

Documentación del Proyecto Llantas DC

Empresa: Llantas DC

Fecha: 27/02/2025

LLANTAS DC

1. Identificación de la problemática y necesidades

1.1. Descripción del Problema

Llantas DC es una empresa dedicada a la venta de llantas para vehículos de carga, ubicada en la localidad de Fontibón, en Bogotá. Actualmente, la empresa realiza ventas tanto en su punto físico como a través de redes sociales. Sin embargo, no cuenta con un sistema de aplicación móvil que optimice sus procesos de gestión de pedidos, clientes e inventario, lo que puede generar demoras, errores y dificultades en la atención al cliente.

1.2 Necesidades del Cliente

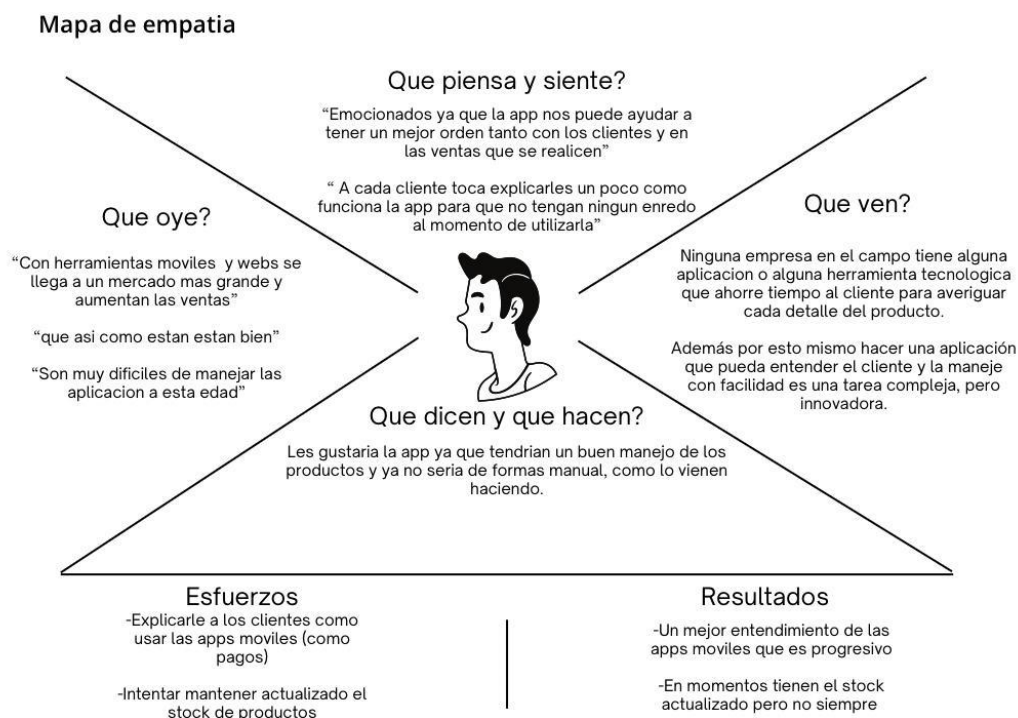


Diagrama 1. Mapa de empatía

- Optimizar la gestión de ventas y pedidos.
- Mejorar la organización del inventario.
- Facilitar la comunicación con los clientes.
- Centralizar la información en una sola plataforma.
- Implementar un sistema de notificaciones automáticas para el seguimiento de pedidos.
- Permitir la generación de reportes de ventas y stock en tiempo real.
- Integrar un método de pago seguro y eficiente dentro de la aplicación.
- Ofrecer un sistema de fidelización para clientes recurrentes.

1.3 Criterios de Evaluación

- Facilidad de uso de la aplicación.
- Reducción en tiempos de respuesta a pedidos.
- Integración con redes sociales y sistemas de pago.
- Seguridad en el manejo de datos de clientes y transacciones.
- Capacidad de la aplicación para generar reportes automatizados de ventas y stock.
- Nivel de satisfacción del usuario final en cuanto a navegación y experiencia.
- Eficiencia en la actualización de datos en tiempo real.
- Posibilidad de escalabilidad y adaptabilidad a futuras mejoras y expansiones.

