Eliminación de Órdenes de trabajo		
Prioridad:	Alta	Esfuerzo en desarrollo:	Medio
Esfuerzo estimado:	2 días	Sprint asignado:	3
Descripción:	La aplicación permitirá eliminar registros de órdenes de trabajo.		
Observaciones:	Implementar un proceso de confirmación para evitar eliminaciones accidentales por parte de los usuarios.		

Resultados HU-24

Figura 93 Eliminación Orden de Trabajo

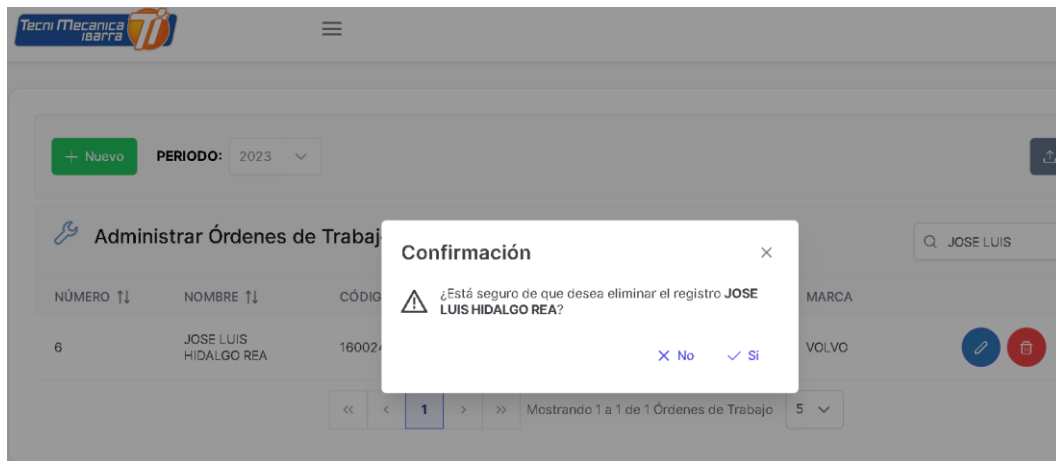
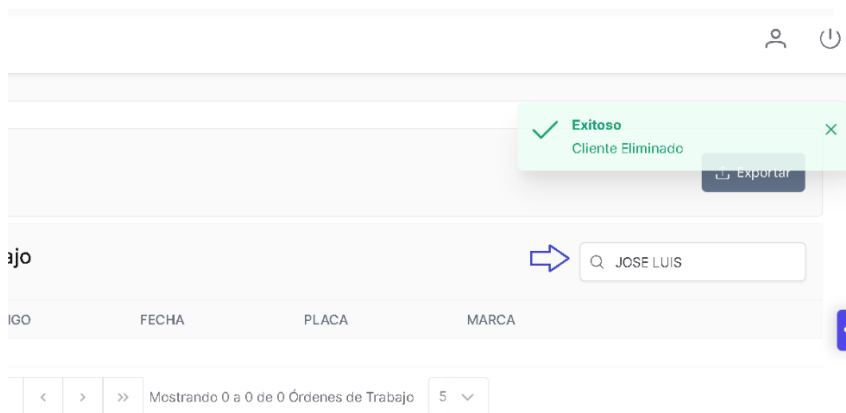


Figura 94 Notificación – Orden de Trabajo eliminada



4.3.3 Pruebas unitarias Sprint 3

Se realizarán pruebas de funcionalidad a lo largo del desarrollo del aplicativo para detectar errores de manera temprana y facilitar su solución, asegurando así que todas las partes de la aplicación se integren de manera gradual.

Tabla 34 Pruebas unitarias – Sprint 3

Caso de prueba	Resultados esperados	Resultados reales	Estado
Validar consulta de artículos con filtros personalizados (Período 2018 y Código 217)	El sistema presenta la información del artículo que coincide con los parámetros especificados	El sistema presentó la información del artículo que se ajusta los criterios definidos	Exitoso

Validar la generación correcta de reportes de existencias, asegurando la coherencia de los datos según los parámetros establecidos.	El sistema genera un reporte consistente acorde a las ventas que cumplen con los parámetros especificados.	Generación satisfactoria del reporte. Lista las ventas integras esperadas.	Exitoso
Validar el registro de orden de trabajo de prueba	El sistema registra la orden de trabajo en la base de datos.	Registro satisfactorio de la orden de compra de prueba en la base de datos.	Exitoso
Evaluar actualización de la orden de trabajo de prueba.	El sistema actualiza la información de la orden de trabajo en la base de datos.	Actualización satisfactoria de la orden de trabajo de prueba en la base de datos.	Exitoso
Evaluar eliminación de la orden de trabajo de prueba.	El sistema elimina el registro de la orden de trabajo de la base de datos.	Eliminación satisfactoria de la orden de trabajo de prueba de la base de datos.	Exitoso

4.3.4 Sprint Review 3

Tabla 35 Reunión sprint 3

Reunión del sprint 3	
Propósito	Validar el desarrollo del tercer sprint con el administrador del taller mecánico, quien es el principal interesado en el aplicativo debido a sus dificultades con el manejo del sistema actual.
	Stakeholder: Diego Quishpe
Asistentes	Desarrollador: Noelia Quishpe
Duración	1 hora
Entradas	Funcionalidad módulo inventario y órdenes trabajo.
Salidas	Retroalimentación

Agenda	<ol style="list-style-type: none">4. Presentación del avance de desarrollo Sprint 3.5. Evaluación del Product Backlog General y Sprint Backlog 3:<ul style="list-style-type: none">- Validación de lo cumplido según lo detallado en las Historias de Usuario.6. Retroalimentación de la parte interesada:<ul style="list-style-type: none">- Se especificó y aclaró de mejor manera como organizar el formulario de ingreso de la orden de trabajo.
---------------	--

4.3.5 Sprint Retrospective 3

¿Qué salió bien?

- Las pruebas unitarias del tercer sprint resultaron exitosas, lo que permite contar con un aplicativo completamente funcional alineado a los requerimientos del cliente.
- El módulo de inventario y órdenes de trabajo se desarrollaron sin mayores complicaciones, proporcionando todas las funcionalidades necesarias detalladas en los requerimientos.
- Las interfaces de acceso al sistema, inventario y órdenes de trabajo fueron diseñadas según las especificaciones del cliente, asegurando cumplimiento de sus requisitos.

¿Qué salió mal?

- La falta de experiencia en la manipulación de componentes Vue HTML para la creación de una interfaz agradable para el ingreso, modificación, eliminación de órdenes de compra retrasó ligeramente el desarrollo.
- Dado que se integraron múltiples APIs para la consulta del inventario y se manejaba una gran cantidad de datos, las consultas resultaban lentas. Por lo que se implementó la técnica de programación paralela en los métodos de llamada a las APIs para optimizar el rendimiento y mejorar la velocidad de respuesta del sistema.

