| http://www.duoc.cl/sites/default/files/logo_summit_0.png |
| --- |
| Informe Final  Desarrollo de Ingeniería |
|  |
|  |

# **Índice**

[**Índice 1**](#_zgw2to6o4cir)

[**Acta de constitución 3**](#_c7zx5wirygzz)

[Información del proyecto **3**](#_wbnt7bfi3i07)

[Datos del documento **3**](#_30j0zll)

[Información del Proyecto 3](#_1fob9te)

[Propósito y justificación del proyecto **4**](#_2et92p0)

[Descripción del proyecto y entregables 4](#_vg9e4zgqccgi)

[**Requerimientos de software 8**](#_isd34pr91zjw)

[**Metodología 12**](#_xj2gz4moto2w)

[Historias de usuario 13](#_eoi42c43ty53)

[Product Backlog 15](#_wvzrfarl4t4)

[**Plan de proyecto 17**](#_t2ww8wof9dd0)

[Propósito del plan de proyecto 17](#_xqzzxopln5mw)

[Alcance del proyecto 17](#_tyjcwt)

[**Definición de roles y responsabilidades 18**](#_1t3h5sf)

[Estructura de Desglose de trabajo 20](#_2s8eyo1)

[Calendarización de las actividades 22](#_17dp8vu)

[Resumen de riesgos 23](#_3rdcrjn)

[**Arquitectura de software-Documento DAS 26**](#_sgd1cww629hk)

[Vistas de la arquitectura 29](#_2lwamvv)

[UML-VISTA DE ESCENARIO 29](#_111kx3o)

[UML-VISTA LOGICA 30](#_3l18frh)

[UML-VISTA DE PROCESO 35](#_4k668n3)

[UML-VISTA FISICA 37](#_2zbgiuw)

[UML-VISTA DE DESPLIEGUE 38](#_vqzwdm5htidb)

[MODELO DE DATOS(BD) 39](#_la4hxqeq91ai)

[Usabilidad y Calidad de software 42](#_3ygebqi)

# 

# 

# 

# 

# Acta de constitución

## **Información del proyecto**

## **Datos del documento**

Histórico de Revisiones

| Versión | Fecha | Descripción/cambio | autor |
| --- | --- | --- | --- |
| 1.0 | 19/08/24 | Versión inicial | Dueño del proyecto |
| 1.1 | 20/08/24 | Actualización de cronograma | Equipo de desarrollo |
| 1.2 | 25/11/24 | Revisión final antes del cierre | Dueño del proyecto |

## **Información del Proyecto**

| Organización | PYME de repuestos de autos |
| --- | --- |
| Proyecto (Nombre) | Página web de venta de repuestos |
| Fecha de Inicio | 19/08/24 |
| Fecha de Término | 20/11/24 |
| Patrocinador principal | (Dueño de la empresa) |

Integrantes

| Rut | Nombre | Correo |
| --- | --- | --- |
| **20.468.503-7** | **Javiera Montano** | **ja.montanouocuc.cl** |
| **20.847.873-7** | **Guido Altamirano** | **gu.altamirano@duocuc.cl** |
| **21.128.589-3** | **Dilan Campos** | **dil.camposr@duocuc.cl** |

# 

## **Propósito y justificación del proyecto**

| El propósito de este proyecto es desarrollar una plataforma web que facilite la compra y venta de repuestos de autos, permitiendo a los clientes buscar, comparar y adquirir repuestos de forma rápida y segura. La plataforma también permitirá gestionar inventarios, procesar pagos en línea, y ofrecer asistencia a clientes mediante soporte en línea. Este sistema busca mejorar la experiencia de compra, optimizando el tiempo y reduciendo los costos asociados a la compra física de repuestos. |
| --- |

# 

## **Descripción del proyecto y entregables**

| El sistema de repuestos de autos se desarrollará como una página web accesible tanto desde computadoras como dispositivos móviles. Los usuarios podrán crear cuentas, buscar repuestos por marca y modelo, realizar compras y gestionar sus órdenes. Los administradores podrán cargar productos, actualizar inventarios y gestionar pagos.  **Entregables**   * Página web funcional para la venta de repuestos de autos. * Funcionalidad de registro y autenticación de usuarios. * Sección para la gestión de inventarios. * Implementación de pasarelas de pago seguras. * Soporte en línea para consultas y dudas de los clientes. |
| --- |

**Requerimientos de alto nivel**

**Requerimientos del producto**

| RAP1: Registro de usuario para realizar compras.  RAP2: Funcionalidad de búsqueda y filtrado de repuestos.  RAP3: Plataforma de pago en línea.  RAP4: Integración con proveedores de repuestos para la actualización del inventario. |
| --- |

## 

**Requerimientos del proyecto**

| RDP1: El sistema debe estar disponible 24/7, con mínimo tiempo de inactividad.  RDP2: Se debe contar con un diseño responsivo que funcione correctamente en dispositivos móviles y de escritorio.  RDP3: El sistema debe permitir la gestión de inventarios en tiempo real.  RDP4: Debe garantizarse la seguridad de las transacciones mediante la implementación de protocolos de seguridad como HTTPS. |
| --- |

**Objetivos**

| **Objetivo** | | **Indicador de éxito** |
| --- | --- | --- |
| **Alcance** | | |
| Desarrollar una plataforma web de repuestos de autos funcional y fácil de usar | | Aumento en la cantidad de ventas mensuales a través del sitio web |
| Mejorar la gestión del inventario y las transacciones | | Reducción del tiempo de procesamiento de pedidos y mayor precisión en el inventario |
|  | |  |
| **Cronograma (Tiempo)** | | |
| **Iteración** | | **Fechas** |
| Iteración 1: Diseño de interfaz de usuario y bases de datos | | 1° a 3° semana |
| Iteración 2: Desarrollo de funciones principales (búsqueda, carrito de compras, pagos) | | 4° a 6° semana |
| Iteración 3: Implementación de seguridad y pruebas de funcionalidad | | 7° a 9° semana |
| Iteración 4: Despliegue y pruebas finales | | 10° a 12° semana |
| **Costo** | | |
|  |  | |
| **Calidad** | | |
| Producto con alta usabilidad, facilidad de uso, de aprendizaje e intuitivo | | Experiencias de usuario, opiniones y reseñas. |
| Disponibilidad para las plataformas Mobile y Escritorio | | Debe ser responsivo para funcionar en distintas plataformas |
| Alta disponibilidad para el usuario | | El usuario al conectarse en la app o sitio web no debe tener problemas de navegación (el sistema no se debe caer) |