2. Definición de los objetivos del proyecto

Análisis: Hacer un estudio de mercado para poder identificar las necesidades que tiene el cliente y las funcionalidades clave (Workshop) para luego definir cuáles son los requerimientos funcionales y no funcionales del sistema

Diseño: Crear los respectivos prototipos de interfaces para el registro de usuarios y gestión de productos. (módulos)

Desarrollo:

Usuarios: Poder implementar un registro y login seguro (ej: autenticación en dos pasos), de igual manera poder desarrollar perfiles personalizados con historial de compras interactivo. De acuerdo con lo anterior, lograr integrar un sistema de fidelización con promociones según el cliente.

Productos: Poder construir satisfactoriamente un sistema de gestión de stock en tiempo real y que este se relacione con la implementación de una búsqueda filtrada (por marca, precio, dimensiones). Finalmente, generar reportes automatizados de demanda y stock.

Pruebas:

Realizar pruebas unitarias, de integración y de sistema.

Asegurar que el inventario refleje cambios en tiempo real sin falsos positivos.

Testear la experiencia de usuario (rendimiento, usabilidad, compatibilidad).

3. Alcance del proyecto

Objetivo General

Desarrollar un Sistema de Gestión Integral para Llantas DC enfocado en dos módulos principales: Usuarios y Productos, con el fin de optimizar procesos comerciales, control de inventario y experiencia del cliente.

Duración estimada: 4 meses

Módulos

Usuario: Busca gestionar los perfiles creados, autenticación y roles para administradores, vendedores y clientes.

El alcance del módulo es obtener un registro y autenticación segura (JWT) por parte de la aplicación a los usuarios, que logre también el desarrollo de perfiles personalizados en el que se pueda observar un historial de compras, datos de contacto, etc. Mas allá, la aplicación logra la identificación de roles y permisos tales como los Administradores quienes tendrían acceso total a reportes y configuración, los vendedores cuyo rol se centra en la gestión de pedidos y consulta de inventario y finalmente los clientes que tienen el permiso de autogestión de pedidos y chat de soporte.

Añadir también de la mano con el historial, el desarrollo de un sistema de fidelización que mediante compras recurrentes se generen descuentos automatizados.

Tecnologías:

Backend: Node.js + Express.js.

Base de datos: MongoDB (colecciones: usuarios, roles).

Autenticación: JWT.

Equipo requerido:

1 desarrollador (Node.js, JWT).

1 diseñador UX/UI

Tiempo estimado: 2 meses.

Productos: Busca controlar inventario en tiempo real y facilitar la gestión de ventas.

El alcance del módulo se centra en la gestión de productos CRUD de llantas (marca, precio, dimensiones, stock, etc) para así tener una búsqueda filtrada eficiente y sencilla, que va de la mano con un inventario automático que brinda alertas de stock bajo, stockouts, etc.

De la mano viene el desarrollo de reportes de ya sea ventas diarias, semanales o mensuales, lograr identificar las tendencias de la demanda mediante gráficos sencillos como de barras o línea.

Tecnologías:

Backend: Node.js + Express.js.

Base de datos: MongoDB (colecciones: productos, inventario).

Reportes: PDFKit o ExcelJS.

Equipo requerido:

1 desarrollador (Node.js + MongoDB).

1 analista de datos

Tiempo estimado: 2 meses.

3.1.1.1. Módulos a desarrollar (uno por estudiante)

El modulo que realizará Mafe va a ser el de usuarios, el de nicolas va a ser el de ventas y el de Camilo va a ser el de producto.

Cada estudiante trabajará en su módulo de forma independiente, pero asegurando la integración con los demás componentes mediante API REST en el backend.

Diagramas de flujo conceptuales de cada alternativa.