Propósito de mejora

- Como trabajo futuro se propone la mejora de las interfaces de órdenes de trabajo y la implementación de programación con GPU para optimizar el tiempo de ejecución de las consultas, considerando el manejo de grandes volúmenes de datos del sistema.

4.4 Pruebas de Integración

Tras la revisión individual de los módulos en cada iteración , es fundamental, además, validar su funcionamiento en conjunto con otras entidades significativas del aplicativo. Esto es particularmente relevante en las Órdenes de Trabajo, ya que su registro implica la interacción con entidades claves del Backend como clientes, marcas, colores, servicios e inventario de vehículo.

Tabla 36 *Caso de Prueba #1 PI - Orden de Trabajo*

Caso de Prueba #1 PI	
Objetivo del Caso de Prueba	Asegurar el correcto funcionamiento del módulo de Órdenes de trabajo en relación con las entidades del Backend, como cliente, marca, color, inventariado de vehículos y servicios.
Nombre del Caso	Ingreso de Órdenes de trabajo
Precondiciones	<ul style="list-style-type: none"> • El usuario administrador debe tener una cuenta activa.
Paso	Resultado Esperado
Ingresar al sistema	Registro exitoso. Ingreso al aplicativo web como usuario Administrador.
Elegir la opción Modulo Órdenes de Trabajo	<ul style="list-style-type: none"> • Presentar la interfaz de Órdenes de Trabajo . • El sistema lista los registros de Órdenes de Trabajo de la base de datos.
Ingresar datos de la Orden	<ul style="list-style-type: none"> • Ingresar datos generales de la Orden, como placa, observaciones y kilometraje del vehículo.
Seleccionar Cliente	<ul style="list-style-type: none"> • Traer todos los Clientes del periodo actual. • Presentar el código y el nombre del cliente en un combo box que posibilite el filtrado por selección y texto. • Seleccionar uno de los clientes listados en el combo box.
Seleccionar Marca	<ul style="list-style-type: none"> • Traer todas las Marcas del periodo actual.

	<ul style="list-style-type: none"> • Presentar la descripción de las marca en un combo box que posibilite el filtrado por selección y texto. • Seleccionar una de las marcas listadas en el combo box.
Seleccionar Color	<ul style="list-style-type: none"> • Traer todos los Colores del periodo actual. • Presentar la descripción de los colores en el en un combo box que posibilite el filtrado por selección y texto. • Seleccionar uno de los colores listados en el combo box.
Marcar inventariado de Vehículo	<ul style="list-style-type: none"> • Recuperar registros de inventario de vehículos asociados a su respectiva categoría, que incluyen detalles sobre interiores, exteriores y accesorios. • Mostrar la descripción de estos registros acompañados de una casilla de selección y, en caso necesario, una caja de texto. • Marcar y anotar, en caso de que el vehículo cuente con estas características.
Seleccionar Servicios	<ul style="list-style-type: none"> • Traer todos los Servicios del periodo actual. • Presentar la descripción de los servicios en un combo box múltiple que posibilite el filtrado por selección y texto de varios servicios. • Seleccionar uno o más servicios de los colores del combo box múltiple.
Clic en el botón guardar	Al hacer clic en el botón de guardar, se verifica que el formulario esté completo y válido. Si cumple con los requisitos, se procede con el registro exitoso de la orden.
Resultado Real	Ingreso de registro de orden exitoso de acuerdo con el cliente, marca, color, inventariado, y servicios seleccionados.
Resultado de la Prueba	Exitoso.

4.5 Pruebas de Integridad de Datos

Es esencial garantizar que los procedimientos de consulta y generación de reportes mantengan la precisión y coherencia de los datos almacenados a lo largo del tiempo en la base de datos. Estas verificaciones se ejecutarán de manera específica en el módulo de Ventas e Inventario.

Tabla 37 Caso de Prueba #1 PIN - Ventas

Caso de Prueba #1 PIN	
Objetivo del Caso de Prueba	Asegurar el cumplimiento del principio de integridad de datos en el módulo de Ventas, tanto en las consultas como en la generación de reportes dentro de un rango de fechas específico.
Nombre del Caso	Consultas por parámetro y generación de reportes de Ventas.
Precondiciones	<ul style="list-style-type: none"> • El usuario administrador debe tener una cuenta activa.
Paso	Resultado Esperado
Ingresar al sistema	Registro exitoso. Ingreso al aplicativo web como usuario Administrador.
Elegir la opción Modulo Ventas ✓ Filtrar por periodo ✓ Filtrar por parámetros	<ul style="list-style-type: none"> • Presentar la interfaz de Ventas. • El sistema lista los registros de ventas del periodo por defecto (2023) de la base de datos. • El administrador selecciona el periodo de consulta. • El administrador ingresa un valor tipo parámetro en la caja de consulta.
Seleccionar Venta para consulta	<ul style="list-style-type: none"> • El administrador selecciona la venta que cumple con los parámetros establecidos. • El sistema verifica la información de la venta utilizando el periodo, el número de factura y la placa del vehículo, asegurando la devolución del registro consistente correspondiente a dichos parámetros. • El sistema presenta la factura de venta, que incluye datos del mantenimiento, como información del cliente, detalles del vehículo, la cabecera de la factura y su desglose (servicios y repuestos).
Generar reporte general	<ul style="list-style-type: none"> • El administrador hace clic en la ventana emergente de reportes, ingresando el rango de fechas para obtener el registro de ventas correspondientes a ese periodo de tiempo.

	<ul style="list-style-type: none"> • El sistema verifica la presencia de registros de ventas utilizando la fecha de inicio y fecha de fin, garantizando la devolución de registros coherentes correspondientes a dichos parámetros. • Si existen registros, el sistema genera y descarga automáticamente el reporte para el usuario. • En caso de ausencia de registros, el sistema informa que no se han registrado ventas durante el periodo de tiempo especificado.
Generar reporte por Cliente	<ul style="list-style-type: none"> • El administrador hace clic en la ventana emergente de reportes, ingresando el rango de fechas y el código del cliente para obtener el registro de ventas del cliente correspondientes a ese periodo de tiempo. • El sistema verifica la presencia de registros de ventas utilizando el código del cliente, la fecha de inicio y fecha de fin, garantizando la devolución de registros coherentes correspondientes a dichos parámetros. • Si existen registros, el sistema genera y descarga automáticamente el reporte para el usuario. • En caso de ausencia de registros, el sistema informa que no se han registrado ventas del cliente deseado durante el periodo de tiempo especificado.
Resultado Real	<ul style="list-style-type: none"> • Consulta y generación de reportes cumple con el principio de integridad de la data.
Resultado de la Prueba	<ul style="list-style-type: none"> • Exitoso.