**Premisas y restricciones**

| El proyecto se basará en tecnologías de código abierto para reducir costos.  El sistema debe ser accesible tanto en plataformas móviles como de escritorio.  El desarrollo debe estar listo dentro de los 3 meses estipulados para cumplir con los plazos. |
| --- |

**Riesgos iniciales de alto nivel**

| Riesgo 1: Posibles retrasos en la entrega de los componentes de software necesarios para la implementación de pagos en línea.  Riesgo 2: Problemas de seguridad cibernética durante el procesamiento de pagos en línea. |
| --- |

# 

**Cronograma de hitos principales**

| **Hito** | **Fecha tope** |
| --- | --- |
| Análisis de Requerimientos | 14/08/24 - 18/08/24 |
| Diseño de la Interfaz de Usuario | 19/08/24 - 31/08/24 |
| Desarrollo Backend (Gestión de Inventarios, Carrito y Pago) | 01/09/24 - 15/09/24 |
| Integración y Desarrollo Frontend | 16/09/24 - 30/09/24 |
| Pruebas del Sistema (Usabilidad y Funcionalidad) | 01/10/24 - 15/10/24 |
| Optimización para Móviles y Desktop | 16/10/24 - 31/10/24 |
| Pruebas de Seguridad y Carga | 01/11/24 - 07/11/24 |
| Lanzamiento Beta (Prueba Interna) | 08/11/24 - 14/11/24 |
| Lanzamiento Final del Proyecto | 15/11/24 - 30/11/24 |

**Lista de Interesados (stakeholders)**

| **Nombre** | **Cargo** | **Departamento / División** | **Rama ejecutiva (Vicepresidencia)** |
| --- | --- | --- | --- |
|  | Propietario y Fundador | Dirección General | Dirección General |
| Cliente Final (Conductores y Talleres Mecánicos) | Usuarios de la plataforma | - | - |
| Desarrollador Web (Guido Altamirano, Dilan Campos, Javiera Montano) | Desarrollador y Gestor del Proyecto | Tecnología y Desarrollo | - |
| Proveedores de Repuestos | Proveedor de productos | Logística y Suministros | - |

# 

**Requisitos de aprobación del proyecto**

| El proyecto debe cumplir con los requisitos de seguridad, funcionalidad y rendimiento acordados. Las entregas deben estar documentadas y se debe garantizar la fiabilidad de las transacciones y la protección de los datos del cliente. |
| --- |

**Aprobaciones**

| **Patrocinador** | **Fecha** | **Firma** |
| --- | --- | --- |
| (Propietario) | 14/08/24 | [Firma del propietario] |
| Cliente: Talleres o Mecánicos | 01/09/24 | [Firma del cliente] |
| Proveedor de Repuestos | 15/09/24 | [Firma del proveedor] |

## 

# Requerimientos de software

1. Requerimientos de los Stakeholders.
   1. Descripción General del negocio.

La página web ofrecerá la venta de repuestos para vehículos, permitiendo a los clientes encontrar y comprar piezas de autos de manera fácil y segura. También se les brindará la opción de crear una cuenta para ver el historial de sus pedidos, lista de deseos, recibir notificaciones y acceder a promociones exclusivas.

* 1. Partes interesadas (stakeholders).
* Propietario de la tienda de repuestos: quien supervisa el desarrollo y funcionamiento de la página y asegura que cumpla con los objetivos comerciales.
* Clientes: conductores y talleres mecánicos que buscan repuestos de autos.
* Personal de ventas y soporte: quienes se encargan de la atención al cliente y de gestionar la información de ventas.
* Equipo de desarrollo: encargado del diseño, programación y mantenimiento de la página.
  1. Objetivos a nivel de negocio.
* Facilitar y optimizar la venta de repuestos de autos a través de una página web intuitiva.
* Mejorar la satisfacción del cliente mediante un sistema de compras fácil de usar y con opciones de soporte personalizadas.
* Aumentar la transparencia en los pedidos y en los procesos de compra.
  1. Proceso de negocio involucrado.

El sistema permitirá a los clientes navegar por el catálogo de productos, seleccionar los repuestos que necesitan y realizar su compra en línea. Incluye un proceso de registro de usuarios para ofrecer beneficios adicionales, como descuentos y notificaciones de stock.

* 1. Concepto de operación.

La operación de este sistema será en línea, permitiendo a los clientes acceder a la página web en cualquier momento y lugar. La página mostrará productos, detalles técnicos, precios, y contará con un sistema de pago seguro. También incluirá una sección de soporte donde los clientes puedan realizar consultas o solicitudes de ayuda.

2. Requerimientos del Sistema.

2.1 Descripción general del sistema.

Un sistema web de e-commerce que permita a la tienda vender repuestos de autos en línea. Los clientes podrán navegar en el catálogo de productos, realizar compras y recibir notificaciones sobre su pedido.

2.2 Propósito del sistema.

Ofrecer a los clientes una plataforma en línea accesible donde puedan realizar sus compras de repuestos de autos de manera segura, con acceso a información detallada de productos y a un sistema de notificaciones sobre sus pedidos y promociones.