Solución de Nico:

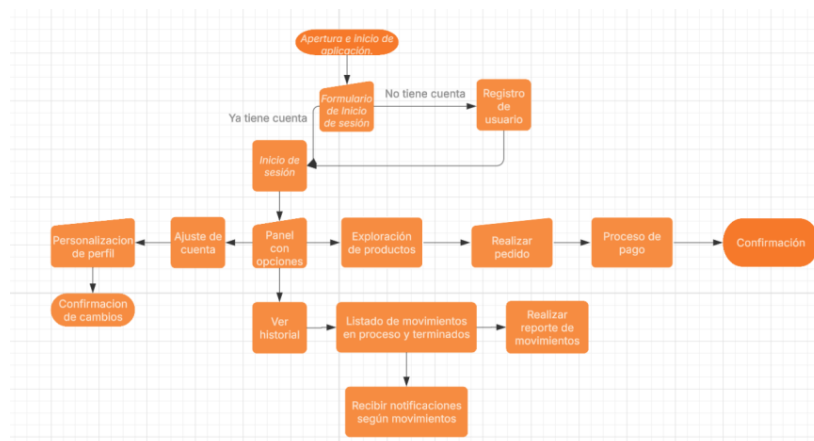


Diagrama 2.1. Diagrama de solución Nicolás

Solución de Mafe:

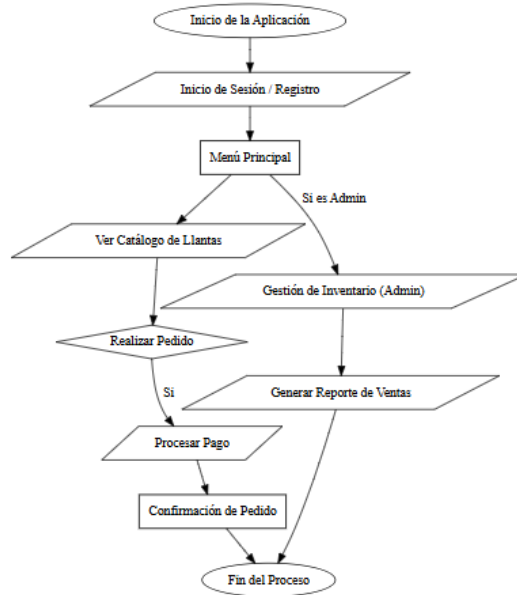


Diagrama 2.2 Diagrama de solución Mafe

Solución Camilo:

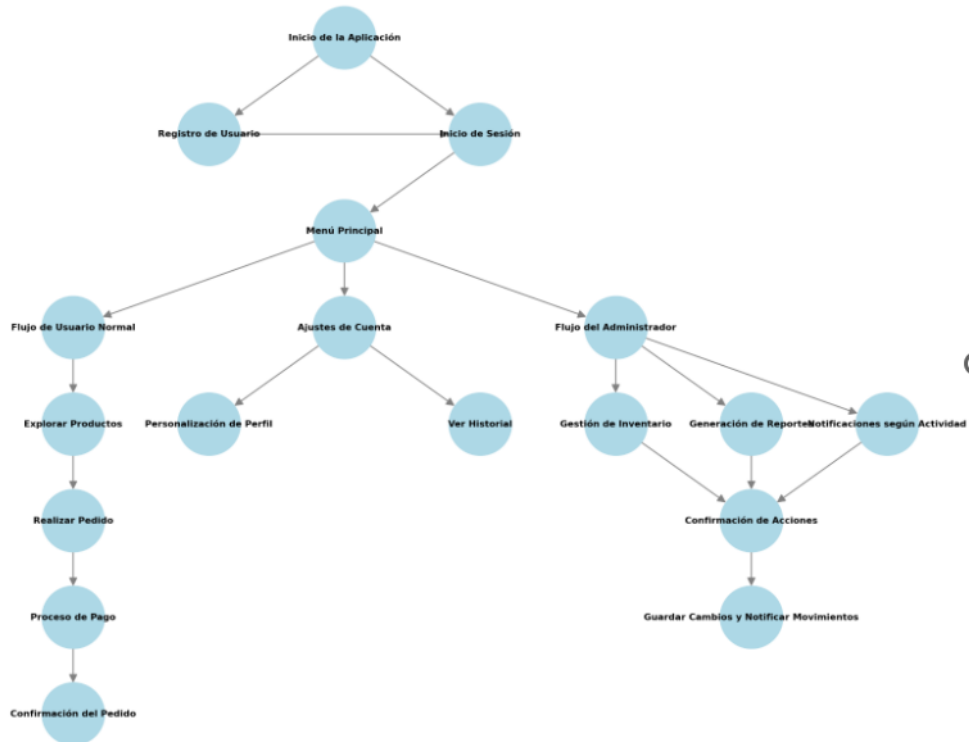


Diagrama 2.3 Diagrama de solución Camilo

Diagrama de la solución seleccionada:

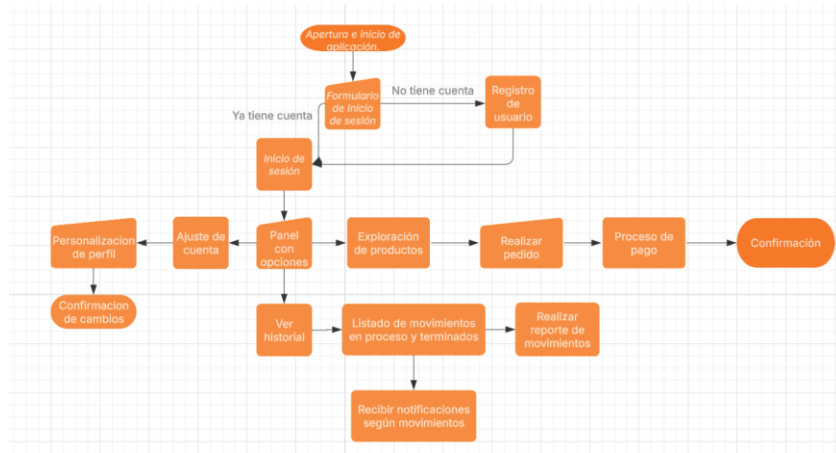


Diagrama 2.1. Diagrama de solución Nicolás

4. Diagrama de Flujo del Módulo de Productos

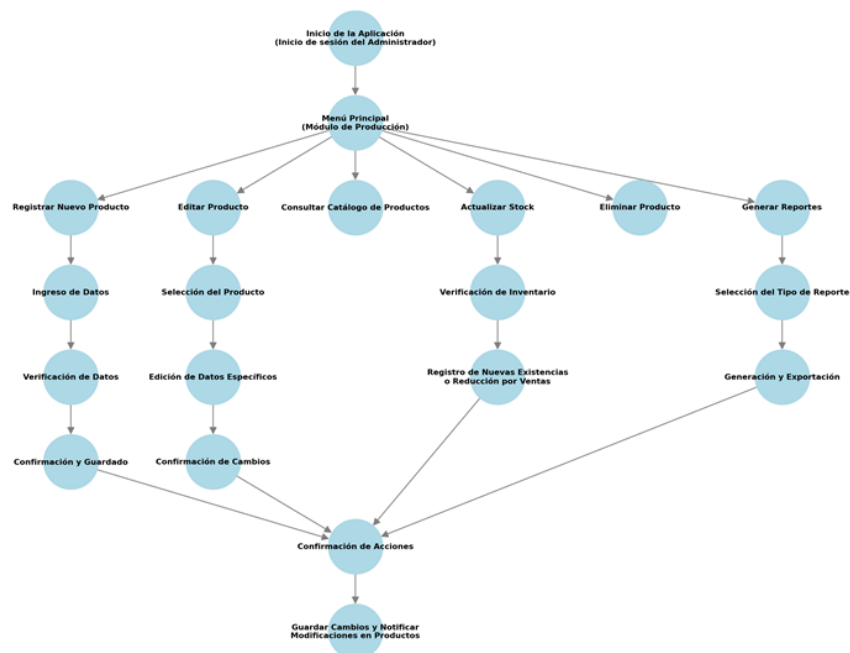


Diagrama 3.1 Diagrama de flujo módulo Productos

Flujo modulo de usuario:

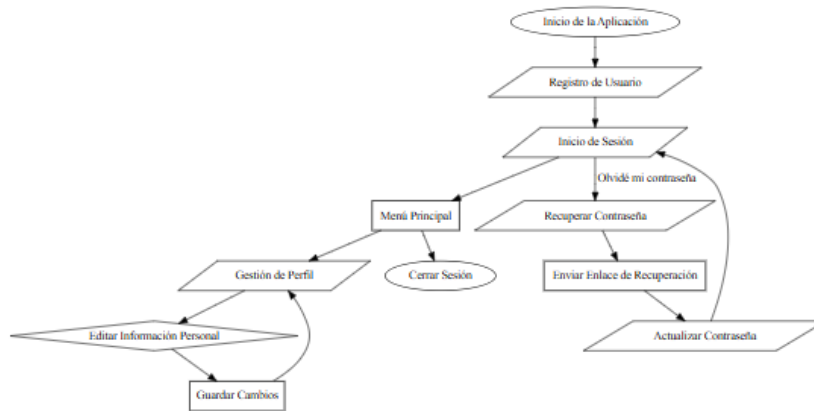


Diagrama 3.2 Diagrama de flujo módulo Usuario

5. Metodología de desarrollo

MODULO USUARIO:

5.1. Tareas del Proyecto

Las tareas se organizan en tres fases principales: Desarrollo de autenticación, gestión de perfil y administración del historial de compras.

- Fase 1: Autenticación de Usuarios

Implementar el sistema de inicio de sesión y registro.

Integración con Google y Facebook para autenticación rápida.

Implementación de recuperación de contraseña.

Verificación de email y autenticación en dos pasos (2FA).

- Fase 2: Gestión de Perfil

Diseñar y desarrollar la interfaz del perfil de usuario.

Implementar la edición de datos personales (nombre, teléfono, dirección).

Opciones de privacidad y configuración de notificaciones.

Guardar métodos de pago para futuras compras.

- Fase 3: Historial de Compras

Visualización del historial de compras.

Detalles de cada pedido realizado, incluyendo estado de envío.

Implementar opción de repetir pedido desde el historial.

Permitir a los usuarios solicitar soporte para pedidos anteriores.

MODULO VENTA:

5.2. Tareas del Proyecto

- Fase 1: Gestión de Productos e Inventario

Diseñar la base de datos de productos

Desarrollar la funcionalidad de búsqueda de productos filtrada

Implementar stock en tiempo real

Diseñar interfaz para administradores (agregar, editar y eliminar productos.)

Generar reportes de inventario en tiempo real.

- Fase 2: Gestión de Pedidos y Carrito de Compras

Implementar la funcionalidad de agregar productos al carrito.

Desarrollar la vista del carrito de compras con edición y eliminación de productos.

Calcular precios

Generar pedido tras confirmacion

Actualizar el stock tras generar pedido.

Notificar al usuario sobre la confirmación del pedido.

- Fase 3: Procesamiento de Pagos y Facturación

Integrar métodos de pago.

Implementar validación de pago y respuesta en tiempo real (aceptado o rechazado).

Configurar el historial de transacciones del usuario.

- Fase 4: Notificaciones

Implementar notificaciones en tiempo real.

Desarrollar una interfaz de seguimiento de pedidos (en proceso, enviado, entregado).

Configurar recordatorios automáticos de estado del pedido.

Desarrollar la opción de activar/desactivar notificaciones.

- Fase 5: Reportes

Generar reportes de ventas en tiempo real.

Desarrollar herramientas para la gestión de clientes (historial, contacto).

Configurar permisos de acceso para administradores.

6. Cronograma con los tiempos de cada paso metodológico

A continuación, se presenta un cronograma estimado para el desarrollo del proyecto, con una duración de 3 meses.

Fase	Tarea	Duración
Semana 1 - 2	Análisis y definición de requisitos funcionales (RQF) y no funcionales (RQNF)	6 días
	Investigación de UX/UI y benchmarking de aplicaciones similares	4 días
	Definición del flujo de usuario y arquitectura de información	4 días
Semana 3 - 4	Creación de wireframes de baja fidelidad (bocetos iniciales)	5 días
	Validación y ajustes iniciales con stakeholders	3 días
	Definición de tipografías, colores y estilos visuales	4 días
Semana 5 - 6	Diseño de alta fidelidad de pantallas clave (UI)	7 días
	Diseño de iconografía, botones y elementos gráficos	3 días
	Presentación de avances y ajustes según feedback	4 días
Semana 7 - 8	Creación de prototipo interactivo en herramienta como Figma o Adobe XD	6 días
	Pruebas de usabilidad con usuarios potenciales	4 días
	Iteraciones y mejoras con base en retroalimentación	4 días
Semana 9 - 10	Documentación del diseño del módulo de usuario (guía de estilo y especificaciones de UI)	6 días
	Creación de guía de navegación para programadores	4 días
Semana 11 - 12	Ajustes finales en el diseño	4 días
	Preparación y presentación final del diseño	4 días
	Entrega del prototipo final y documentación	4 días