Tabla 38 *Caso de Prueba #2 PIN - Inventario*

Caso de Prueba #2 PIN	
Objetivo del Caso de Prueba	Asegurar el cumplimiento del principio de integridad de datos en el módulo de Inventario, tanto en las consultas como en la generación de reportes de existencia dada una fecha de corte.
Nombre del Caso	Consultas por parámetro y generación de reportes de Existencia de Artículos.

Precondiciones	<ul style="list-style-type: none"> • El usuario administrador debe tener una cuenta activa.
Paso	Resultado Esperado
Ingresar al sistema	Registro exitoso. Ingreso al aplicativo web como usuario Administrador.
Elegir la opción Modulo Inventario ✓ Filtrar por periodo ✓ Filtrar por categoría ✓ Filtrar por parámetros	<ul style="list-style-type: none"> • Presentar la interfaz de Inventario. • El sistema lista los registros de artículos del periodo por defecto (2023) de la base de datos. • El administrador selecciona el periodo de consulta. • El administrador selecciona la categoría de consulta. • El administrador ingresa un valor tipo parámetro en la caja de consulta.
Seleccionar Articulo para consulta	<ul style="list-style-type: none"> • El administrador selecciona el artículo que cumple con los parámetros establecidos. • El sistema verifica la información de inventario del artículo, utilizando el periodo, el código del artículo y el id de la bodega, asegurando la devolución del registro consistente correspondiente a dichos parámetros. • El sistema presenta los movimientos del artículo (Kardex), costo, precio de venta y existencia.
Generar reporte general	<ul style="list-style-type: none"> • El administrador hace clic en la ventana emergente de reportes, ingresando la fecha de corte para obtener el registro de existencias de artículos correspondientes hasta la fecha establecida. • El sistema verifica la presencia de registros de ventas utilizando la fecha de corte y asegurar la devolución de registros coherentes correspondientes a dichos parámetros. • Si existen registros, el sistema genera y descarga automáticamente el reporte para el usuario. • En caso de ausencia de registros, el sistema informa que no hay existencias de los artículos hasta la fecha establecida.
Generar reporte por Categoría	<ul style="list-style-type: none"> • El administrador hace clic en la ventana emergente de reportes, ingresando la fecha de corte y seleccionando la categoría para obtener el registro de ventas del cliente correspondientes a

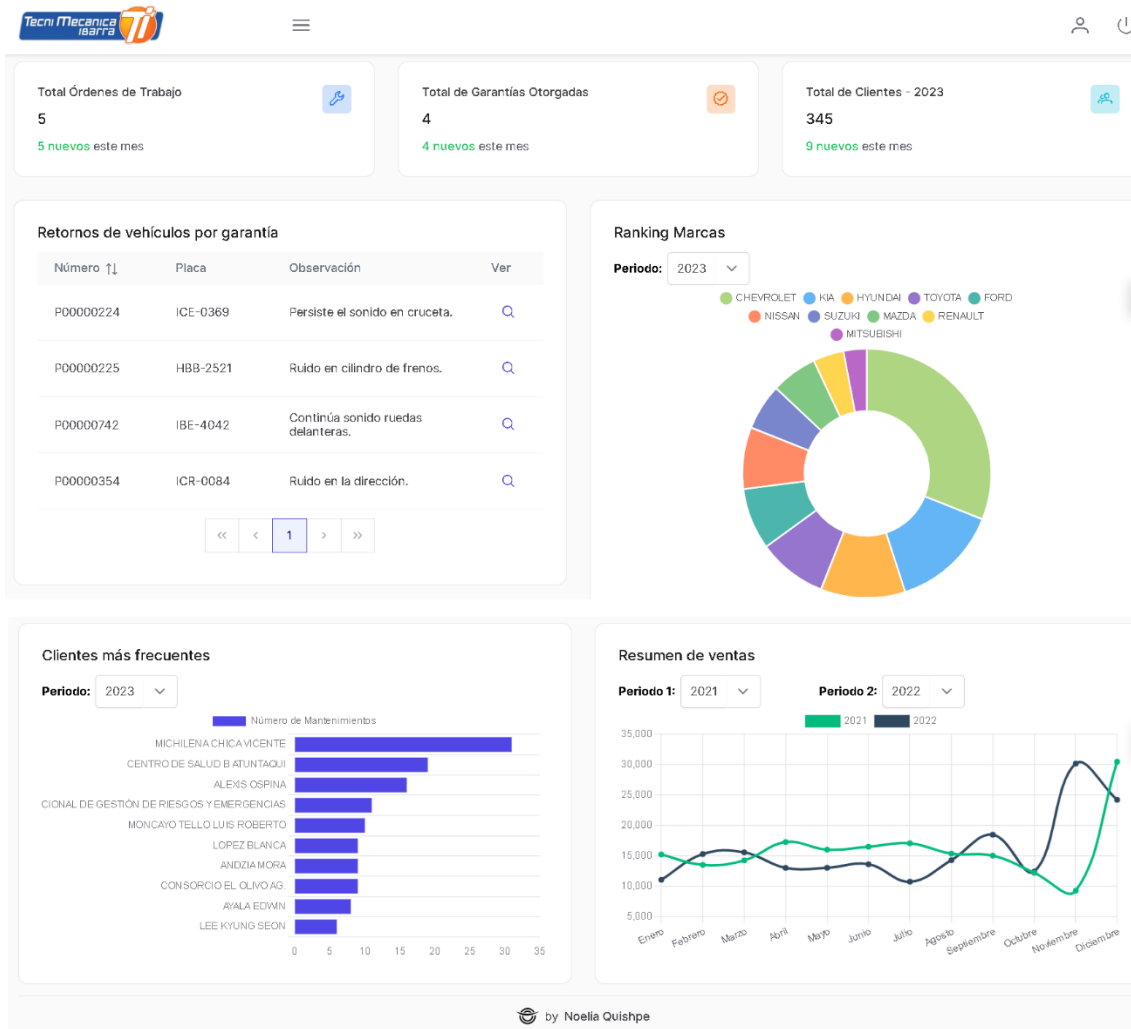
	<p>ese periodo de tiempo.</p> <ul style="list-style-type: none"> • El sistema verifica la presencia de registros de ventas utilizando el código de la categoría, y la fecha de corte, garantizando la devolución de registros coherentes correspondientes a dichos parámetros. • Si existen registros, el sistema genera y descarga automáticamente el reporte para el usuario. • En caso de ausencia de registros, el sistema informa que no hay existencias de los artículos pertenecientes a la categoría seleccionada hasta la fecha establecida.
Resultado Real	<ul style="list-style-type: none"> • Consulta y generación de reportes cumple con el principio de integridad de la data.
Resultado de la Prueba	<ul style="list-style-type: none"> • Exitoso.