2.3 Alcance del sistema

El sistema cubrirá todas las etapas necesarias para gestionar la venta de repuestos de autos, desde la navegación por el catálogo de productos hasta la confirmación de entrega y soporte postventa. Los usuarios tendrán acceso a una plataforma intuitiva y fácil de usar que permitirá realizar compras en línea de forma rápida y segura. Los principales elementos del alcance son: Gestión de Productos, Cuentas de Usuario y Personalización, Carrito de Compras y Proceso de Pago, Gestión de Pedidos y Logística, Soporte al Cliente y Garantías, Análisis y Reportes, Seguridad y Protección de Datos.

3. Requerimiento del software

3.1 Requerimientos Funcionales detallados

| Id | Nombre | Objetivo | Contexto | Precondición | Actores | Descripción | Excepciones | Resultado Esperado |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| RF001 | Registro de clientes y validación de Rut | Permitir a los clientes registrarse en el sistema y validar su Rut para asegurarse de que es válido y está vigente. | Gestión y validación de clientes en el sistema. | El cliente debe ingresar su Rut y demás datos de registro. | Cliente, Sistema | El cliente ingresa su información personal y su Rut, que es validado contra el registro civil. Si es válido, se registra en el sistema. | Si el Rut es inválido, el sistema no permitirá el registro y mostrará un mensaje de error. | El cliente es registrado exitosamente en el sistema y puede continuar con el proceso de compra. |
| RF002 | Consultar estado de pedidos | Permitir a los clientes y administradores consultar el estado del pedido (en proceso, enviado, entregado). | Seguimiento de pedidos en el sistema. | El cliente debe tener un pedido realizado. | Cliente, Administrador, Sistema | El cliente puede consultar el estado de su pedido en cualquier momento a través de su cuenta. El administrador también tiene acceso para gestionar los pedidos. | Si el pedido está en proceso, el sistema informará que aún no ha sido enviado. | El cliente y el administrador pueden consultar el estado actualizado del pedido. |
| RF003 | Gestionar inventario | Administrar el inventario de repuestos, actualizando el stock según las ventas y nuevos ingresos de productos. | Gestión interna de inventario. | El administrador debe estar registrado en el sistema. | Administrador, Sistema | El administrador puede agregar, editar o eliminar productos en el inventario. El sistema actualiza automáticamente el stock tras cada venta. | Si el administrador intenta agregar un producto ya existente, el sistema debe mostrar una advertencia. | El inventario se actualiza correctamente tras cualquier acción de gestión de productos. |

| Id | Descripción | Motivo | Criticidad | Tipo | Origen |
| --- | --- | --- | --- | --- | --- |
| RNF001 | Los cálculos de stock y precios deben ser comprobables y precisos. | El sistema debe contar con una alta capacidad de procesamiento y asegurar la fiabilidad de los cálculos. | Alta | Fiabilidad | Experiencia de usuario |
| RNF002 | El sistema debe presentar niveles de usabilidad que permitan a usuarios con formación básica operar el sistema. | Debe ser flexible y adaptable a usuarios sin conocimientos técnicos, para facilitar la navegación y operación. | Alta | Usabilidad | Tipo de usuario |
| RNF003 | El sistema debe garantizar una alta escalabilidad | Permite soportar eficientemente un gran volumen de usuarios concurrentes | Alta | Escalabilidad | Experiencia de usuario |

# Metodología

Para el desarrollo del proyecto de una página web de repuestos de autos para un cliente real, se eligió la metodología ágil **Scrum** debido a su flexibilidad y enfoque en entregas incrementales. Dado que trabajar con un cliente real implica posibles cambios en los requerimientos a lo largo del desarrollo, Scrum permite adaptarse a estas modificaciones sin comprometer los plazos ni la calidad del producto. Además, la organización en Sprints cortos permitió realizar entregas funcionales al final de cada ciclo, lo que facilitó recibir feedback continuo del cliente, garantizar que las funcionalidades cumplían con sus expectativas y priorizar los elementos de mayor valor para el negocio.

Introducción

Para garantizar un desarrollo eficiente y ordenado del proyecto, se utilizaron herramientas propias de la metodología Scrum, como las historias de usuario y el Product Backlog. Estas permitieron priorizar y gestionar las tareas del equipo, asegurando que cada Sprint entregará funcionalidades de alto valor para el cliente.

Desarrollo

En el desarrollo de este proyecto, se utilizaron herramientas de planificación y gestión ágiles para garantizar que las necesidades del cliente se tradujeran en funcionalidades claras y priorizadas. Este proceso se estructuró en dos componentes fundamentales: las historias de usuario y el **Product Backlog**.

## 

| Historias de usuario |
| --- |
| Las historias de usuario representan los requerimientos funcionales desde la perspectiva de los usuarios finales. Estas historias permiten al equipo de desarrollo comprender las necesidades específicas del cliente y priorizarlas según el valor que aportan al negocio.  A continuación, se muestran las historias de usuario definidas para el proyecto, organizadas en categorías clave como [ejemplo: Datos Seguros, Usabilidad y Funcionamiento, Usuario Informado]. Cada historia está estructurada con un formato estándar que incluye un rol, una necesidad y un beneficio esperado. |
| DATOS SEGUROS    USUARIO INFORMADO    RENDIMIENTO DEL SISTEMA    USABILIDAD Y FUNCIONAMIENTO    GESTIÓN ADMINISTRATIVA |

| Product Backlog |
| --- |
| El **Product Backlog** es un listado dinámico que contiene todas las tareas y funcionalidades requeridas para el desarrollo del proyecto. Fue creado en conjunto por el Product Owner y el equipo de desarrollo, y se actualiza continuamente a medida que se identifican nuevas necesidades o prioridades.  En este proyecto, el Product Backlog incluye elementos como historias de usuario, tareas técnicas y actividades de mejora. Estos ítems se priorizan según criterios de valor al negocio, viabilidad técnica y urgencia. |
| Desarrollo plantilla Product Backlog |