Página de Inicio del Aplicativo

Siguiendo la planificación, se han cumplido todos los requisitos propuestos para la fase de desarrollo y pruebas del sistema. Como valor agregado, se ha decidido incorporar un panel de control simple en la página principal del aplicativo.

Es importante señalar que esta adición no estaba inicialmente contemplada en el alcance del proyecto, pero se considera beneficiosa como aporte adicional, ya que ofrece información relevante al administrador.

Figura 95 *Página Principal del Aplicativo*



4.6 Plan de Carga de Datos para el Aplicativo de Consultas

La carga de datos en el sistema de consulta se ha configurado como un proceso programado en el administrador de tareas. Este proceso se ejecutará diariamente a las 7 a.m., utilizando los datos ingresados del día anterior. La programación se ha establecido para asegurar la coherencia de la información, ya que durante las noches se realizan respaldos diarios de la base de datos.

Figura 96 Proceso ETL Diario: Carga de Datos en el Sistema de Consultas

```

ETL_SP_ACTUALIZA_INFOR_DIARIA: Created: 11/19/2023 12:18:42 PM Last DDL: 11/19/2023 12:18:42 PM
Source Arguments Deps (Uses) Deps (Used By) Errors Grants Synonyms Auditing

CREATE OR REPLACE PROCEDURE "ETL_SP_ACTUALIZA_INFOR_DIARIA" (p_ano VARCHAR2)
AS
BEGIN
INSERT INTO inv_maearticulo
(ano, art_codigo, com_codigo, grup_codigo, art_nombre,
art_nombrec, grup_codigop, art_tributaiva, art_tributaice,
art_fechaing, art_laboratorio, art_ubicacion,
art_multiunidad, art_unidadventa, art_unidadcosteo,
art_caduca, art_codigoalt1, art_codigoalt2, art_codigoalt3,
art_codigoalt4, art_largo, art_altura, art_ancho, art_peso,
art_compuesto, art_cantmin, art_cantmax, con_codigoact,
con_codigogas, con_codigoing, con_codigooord1, con_codigoord2,
art_observacion, art_codigoice, art_marca, art_modelo,
art_serie, art_volumen, presentacion_codigo, art_factor,
art_flag, art_formavta, art_descuento, art_serialflag,
art_diasgarantia, art_valorice, art_inactivo, bod_codigo,
art_secbodtrans, ven_codigo)
SELECT p_ano, art_codigo, com_codigo, grup_codigo, art_nombre,
art_nombrec, grup_codigop, art_tributaiva, art_tributaice,
art_fechaing, art_laboratorio, art_ubicacion, art_multiunidad,
art_unidadventa, art_unidadcosteo, art_caduca, art_codigoalt1,
art_codigoalt2, art_codigoalt3, art_codigoalt4, art_largo,
art_altura, art_ancho, art_peso, art_compuesto, art_cantmin,
art_cantmax, con_codigoact, con_codigogas, con_codigoing,
con_codigooord1, con_codigoord2, art_observacion, art_codigoice,
art_marca, art_modelo, art_serie, art_volumen,
presentacion_codigo, art_factor, art_flag, art_formavta,
art_descuento, art_serialflag, art_diasgarantia, art_valorice,
art_inactivo, bod_codigo, art_secbodtrans, ven_codigo
FROM tctr_23.inv_maearticulo a
WHERE a.art_codigo NOT IN (
SELECT b.art_codigo
FROM inv_maearticulo b
WHERE a.art_codigo = b.art_codigo

```

Figura 97 Configuración para la Ejecución Diaria del Proceso ETL de Carga de Datos

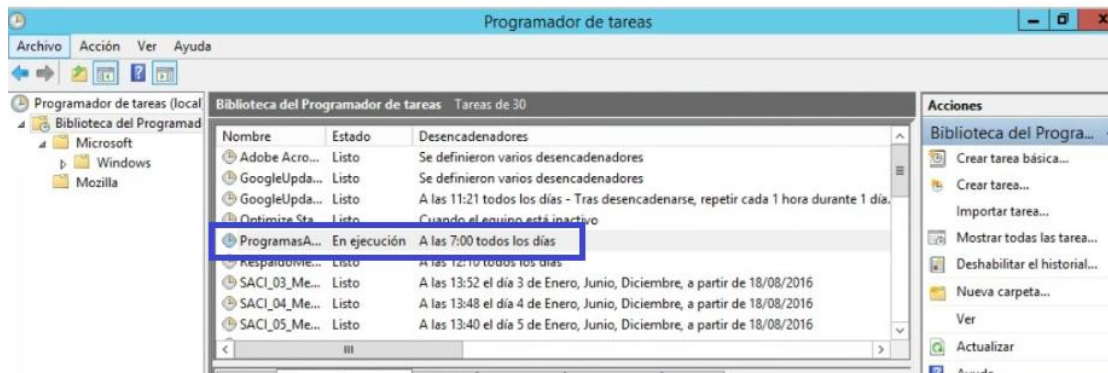


Figura 98 Supervisión del Proceso en Ejecución: Administrador de Tareas

Nombre	Estado	Administrador	Memoria	Descripción
notepad.exe	En ejecución	Administrador	1,068 K	Bloc de notas
notepad.exe	En ejecución	Administrador	1,068 K	Bloc de notas
oracle.exe	En ejecución	SYSTEM	1,722,332 K	Oracle RDBMS Kernel Executable
oracle.exe	En ejecución	SYSTEM	9,280 K	Oracle RDBMS Kernel Executable
Proceso inactivo del sistema	En ejecución	SYSTEM	4 K	Porcentaje de tiempo de inactividad
PrdLiastMonitor.exe	En ejecución	SYSTEM	1,406 K	HP ProLiant Monitor Service
PrySaciProcAutomaticos.exe	En ejecución	Administrador	9,848 K	PrySaciProcAutomaticos
remoting_desktop.exe	En ejecución	SYSTEM	2,744 K	Proceso de integración del escritorio
remoting_host.exe	En ejecución	SYSTEM	2,828 K	Proceso de host
remoting_host.exe	En ejecución	SERVICIO L...	27,076 K	Proceso de host
ReportingServicesService.exe	En ejecución	SYSTEM	384,424 K	Reporting Services Service
rotatologs.exe	En ejecución	SYSTEM	680 K	HP System Management Homepage

CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES

Conclusiones

1. El aplicativo de consulta, desarrollado y mejorado en cada sprint, ha logrado centralizar y organizar la información relacionada con los servicios de mantenimiento vehicular y la gestión de recursos, facilitando la consulta de datos relevantes al administrador del taller mecánico.
2. La selección de las herramientas de desarrollo, basada en un análisis de las necesidades y criterios del taller mecánico, fue fundamental para establecer un ecosistema de trabajo adecuado y altamente compatible con las tecnologías que maneja la empresa.
3. La codificación del Frontend y Backend en componentes independientes redujo la complejidad de la administración en los módulos desarrollados. Los cambios efectuados en una parte del código no afectaron directamente a la otra, lo que resultó en un proceso de desarrollo ágil y menos propenso a errores.
4. El uso de librerías y componentes nativos del framework Vue.js, ha permitido optimizar el diseño de las interfaces y el manejo de adaptabilidad, permitiendo que la aplicación web presente la información necesaria de forma adecuada en cualquier dispositivo, cumpliendo con el principio de diseño minimalista, caracterizado por su simplicidad y limpieza, así como con el diseño responsive, que adapta una páginas web a distintos tamaños de pantalla.
5. Las pruebas unitarias realizadas en cada una de las iteraciones de la fase de desarrollo desempeñaron un papel fundamental en la identificación de campos que requerían validaciones, así como en la implementación de alertas de usuario al borrar registros heredados de otra entidad. Esto permitió solucionar errores de forma temprana y garantizar la integridad de los datos.

Recomendaciones

1. El éxito de todo proyecto de desarrollo de software depende en gran medida de la selección adecuada de la metodología de trabajo. Por ello, es recomendable realizar un análisis detenido de las diferentes metodologías disponibles y elegir aquella que mejor se adapte a las necesidades específicas del proyecto, considerando aspectos como el tamaño del equipo, la experiencia de los miembros, la complejidad de la aplicación y el plazo de tiempo disponible.
2. En el caso de continuar con el desarrollo del proyecto, se recomienda implementar la gestión integral de todos los procesos del taller. Esto posibilitará que la empresa opere con un sistema moderno y eficaz, mejorando considerablemente la gestión del tiempo y optimizando las operaciones diarias.
3. Los sistemas deben ser diseñados de tal manera que se ajusten a las necesidades, habilidades y preferencias de los usuarios, en lugar de exigir que los usuarios se adapten a ellos. Por lo que resulta fundamental trabajar con los principios de usabilidad para facilitar el diseño de interfaces de usuario intuitivas y eficientes.
4. Una vez seleccionado el framework de desarrollo Frontend, es necesario revisar la documentación para identificar los componentes interactivos que mejor se adapten a las necesidades del usuario. Debido a que estos deben permitir que el usuario complete sus tareas de manera rápida y sencilla, como el llenado de formularios de registro y el manejo de filtros para consultas.
5. Cuando se manejan grandes volúmenes de datos en un sistema de gestión de consultas, es recomendable emplear técnicas de programación paralela, con el objetivo de reducir significativamente el tiempo de ejecución de las consultas, lo que a su vez mejora la eficiencia del sistema y permite proporcionar respuestas más rápidas a las solicitudes de los usuarios.

ANEXOS

Repositorio del Proyecto

Enlace al repositorio del Proyecto (Backend)

[nsquishpe/BackProyTesis: Back End - Thesis - Noelia Quishpe \(github.com\)](#)

Enlace al repositorio del Proyecto (Frontend)

[nsquishpe/FrontProyTesis: Front End - Thesis- Noelia Quishpe \(github.com\)](#)

Enlace al repositorio del Proyecto (Documentos Importantes)

[Repositorio de Documentos para Trabajo de Titulación](#)

Entorno del sistema actual que maneja TecniMecánica Ibarra

Con el propósito de comprender la importancia del desarrollo de la aplicación de consulta, se adjuntan anexos del sistema actual, evidenciando que, para un usuario novato, con escaso conocimiento tecnológico, resulta difícil de utilizar y carece de una presentación amigable.

Figura 99 Acceso al Sistema SACI por Periodo

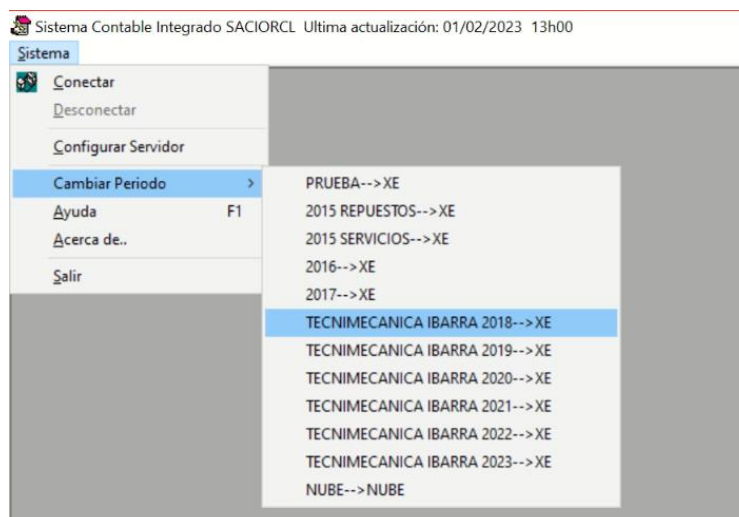


Figura 103 Interfaz Gestión de Ventas - Sistema SACI

The screenshot displays the SACI Sales Management Interface. The main window shows a sales invoice for 'CLIENTE CON CUENTAS PENDIENTES' (Client with Pending Accounts). The invoice details include:

- Estado:** PENDIENTE
- Fecha:** 19/11/2023
- Número:** P00001301
- FE:** 17/11/2023 17:57
- DC:** 0
- FV:** 17/11/2023
- CT:** 00
- TD:** 99
- Ref:** P00001301
- Comprobante Electrónico:** PLASENCIA ALFREDO
- Cliente:** 1000849370001
- Vendedor:** BODEGA EN TRANSITO (TR)
- Costo:** 12.00

The invoice items table is as follows:

T	Bod	Código	Descripción	Iva	Cant	Precio	%Disc	Total	Tco/Prov	C
1	A	TRAN BT-2073	CARSHIELD LF-D1202 Jgo Pastillas delanteras	S	1.00	19.91	2.50	17.52		
2	A	TRAN BT-2074	Juego de pastillas posteriores	S	1.00	20.73	2.50	18.24		
3	*				1.00	0.00	0.00	0.00		
4	S	001	ABC Tiras 4 ruedas	N	1.00	30.97	0.00	27.95	APF	
5	*		Cambio pastillas delanteras y posteriores		1.00	0.00	0.00	0.00		
6	*		Regular zapatas freno de mano		1.00	0.00	0.00	0.00		

The summary table at the bottom shows:

Neto	Descuento	Base I.V.A.	% I.V.A.	Base I.C.E.	% I.C.E.	Base Cero	Otros	TOTAL
63.71	0.00	35.76	4.29	0.00	0.00	27.95	0.00	68.00

On the right side, there is a search bar and a table of references:

Número	F. Emis	Referencia	Nombre Cliente	Vehículo	Color	Placa
P00001178	30/11/2023 11:59	P00001178	Diana Godoy			
P00000481	30/11/2023 11:58	P00000481	MONTESDEDI			
P00000357	30/11/2023 11:57	P00000357	LOPEZ PATRI			
P00001030	30/11/2023 8:02	P00001030	BELALCAZAR			
P00001062	30/11/2023 8:01	P00001062	Gabriela Cerv			

Figura 104 Reportes Generales - Sistema SACI

The screenshot displays the SACI General Reports Interface. The report is titled 'LISTADO DE ARTICULOS' (List of Articles) for 'TECNIMECANICA IBARRA'. The report details include:


- Fecha de Emisión:** 19/11/2023 8:27:37p.m.
- Página:** 1

The list of articles is as follows:

PART NUMBER	ARTICULO
ACP	Audifono Agojo Confiseca PROMO
AMPSP	Agua SPLENDOR PROMO Whisky
ASPBAL	Anillo Sella PROMO
AT 462A	Super Max Afetit COMFORT GRIP III x 28 + KWIK II X 12
AT 462B	Super Max Afetit COMFORT GRIP III Disp. + Promo
AXION235	AXION 235 -PROMO CP-
AXION450CPROMO	AXION 450 -PROMO CP-
AXION850CPROMO	AXION 850 -PROMO CP-
AZ A001	Azucar Tababuela Suelto 50 Kg
AZ A002	Azucar Tababuela 10 Kg
AZ A004	Azucar Tababuela 1 Kg
AZ A005	Azucar Tababuela 2 Kg x 25 u.
AZ A013	Azucar Tababuela 2 Kg
AZ A014	Azucar Tababuela 1 Kg x 25
AZ M005	Azucar Morena Tababuela 2 Kg
AZ NRS APR	Azucar 250g N.R.S.A -Promocional-
A23	Energizer Pila Litio A23 C12D
BFS5CB	Bomba Pulverización 2 lit. Promo
BOLN.R.S.A	Bolitos Amigo T endero Promo N.R.S.A
BOMBUMHIPER	Bombon -PROMOCIONAL-
BOPSM	Bolito PROXIO Something
BOT001-15	Cerveza CARLSBERG 330 ml x 24
BPROM	Balón PROMO Something
BR00931B	Plan ICE Colgate 250 ml
BR02702A	Plan Ice INFINITY 250 ml
BR03080A	Plan Kids Minions Colgate 250 ml
BR03305A	Plan Ice GLACIAL 250 ml
BR03409A	Gel Dental Colgate Kids Frutilla 50 g

Registro Manual de Órdenes de Trabajo


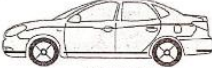

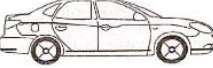
Figura 105 Órdenes de Trabajo Manuales


ORDEN DE TRABAJO
P7.

Facturar a:	Recibido a: <i>Christian Enrique</i>	Placa: <i>JAD-1956</i>
C.C./RUC:	Fecha:	Marca: <i>Great Wall</i>
Celular: <i>0958777492</i>	Tel.:	Modelo: <i>MA</i>
		Color: <i>Negro</i>
		Km.: <i>84.644</i>

INVENTARIO VEHICULO

<input type="checkbox"/> Alarma	<input type="checkbox"/> Llaves Vehic.	<input type="checkbox"/> Pito	<input type="checkbox"/> Equipo Sonido	<input type="checkbox"/> Mascarilla	<input type="checkbox"/> Memory USB	<input type="checkbox"/> Manual Usuario
<input type="checkbox"/> Parillas	<input type="checkbox"/> Espejo Int.	<input type="checkbox"/> # Cubresoles	<input type="checkbox"/> Matricula	<input type="checkbox"/> Revisión ANT	<input type="checkbox"/> Licencia Cond.	<input type="checkbox"/> Encendedor
<input type="checkbox"/> Luz Check	<input type="checkbox"/> Cenicero	<input type="checkbox"/> # Moquetas	<input type="checkbox"/> Bandeja Posterior	<input type="checkbox"/> Extintor	<input type="checkbox"/> Triángulo	<input type="checkbox"/> Botiquín
<input type="checkbox"/> Gata-Llave-Palanca	<input type="checkbox"/> Llanta Emerg.	<input type="checkbox"/> Antena	<input type="checkbox"/> Tapa Combustible	<input type="checkbox"/> # Plumas	<input type="checkbox"/> # Espejos Ext.	<input type="checkbox"/> # Tapacubos
<input type="checkbox"/> Seguro de Ruedas	<input type="checkbox"/> # Placas	<input type="checkbox"/> Emblemas	<input type="checkbox"/> Tapa Cubre Motor			

OBSERVACIONES: _____

TRABAJOS PEDIDOS POR EL CLIENTE

Revisar / cambiar aceite hidraulico

Revisar / cambiar refrigerante

Revisar suspension

Cambiar aceite y filtro motor y Aceite Caja

APC Motor, bujias y filtros

APC frenos, rotas suaves si es necesario

Cambiar faja cabina si es necesario


Alinacion y 2 balanzas

REVISAR: Nivel Refrigerante Nivel Aceite de Motor Nivel Líquido de Frenos Estado Bandas Acc.

Estado de Batería Nivel Hidraulico Direc. Funcionamiento Luces Presión Aire Llantas

- Autorizo los trabajos antes descritos y me comprometo a cancelar el monto total de la reparación incluyendo mano de obra y repuestos adicionales.
 - Autorizo a realizar fuera del taller (carreteras) las pruebas que se juzgen convenientes para asegurarse de la efectividad del trabajo.
 - No se asume responsabilidad sobre artículos no inventariados al momento de la recepción y que no consten en la sección INVENTARIO.
 - Cualquier reclamo relacionado con los servicios obtenidos deberá ser presentado dentro de un plazo máximo de 72 horas.