## 

# Plan de proyecto

| Propósito del plan de proyecto *Propósito, objetivo, visión que se espera de la planificación de este proyecto.* |
| --- |
| El propósito de este proyecto es diseñar e implementar una página web para la venta de repuestos de autos, la cual permita a los clientes acceder a un catálogo de productos, realizar compras en línea y recibir información detallada sobre cada repuesto disponible. La plataforma debe ser fácil de usar, confiable y capaz de gestionar inventarios, pagos, y atención al cliente eficientemente. |

| Alcance del proyecto *Problema, solución propuesta y contexto del proyecto.* |
| --- |
| **Problema:** Actualmente, la PYME de repuestos de autos no cuenta con una plataforma de venta en línea, lo que limita el alcance de sus ventas y dificulta el proceso de compra para sus clientes.  **Solución propuesta:** El proyecto desarrollará una página web con un sistema de gestión de productos, carrito de compras, gestión de pedidos y notificaciones automáticas, además de un espacio de contacto para resolver consultas y quejas.  **Contexto:** El proyecto se llevará a cabo como parte de una estrategia para aumentar la competitividad de la PYME en un mercado de repuestos de autos que cada vez exige más la presencia digital. |
| *Funciones esperadas del software a desarrollar.* |
| * Visualización de un catálogo de productos de repuestos de autos. * Gestión de inventarios. * Sistema de pago integrado. * Envío de notificaciones y actualizaciones de pedidos. * Función de contacto para atención al cliente. |
| *Rendimiento esperado del software a desarrollar.* |
| El software deberá ofrecer un rendimiento óptimo, permitiendo a los usuarios navegar y realizar transacciones en la página web de manera rápida y fluida. Los objetivos de rendimiento son:   1. Tiempo de carga: La página debe cargar en menos de 3 segundos en conexiones estándar. 2. Escalabilidad: El sistema debe ser capaz de soportar al menos 500 usuarios concurrentes sin afectar el rendimiento. 3. Disponibilidad: Mantener una disponibilidad del 99% o superior para minimizar la posibilidad de tiempos de inactividad. 4. Usabilidad: La interfaz debe ser intuitiva y permitir a los usuarios realizar acciones comunes (como buscar repuestos, añadir al carrito, y procesar pagos) con facilidad. |
| **Metodología de Desarrollo**  *Definir y justificar la metodología de desarrollo seleccionada.* |
| Se utilizará una metodología ágil basada en Scrum para desarrollar el proyecto. Esta metodología se adapta bien al tamaño del proyecto y permite realizar ajustes rápidos basados en el feedback del cliente durante el desarrollo de la plataforma, garantizando una solución final que cumpla con las expectativas del propietario. |

| **Definición de roles y responsabilidades** *Roles y responsabilidades de todos los participantes en el desarrollo* *de SW*. *Adjuntar Matriz R.A.C.I.* | |
| --- | --- |
| Rol | Responsabilidades |
| Dueño del Proyecto | Definir los objetivos del proyecto, aprobar funcionalidades clave, gestionar presupuesto y tomar decisiones estratégicas. |
| Ingeniero de Software | Desarrollar la arquitectura del software, seleccionar tecnologías adecuadas y asegurar la calidad técnica del código. |
| Analista | Recopilar y analizar los requisitos del cliente, documentar especificaciones funcionales y técnicas y facilitar la comunicación entre el equipo y el cliente. |
| Diseñador UX/UI | Diseñar la interfaz de usuario, optimizar la experiencia del usuario y realizar pruebas de usabilidad. |
| Desarrollador | Codificar las funcionalidades del sitio web según los requisitos, realizar pruebas unitarias y corregir errores. |
| Encargado de Riesgos | Identificar, analizar y mitigar riesgos del proyecto, incluyendo riesgos técnicos, de tiempo y de presupuesto. |
| QA (Aseguramiento de Calidad) | Planificar y ejecutar pruebas para identificar errores y asegurar que el sitio cumpla con los estándares de calidad. |
| Administrador de Servidor | Configurar y mantener el entorno de servidor, asegurar la estabilidad y la escalabilidad del sistema. |
| Cliente/Usuario | Proporcionar retroalimentación sobre el prototipo y realizar pruebas para asegurar que el sitio cumple con sus expectativas. |

| **Actividad** | **Dueño del Proyecto** | **Ingeniero de Software** | **Desarrollador** | **Analista** | **Diseñador UX/UI** | **Admin. Servidor** | **QA** | **Encargado de Riesgos** | **Cliente/Usuario** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Definir requerimientos | R |  |  | A | C |  |  | I | I |
| Diseño de interfaz de usuario | I |  |  | C | A |  |  |  | I |
| Arquitectura del sistema | I | A | C |  |  |  |  |  |  |
| Desarrollo de funcionalidades | I | C | A |  | C |  |  |  |  |
| Pruebas de funcionalidad | I | C |  |  |  |  | A |  |  |
| Configuración del servidor | I |  | C |  |  | A |  |  |  |
| Identificación y análisis de riesgos | I |  |  |  |  |  |  | A |  |
| Ejecución de pruebas de calidad | I | C |  |  |  |  | A |  |  |
| Revisión y retroalimentación | R |  |  |  |  |  |  |  | A |

| **Estructura de Desglose de trabajo** *Definición de Diagrama y diccionario EDT. Adjuntar diagrama y diccionario* |
| --- |
|  |

|  | **DICCIONARIO EDT** |
| --- | --- |
|  |  |
| Código | Actividades a realizar |
| 1 | Página web de repuestos de autos |
| 1.1 | Autenticación y registro |
| 1.1.1 | Crear formulario de registro: Diseñar una pantalla donde los usuarios puedan ingresar sus datos para crear una cuenta (nombre, correo, contraseña, etc.) |
| 1.1.2 | Validar datos de entrada: Comprobar que los datos ingresados por los usuarios cumplan con el formato requerido (correo válido, contraseñas seguras) |
| 1.1.3 | Implementar creación de usuarios en Firebase: Configurar la base de datos para almacenar los datos de los nuevos usuarios. |
| 1.2 | Carrito de compras |
| 1.2.1 | Implementar lógica para agregar y eliminar productos: Programar las funciones necesarias para que los usuarios puedan añadir o quitar productos del carrito. |
| 1.2.2 | Calcular totales y costos finales: Mostrar el subtotal, impuestos y el monto total del carrito. |
| 1.2.3 | Guardar el carrito en Firebase: Asegurar que la información del carrito se sincronice con la base de datos en tiempo real. |
| 1.3 | Historial de compras |
| 1.3.1 | Diseñar pantalla de historial: Crear una página donde los usuarios puedan ver un registro de todas sus compras pasadas. |
| 1.3.2 | Consultar datos del historial desde Firebase: Implementar la consulta para recuperar las compras desde la base de datos. |
| 1.3.3 | Mostrar detalles de cada compra: Visualizar fecha, productos comprados, cantidad, y monto total. |
| 1.4 | Favoritos |
| 1.4.1 | Diseñar funcionalidad para marcar productos como favoritos: Agregar una opción para que los usuarios guarden productos en una lista de favoritos. |
| 1.4.2 | Crear interfaz para listar favoritos del usuario: Mostrar los productos guardados por el usuario en una sección dedicada. |
| 1.4.3 | Vincular favoritos al usuario en Firebase: Guardar la lista de favoritos asociada a cada usuario en la base de datos. |
| 1.5 | Sistema de pago por transferencia |
| 1.5.1 | Crear la lógica que permita manejar el flujo de pagos por transferencias, asegurando que se registre correctamente el pago y que se ejecute la validación necesaria |
| 1.5.2 | Integrar el sistema de pagos con las plataformas bancarias o de pagos que procesen transferencias |
| 1.6 | Catálogo de productos |
| 1.6.1 | Diseñar interfaz del catálogo: Crear la pantalla principal donde se muestran todos los productos disponibles para la venta. |
| 1.6.2 | Implementar búsqueda y filtros de productos: Programar herramientas para buscar productos por nombre, categoría, precio, etc. |
| 1.6.3 | Mostrar detalles de cada producto: Visualizar información como el nombre, descripción, precio, y estado del producto. |
| 1.6.4 | Conectar el catálogo con Firebase: Asegurar que los datos del catálogo sean dinámicos y actualizados desde la base de datos. |

| **Calendarización de las actividades**  Listado de actividades, tareas, duración, fechas, responsables, etc. Adjuntar Carta Gantt. |
| --- |
| https://app.instagantt.com/r/#projects/oASNbZg6lKyJ2PWBWYvK/oASNbZg6lKyJ2PWBWYvK |

| **Resumen de riesgos** *Listado de riesgos relacionado al desarrollo de S.W. Indicar riesgo, etapa o fase en que se presenta, la probabilidad de que ocurra, magnitud o impacto de este riesgo por etapa en el proceso.*  *Probabilidad: Alta, media, baja.*  *Impacto: Alto, Significativo, Moderado, Inferior y Baja.* | | | | |
| --- | --- | --- | --- | --- |
| **Riesgo** | **Fase** | **Probabilidad** | **Impacto** | **Acción de mitigación** |
| Falta de claridad en las funcionalidades | Planificación | Media | Alto | Organizar reuniones detalladas con el cliente para capturar todos los requerimientos necesarios, especialmente para gestionar el inventario y precios de repuestos. |
| Estructura incorrecta en la base de datos | Diseño del sistema | Media | Alto | Diseñar y revisar una base de datos adecuada para manejar eficientemente los inventarios de repuestos, incluyendo categorías y disponibilidad. |
| Incompatibilidad con dispositivos móviles | Diseño de la interfaz (UI/UX) | Media | Alto | Asegurar que el diseño de la interfaz se adapte a dispositivos móviles para mejorar la experiencia del usuario. |
| Dependencia de la API de pagos externa | Desarrollo del backend | Media | Moderado | Evaluar la confiabilidad de la API de pagos seleccionada y realizar pruebas de integración para asegurar que cumpla con los requisitos del sistema. |
| Errores en la actualización del inventario | Desarrollo del backend | Alta | Significativo | Implementar una sincronización en tiempo real para mostrar correctamente la disponibilidad de productos en el sitio web. |
| Lentitud en el tiempo de carga | Pruebas de rendimiento | Media | Significativo | Optimizar consultas a la base de datos y comprimir archivos para mejorar la velocidad de carga, ya que el catálogo de productos puede ser extenso. |
| Vulnerabilidad en la seguridad de la información | Pruebas y aseguramiento de calidad (QA) | Alta | Alto | Implementar cifrado de datos sensibles y realizar revisiones periódicas de seguridad. |

| **Definición de artefactos**  *Listar y describir los artefactos o entregables que serán administrados y entregados durante el desarrollo del proyecto.* | |
| --- | --- |
| Requisitos Funcionales | Documento que describe en detalle todas las funcionalidades del sistema, especificando lo que el sistema debe hacer desde la perspectiva del usuario. |
| Diagrama de Casos de Uso | Diagramas que muestran cómo interactúan los usuarios con el sistema. Se detallan las acciones que cada tipo de usuario puede realizar (búsqueda de productos, realización de pedidos, etc.). |
| Diagrama de Clases | Diagrama que describe la estructura de clases del sistema, mostrando las entidades principales (usuarios, productos, pedidos) y sus relaciones. |
| Modelo de Datos (BD) | Modelo que describe cómo los datos se organizarán dentro de la base de datos. Incluye tablas para usuarios, productos, pedidos y más. |
| Prototipo de Interfaz de Usuario (UI/UX) | Diseño de la interfaz gráfica del sitio web, que incluye la disposición de las páginas, formularios, y flujos de interacción del usuario. |
| Código Fuente del Sitio Web | El código completo del sitio web, incluyendo el frontend (HTML, CSS, JS) y backend (lógica de negocio, integración de base de datos, etc.). |
| Documentación Técnica | Manual que incluye la explicación del código, arquitectura, dependencias tecnológicas, y cómo configurar y desplegar el sistema en un entorno de producción. |
| Informe de Pruebas | Documento que detalla las pruebas realizadas, como las pruebas funcionales, de usabilidad y de seguridad, y los resultados de estas pruebas. |