Av. Cristóbal de Troya 14-73, frente al redondel de la Madre.
Telf: (06)2631405 / Cel: (099)9326995
HORARIO DE ATENCIÓN
 Lunes a Viernes: 8H00-13H00 / 14H30-18H00
 Sábados: 8H00-13H00


 FIRMA CLIENTE AUTORIZA

Por TecníMecánica Ibarra

Menú de Navegación del Aplicativo

A lo largo de cada iteración del proyecto, se han adjuntado evidencias del desarrollo de las interfaces del aplicativo.

En esta sección se incluyen anexos del menú de navegación.

Figura 106 Menú de navegación

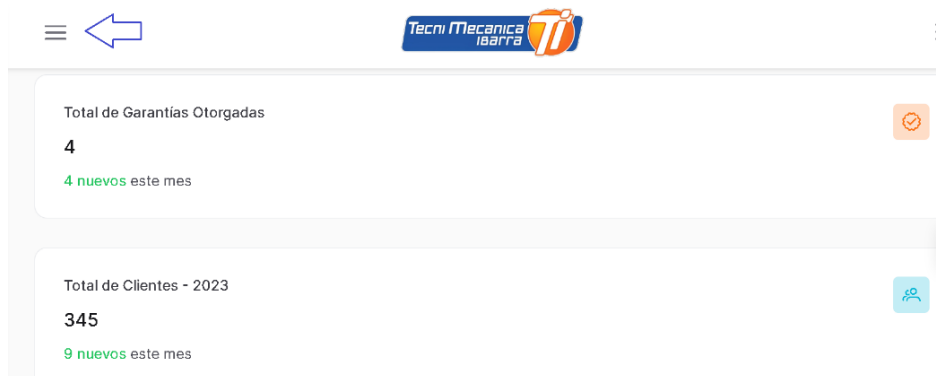
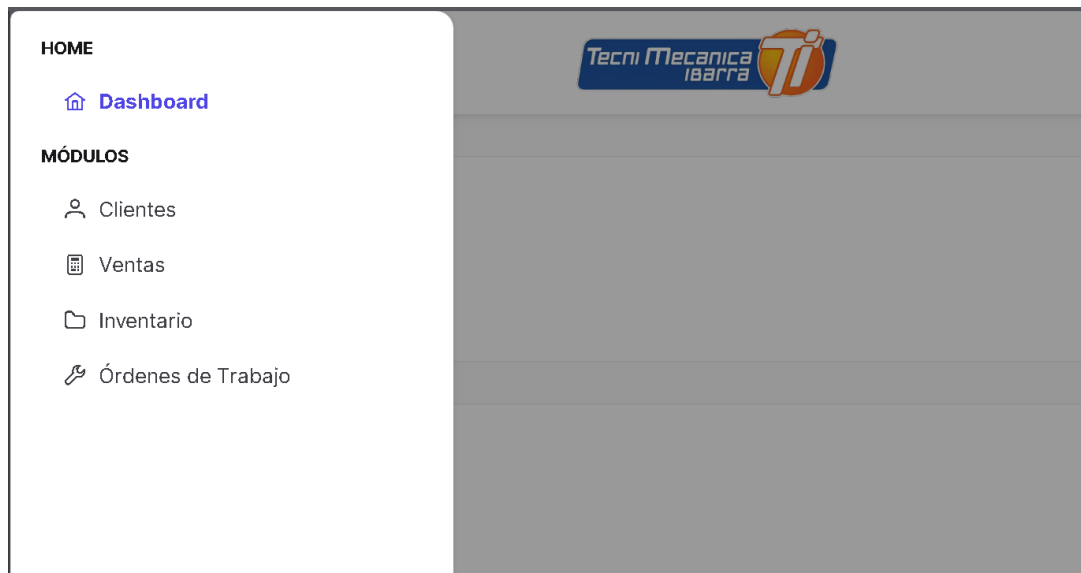


Figura 107 Páginas de Navegación - Menú desplegable



BIBLIOGRAFÍA

- AEADE. (2022). *ANUARIO-AEADE_2022*. https://www.aeade.net/wp-content/uploads/2023/03/ANUARIO-AEADE_2022_comp.pdf
- Albaladejo, X. (2022). *Planificación de la iteración (Sprint Planning)*. <https://proyectosagiles.org/planificacion-iteracion-sprint-planning/>
- Amaro, S., & Valverde, J. (2007). *Metodologías Ágiles*. Universidad Nacional de Trujillo. https://www.academia.edu/23325522/Universidad_Nacional_de_Trujillo
- Arregui, M. d. (2022). *Business School*. <https://www.obsbusiness.school/blog/que-diferencia-una-base-de-datos-relacional-vs-una-no-relacional>
- Atlassian. (2022). *¿Qué es ágil?*. <https://www.atlassian.com/es/agile>
- Avison, D., & Fitzgerald, G. (2003). *Information systems development: methodologies, techniques and tools*.
- Bonilla, E. A. (2023). *SISTEMA WEB DE GESTIÓN ADMINISTRATIVA Y APLICACIÓN MÓVIL PARA CONTROL DE SERVICIOS TÉCNICOS DEL TALLER AUTOMOTRIZ GAME*. <https://cia.uagraria.edu.ec/Archivos/DOMINGUEZ%20AGUAYO%20DARWIN%20ALEXIS.pdf>
- Cadavid, A. N., Martínez, J. D., & Vélez, J. M. (2013). Revisión de metodologías ágiles para el desarrollo de software. <https://www.redalyc.org/pdf/4962/496250736004.pdf>
- Campana, X. (2022). *¿Qué frameworks de back-end están impactando inmensamente el desarrollo de aplicaciones web?*. <https://www.itconsultors.com/que-frameworks-de-back-end-estan-impactando-inmensamente-el-desarrollo-de-aplicaciones-web>
- Carreño, A. F. (2020). *Supervisión, inspección y alimentación de la ficha técnica*. <https://repository.upb.edu.co/handle/20.500.11912/8670>
- Carrillo, D. (2009). *Diagnóstico del Sector Automotriz*. content/uploads/sites/6/2021/04/AUTOMOTRIZ-1.pdf
- Chacón, J. L. (2021). *TypeScript: qué es, diferencias con JavaScript y por qué aprenderlo*. <https://profile.es/blog/que-es-typescript-vs-javascript/>