| **Condiciones de aceptación para cierre del proyecto**  *Condiciones que se deben cumplir para dar termino al proyecto y margen de tolerancia de aceptación de defectos.* |
| --- |
| Para dar por cerrado el proyecto, debemos asegurarnos de que el sistema cumpla con todos los requisitos y funciones que acordamos, que pase todas las pruebas de calidad (como funcionalidad, seguridad y rendimiento), y que el cliente esté conforme con el diseño y la interfaz del sistema. Además, debemos entregar toda la documentación técnica y para el usuario, y asegurarnos de que el cliente sepa cómo usar el sistema. En cuanto a los defectos, no aceptaremos fallos graves que impidan el funcionamiento del sistema, pero sí podremos tolerar algunos errores pequeños o detalles estéticos, siempre y cuando no afecten el uso del sistema y se corrijan rápidamente después de la entrega. Todo esto debe hacerse dentro de los plazos que tenemos establecidos. |

# Arquitectura de software-Documento DAS

1. Introducción

Este documento presenta la arquitectura de software para el sistema de gestión de repuestos de autos, un proyecto cuyo objetivo es mejorar la eficiencia en el manejo y administración de inventarios, ventas y abastecimiento de repuestos. La implementación de este sistema permitirá gestionar de forma centralizada el inventario, agilizar el proceso de ventas y optimizar la relación con proveedores para garantizar la disponibilidad de productos en todo momento.

El sistema está diseñado para proporcionar a los usuarios una interfaz intuitiva y fácil de usar, que facilite las operaciones diarias y permita el monitoreo en tiempo real del stock, pedidos y ventas. Adicionalmente, la arquitectura del sistema está orientada a la escalabilidad y la adaptabilidad, permitiendo futuras integraciones y mejoras de acuerdo con las necesidades del negocio.

Esta documentación describe las principales decisiones de diseño y estructura los diferentes aspectos de la arquitectura en vistas que capturan tanto la funcionalidad como los elementos técnicos esenciales para su desarrollo.

1. Propósito

El propósito de este documento es presentar la arquitectura de software para el sistema de gestión de repuestos de autos. La arquitectura descrita proporciona una base estructurada que facilita el desarrollo y mantenimiento del sistema, asegurando su escalabilidad y adaptabilidad a las necesidades futuras.

El documento incluye diversas vistas para ilustrar las decisiones arquitectónicas clave y asegurar que se cumplen los requisitos funcionales y no funcionales del sistema.

1. Alcance

Este documento describe el diseño arquitectónico para el sistema de gestión de repuestos de autos, que tiene como objetivo centralizar y optimizar los procesos relacionados con la gestión de productos y la relación con proveedores y clientes. El sistema buscará mejorar la eficiencia de los concesionarios, talleres y proveedores, brindando herramientas para una administración efectiva y en tiempo real de los repuestos automotrices.

El alcance del proyecto incluye:

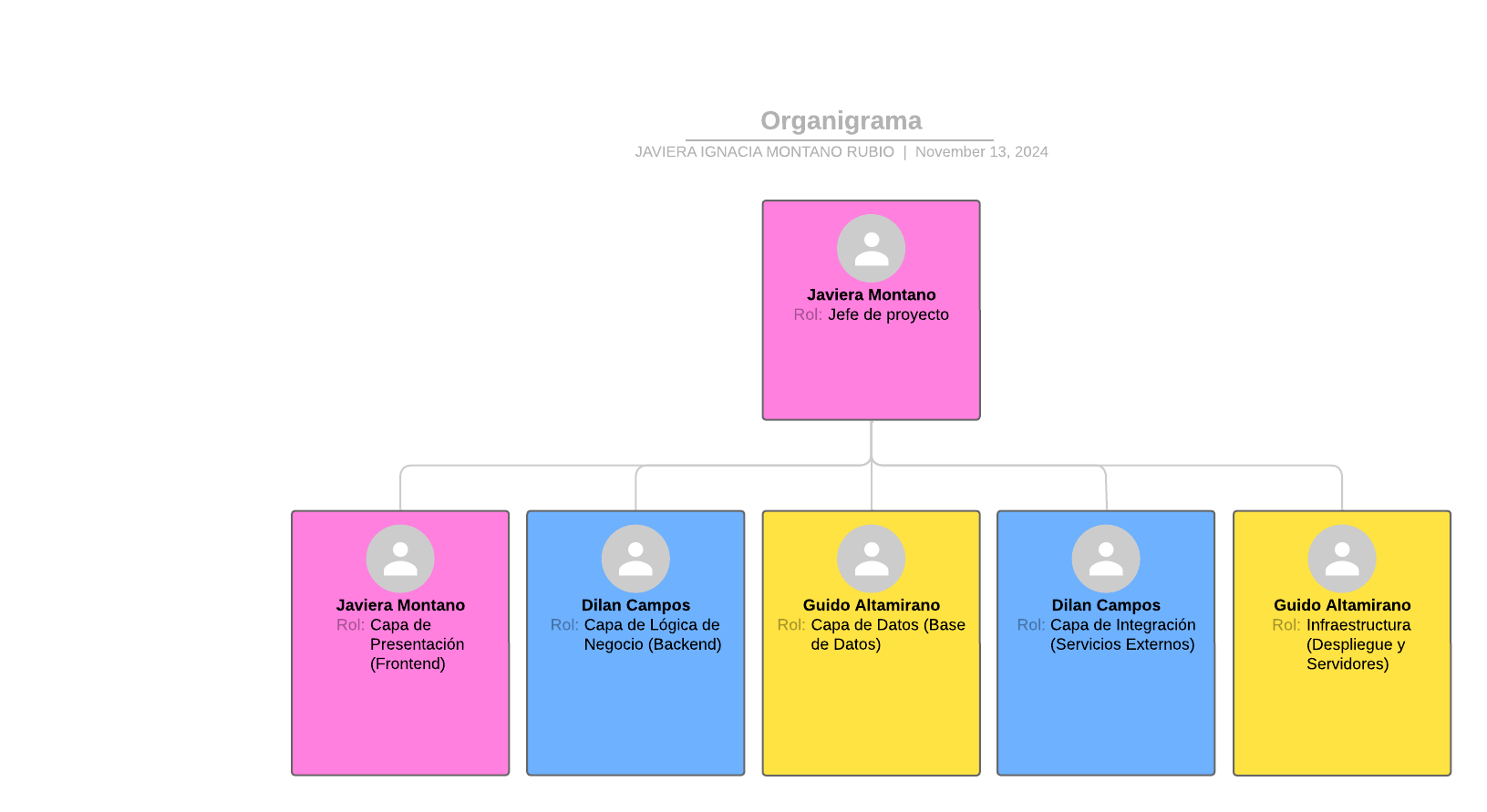
La creación de una plataforma integral que gestione las operaciones de venta, abastecimiento, y control de inventarios de repuestos de autos.