- Drumond, C. (2022). *Atlassian*. <https://www.atlassian.com/es/agile/scrum>
- Entity Framework Tutorial. (2023). *Entity Framework Core with existing database*.
<https://www.entityframeworktutorial.net/efcore/create-model-for-existing-database-in-ef-core.aspx>
- Fielding, R. (2000). *Architectural Styles and the Design of Network-based Software Architectures*.
- González, P. R. (2008). *ESTUDIO DE LA APLICACIÓN DE METODOLOGÍAS ÁGILES PARA LA EVOLUCIÓN DE PRODUCTOS SOFTWARE*.
https://oa.upm.es/1939/1/TESIS_MASTER_PILAR_RODRIGUEZ_GONZALEZ.pdf
- ICTEA. (2023). *¿Qué es una aplicación web?*.
<https://www.ictea.com/cs/index.php?rp=/knowledgebase/4205/iQue-es-una-aplicacion-web.html>
- ITD. (2022). *Oracle Enterprise Manager*. <https://itdconsulting.com/base-de-datos/oracle/oracle-enterprise-manager/>
- Junta de Andalucía. (2023). *Manual de desarrollo en PL/SQL*.
<https://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/107>
- Maluenda, R. (2020). *Tipos de desarrollo de aplicaciones web: ejemplos y características*.
<https://profile.es/blog/desarrollo-aplicaciones-web/>
- Mendoza, M. L. (2020). *OpenWebinars*. <https://openwebinars.net/blog/scrum-y-extreme-programming-no-se-trata-de-cual-se-trata-de-como/#:~:text=Scrum%20es%20un%20marco%20de,dentro%20del%20marco%20de%20Scrum>.
- Menéndez, R. (2022). *JAVASCRIPT*.
<https://www.um.es/docencia/barzana/DAWEB/Lenguaje-de-programacion-JavaScript-1.pdf>
- Meneses, N. (2023). *Las 15 mejores herramientas de Desarrollo Web (Guía 2023)*.
<https://www.codingdojo.la/2023/01/05/las-15-mejores-herramientas-de-desarrollo-web-guia-2023/>

- Microsoft. (2022). *Herramientas web modernas*.
<https://visualstudio.microsoft.com/es/vs/features/web/>
- Microsoft. (2023). *C#*. <https://dotnet.microsoft.com/es-es/languages/csharp>
- Microsoft. (2023). *Creación de API web con ASP.NET Core*. <https://learn.microsoft.com/es-es/aspnet/core/web-api/?view=aspnetcore-7.0>
- Oracle. (2022). *¿Qué es JavaScript?*. <https://developer.oracle.com/es/learn/technical-articles/what-is-javascript>
- Oracle. (2023). *Oracle Database Express Edition 11g Release*.
<https://www.oracle.com/database/technologies/database-11g-express-edition.html>
- Párraga, A. (2023). *Top Frameworks JavaScript 2023*.
<https://soloprogramadores.com/clasificando-los-marcos-de-javascript-mas-populares-en-2023/>
- Pérez, A. (2022). *Taller mecánico: tipos y servicios de los centros de mecánica*.
<https://www.autobild.es/noticias/taller-mecanico-tipos-servicios-centros-mecanica-1111611>
- Pressman, R. (2010). *Software Engineering: A Practitioner's Approach 8th Edition*.
- RECOPE. (2020). *MANTENIMIENTO DE VEHÍCULOS - Una forma de garantizar su propia economía. Programa Conservación de Energía*. <https://www.recope.go.cr/wp-content/uploads/2012/11/FOLLETODSE.pdf>
- Reyes, I. C. (2022). *¿Qué es el Frontend y el backend?*.
<https://cognosonline.com/co/blog/que-es-Frontend-y-backend/>
- Rifi, M. (2023). *5 herramientas de desarrollo web que debes usar*.
<https://www.linkedin.com/pulse/5-herramientas-de-desarrollo-web-que-debes-usar-mohamed-rifi/?originalSubdomain=es>
- Ripoll, L. Q. (2007). *SISTEMAS DE GESTIÓN DE BASES DE DATOS*.
<https://www.eoi.es/sites/default/files/savia/documents/componente45499.pdf>
- Rodríguez, Á. (2023). *Herramientas para la administración de bases de datos*.
<https://www.studocu.com/latam/document/universidad-politecnica-territorial-de->

yaracuy-aristides-bastidas/innovaciones-tecnologicas/herramientas-para-la-administracion-de-bases-de-datos/38846724

Sánchez, J. E. (2006). *UN ANÁLISIS DEL SECTOR AUTOMOTRIZ Y SU MODELO DE GESTIÓN EN EL SUMINISTRO DE LAS AUTOPARTES.*

<https://imt.mx/archivos/publicaciones/publicaciontecnica/pt288.pdf>

Sánchez, N. C. (2023). *La importancia de las metodologías ágiles en los proyectos de desarrollo de software.*

https://idus.us.es/bitstream/handle/11441/148116/documento_trabajo_G4517.pdf?sequence=1&isAllowed=y

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide.*

<https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>

Sepúlveda, M. Á. (2023). *Frameworks en el desarrollo web: las mejores prácticas para tu negocio online.* [https://www.wearemarketing.com/es/blog/frameworks-en-el-](https://www.wearemarketing.com/es/blog/frameworks-en-el-desarrollo-web-las-mejores-practicas-para-tu-negocio-online.html#:~:text=Un%20framework%20es%20una%20herramienta,y%20Fo%20funcionalidades%20ya%20creadas.)

[desarrollo-web-las-mejores-practicas-para-tu-negocio-online.html#:~:text=Un%20framework%20es%20una%20herramienta,y%20Fo%20funcionalidades%20ya%20creadas.](https://www.wearemarketing.com/es/blog/frameworks-en-el-desarrollo-web-las-mejores-practicas-para-tu-negocio-online.html#:~:text=Un%20framework%20es%20una%20herramienta,y%20Fo%20funcionalidades%20ya%20creadas.)

Sutherland, J. (2014). *Scrum: The Art of Doing Twice the Work in Half the Time.*

Toad. (2023). *Quest Toad® for Oracle Subscription.* <https://toadworld.com/products/toad-for-oracle/>

Toad for Oracle. (2020). *Toad for Oracle.* <https://neuronet.cl/producto/toad-for-oracle-dba-edition/>

Tsichritzis, D., & Klug, A. (1978). *The ANSI/X3/SPARC DBMS framework report of the study group on database management systems.*

https://www.academia.edu/76677906/The_ANSI_X3_SPARC_DBMS_framework_report_of_the_study_group_on_database_management_systems

TypeScript. (2023). *What is TypeScript?.* <https://www.typescriptlang.org/>

UNAM. (2023). *enguajes de Programación.* Unidad de Apoyo para el Aprendizaje:

https://programas.cuaed.unam.mx/repositorio/moodle/pluginfile.php/1023/mod_resou/rce/content/1/contenido/index.html#contenido

UNIR. (2021). *Conceptos básicos de PL/SQL*. <https://www.unir.net/ingenieria/revista/que-es-plsql/>

Vanting, L. B. (2020). *Oracle Enterprise Manager 101*.

Wells, D. (2009). *The Values of Extreme Programming*.
<http://www.extremeprogramming.org/values.html>