El desarrollo de funcionalidades que permitan la integración de datos entre los actores del negocio (concesionarios, talleres, proveedores), optimizando los flujos de información y las decisiones comerciales.

La implementación de una infraestructura flexible y escalable, que permita la adaptación del sistema conforme las necesidades del negocio crezcan o cambien.

1. Equipo de Trabajo-Actores del Desarrollo

**Organigrama del equipo de proyecto que cubriría el desarrollo del proyecto.**



Selección de puntos de vista de la arquitectura

| VISTAS | UML | Cantidad |
| --- | --- | --- |
| Escenario Principal | Diagrama de Caso de uso | 1 |
| Vista Lógica | Diagrama de Clases  Diagrama de Comunicación  Diagrama de Secuencia | 1  1  1 |
| Vista de Proceso | Diagrama de Actividad | 1 |
| Vista Física | Diagrama de Despliegue | 1 |
| Vista de Despliegue | Diagrama de componentes  Diagrama de paquetes | 1  1 |

## Vistas de la arquitectura

| **UML-VISTA DE ESCENARIO** |
| --- |
| Diagrama de Caso de uso |
|  |

| **UML-VISTA LOGICA** |
| --- |
| [Diagrama de Clases] |
|  |

| UML-VISTA LOGICA |
| --- |
| Diagrama de Comunicación |
| Sistema cliente    Sistema Admin |

| UML-VISTA LOGICA |
| --- |
| Diagrama de Secuencias |
| Diagrama de secuencia para Cliente    Diagrama de secuencia para Admin    Diagrama de secuencia para Api Webpay |

| **UML-VISTA DE PROCESO** |
| --- |
| DIAGRAMA DE ACTIVIDAD |
|  |

| **UML-VISTA FISICA** |
| --- |
| DIAGRAMA DE DESPLIEGUE |
|  |

| **UML-VISTA DE DESPLIEGUE** |
| --- |
| DIAGRAM DE COMPONENTES |
|  |

| UML-VISTA DE DESPLIEGUE |
| --- |
| DIAGRAM DE PAQUETES |
|  |

| **MODELO DE DATOS(BD)** |
| --- |
| Modelo de datos anterior |
| Modelo de datos actualizado tras utilizar Firebase  Productos, adminUID es el que tiene los productos    Pedido y productos del servicio firestore database    Usuarios (Servicio authentication)    Imágenes (servicio Storage)    Servicios de firebase utilizados: |

| **Usabilidad y Calidad de software** |
| --- |
| Vistas principales del sistema |
|  |

DESCRIPCION DE CONECTORES

**5. Arquitectura lógica**:

Desde un dispositivo con navegador Chrome o una aplicación móvil (capa del cliente), el usuario realiza una solicitud al servidor web o directamente a los servicios de Firebase mediante una API (usando HTTP/HTTPS). Firebase (Capa de datos) gestiona las respuestas y permite el acceso a datos en tiempo real, como el inventario de repuestos y el historial de pedidos.

**6. Performances**

Los criterios de rendimiento para los componentes del sistema incluyen alta capacidad de respuesta en la gestión de pedidos y consulta de inventario. La plataforma integra estos componentes en una sola solución para garantizar eficiencia en la gestión de ventas y entregas.

Componentes de Firebase Database:

* **Firebase Realtime Database** o **Firestore**: ofrece almacenamiento en la nube con sincronización en tiempo real, ideal para la gestión de inventario y pedidos en tiempo real.
* **Almacenamiento escalable**: Firebase ajusta automáticamente el espacio y los recursos según la demanda, sin necesidad de configurar núcleos o memoria.
* **Optimización de rendimiento**: diseñada para manejar múltiples solicitudes y grandes volúmenes de datos de inventario de manera eficiente.

Componentes del servicio web:

* **Instancia:** l3v2
* **Núcleos:** 8
* **RAM:** 32 GB
* **Almacenamiento:** 1 TB

Este servicio web optimiza el acceso rápido al sistema, proporcionando un rendimiento adecuado para consultas y transacciones de pedidos.

**7. Detalles de la implementación**

El sistema se implementará en una aplicación web que utiliza Firebase como backend para el almacenamiento de datos y autenticación de usuarios. Firebase ofrece una infraestructura en la nube con almacenamiento seguro y sincronización en tiempo real, ideal para gestionar el inventario y los pedidos de repuestos de manera accesible y escalable.

* **Firebase Realtime Database o Firestore** para almacenamiento en tiempo real
* **Firebase Authentication** para gestión segura de usuarios
* **Hosting de Firebase** para desplegar la aplicación web y ofrecer alta disponibilidad y rendimiento.

**8. Criterios de calidad**

1. **Pruebas unitarias** para verificar el correcto funcionamiento de los diferentes componentes del sistema de gestión de inventario y pedidos.
2. **Pruebas end-to-end (E2E)** para garantizar que el sistema de gestión de pedidos funcione correctamente, desde el pedido hasta la actualización del inventario en Firebase.

**9. Lenguajes y plataformas**

* Firebase: Gestionar la base de datos del sistema de inventario y los pedidos de repuestos sin necesidad de configurar un servidor complejo
* HTML: Lenguaje de etiquetas, usado para crear páginas web.
* CSS: Lenguaje usado para dar estructura a la página WEB.
* JavaScript: Lenguaje para programar en la Página Web.
* Node.js: Backend eficiente para gestionar solicitudes y APIs en tiempo real.
* Angular: Framework frontend para una interfaz de usuario dinámica y organizada.
* Ionic: Framework para crear una app multiplataforma o páginas Web.

### 

# Solicitud de cambio

Plataforma de repuestos de autos

***[001]***

***Fecha: 20/11/2024***

**Datos de la solicitud de cambio**

| Nro control de solicitud de cambio | 001 |
| --- | --- |
| Solicitante del cambio | Cliente |
| Área del solicitante | Gerencia |
| Lugar | Sala de reuniones |
| Patrocinador del proyecto | Guido Altamirano |
| Gerente del proyecto | Javiera Montano |

**Categoría de cambio**

| * Cronograma * Costos * Calidad * Documentación * Otro |
| --- |

Marcar todas las que apliquen:**Causa / origen del cambio**

| * Solicitud de cliente * Otros |
| --- |

**Descripción de la propuesta de cambio**

| El cambio propuesto es reemplazar la integración de la API Webpay por un sistema de pagos mediante transferencias bancarias debido a complicaciones técnicas y de compatibilidad. Este ajuste implica la modificación de la forma en que los pagos se procesan en el sistema. |
| --- |

**Justificación de la propuesta de cambio**

| Se han presentado complicaciones técnicas con la integración de la API Webpay, lo que afectó el cumplimiento de los plazos y la funcionalidad esperada. La opción de utilizar transferencias bancarias como medio de pago proporciona mayor flexibilidad y evita problemas técnicos relacionados con la integración, asegurando una experiencia de pago más confiable. |
| --- |

**Impacto del cambio en la línea base**

| **Alcance**: El cambio introduce un nuevo componente funcional para el procesamiento de pagos mediante transferencias bancarias. Esto implicará tareas adicionales de desarrollo y pruebas para integrar este sistema de pagos de manera eficiente, lo que podría generar modificaciones en los requisitos funcionales del sistema.  **Cronograma**:  **Calidad:** La calidad del proyecto se mantendrá alta, ya que el cambio busca una solución más robusta y confiable, mejorando la experiencia del usuario en términos de facilidad de pago y seguridad. |
| --- |

**Implicaciones de recursos (materiales y capital humano)**

| **Recursos humanos:** Se necesitarán recursos adicionales para implementar, probar y ajustar el sistema de pagos mediante transferencias bancarias, incluyendo desarrolladores, testers y personal de soporte técnico.  **Recursos materiales:** Serán necesarias licencias y suscripciones para las plataformas de pago de transferencias, así como el uso de infraestructura en la nube para el procesamiento seguro de las transacciones. |
| --- |

**Implicaciones para los interesados**

| **Clientes:** El sistema de pagos por transferencia ofrecerá una alternativa más accesible y confiable para los clientes que prefieren no utilizar tarjetas de crédito.  **Equipo de desarrollo:** El equipo de desarrollo tendrá que ajustar los plazos y las tareas para asegurar que la nueva integración de pagos sea fluida, sin comprometer otras funcionalidades del sitio web. |
| --- |

**Implicaciones en la documentación del proyecto**

| * **Alcance**: El alcance se actualizará para reflejar la integración del sistema de pagos por transferencia bancaria y los ajustes asociados. * **EDT (Estructura de Desglose del Trabajo)**: El EDT se actualizará para agregar las tareas relacionadas con el desarrollo e integración del nuevo sistema de pagos. * **Cronograma**: El cronograma se ajustará para incorporar las tareas adicionales de desarrollo e integración del sistema de pagos, extendiendo el tiempo de entrega en 4 semanas. * **Requisitos**: Los requisitos funcionales y no funcionales se actualizarán para incluir el sistema de pagos por transferencia bancaria y garantizar su integración adecuada con el resto del sistema. |
| --- |

**Riesgos**

| **Riesgo de integración:** La integración del sistema de pagos por transferencia bancaria podría generar problemas técnicos con otros sistemas del proyecto, lo que requeriría tiempo adicional para garantizar su compatibilidad.  **Riesgo de desempeño:** El sistema de pagos por transferencia bancaria debe ser probado exhaustivamente para asegurar que maneje un volumen adecuado de transacciones sin afectar el rendimiento del sitio web.  **Riesgo de cambios adicionales:** El ajuste de los requisitos y la integración de la nueva solución puede generar cambios adicionales en el proyecto, lo que podría resultar en más tiempo y costos. |
| --- |

**Aprobación**

| Aprobado por Guido Altamirano  Aprobado por Javiera Montano |
| --- |

**Firmas del comité de cambios**

| **Nombre** | **Rol / Cargo** | **Firma** |
| --- | --- | --- |
| **Guido altamirano** | **Dueño del sistema** | **xxx** |
| **Javiera Montano** | **Jefe del proyecto** | **xxx** |