Tabla 1. Cronograma

7. Requisitos del software

7.1. MODULO USUARIO:

CÓDIGO	requisitos funcionales
RQF001	Nombre: Registrar usuarios
	Descripción: El sistema debe permitir a los nuevos usuarios registrarse utilizando su correo electrónico y una contraseña segura.
	Usuarios: Administrador

CÓDIGO	requisitos funcionales
RQF002	Nombre: Iniciar sesión
	Descripción: Los usuarios deben poder iniciar sesión con su correo electrónico y contraseña registrados.
	Usuarios: Administrador, Usuario
CÓDIGO	requisitos funcionales
RQF003	Nombre: Autenticar con redes sociales
	Descripción: El sistema debe permitir el inicio de sesión con cuentas de Google y Facebook.
	Usuarios: Administrador, Usuario
CÓDIGO	requisitos funcionales
RQF004	Nombre: Recuperar contraseña
	Descripción: El usuario debe poder restablecer su contraseña a través de un enlace enviado por correo electrónico.
	Usuarios: Administrador, Usuario
CÓDIGO	requisitos funcionales
RQF005	Nombre: Verificar correo electrónico
	Descripción: Al registrarse, el usuario debe recibir un código de verificación en su correo para confirmar su cuenta.
	Usuarios: Administrador, Usuario
CÓDIGO	requisitos funcionales
RQF006	Nombre: Editar información personal
	Descripción: El usuario debe poder modificar su nombre, dirección y número de teléfono desde su perfil.
	Usuarios: Administrador, Usuario

CÓDIGO	requisitos funcionales
RQF007	Nombre: Configurar notificaciones
	Descripción: El usuario debe poder activar o desactivar notificaciones de promociones y actualizaciones de pedidos.
	Usuarios: Administrador, Usuario
CÓDIGO	requisitos funcionales
RQF008	Nombre: Eliminar cuenta
	Descripción: El usuario debe poder solicitar la eliminación de su cuenta y datos personales.
	Usuarios: Administrador
CÓDIGO	requisitos funcionales
RQF009	Nombre: Visualizar historial de pedidos
	Descripción: El usuario debe poder ver una lista de todas sus compras anteriores con detalles como fecha, monto y estado del pedido.
	Usuarios: Administrador, Usuario
CÓDIGO	requisitos funcionales
RQF010	Nombre: Consultar detalle de un pedido
	Descripción: El usuario debe poder ver la información detallada de cada pedido realizado, incluyendo productos comprados, estado de envío y número de guía.
	Usuarios: Administrador, Usuario
CÓDIGO	requisitos funcionales
RQF011	Nombre: Repetir pedido
	Descripción: El usuario debe poder realizar nuevamente un pedido anterior con un solo clic.
	Usuarios: Administrador, Usuario
CÓDIGO	requisitos funcionales
RQF012	Nombre: Cancelar pedido
	Descripción: El usuario debe poder cancelar un pedido si aún no ha sido procesado por la tienda.
	Usuarios: Administrador, Usuario
CÓDIGO	requisitos funcionales
RQF014	Nombre: Chatear con atención al cliente
	Descripción: El usuario debe poder comunicarse con el servicio de atención al cliente en tiempo real.
	Usuarios: cliente

7.1.1.MODULO VENTA:

CÓDIGO	requisitos funcionales
RQF001	Nombre: Consultar productos
	Descripción: El sistema debe permitir a los usuarios buscar productos por nombre, marca, tipo y precio.
	Usuarios: Administrador, Usuario

CÓDIGO	requisitos funcionales
RQF002	Nombre: Verificar disponibilidad de productos en tiempo real
	Descripción: La aplicación debe mostrar la disponibilidad de los productos en tiempo real al momento de la búsqueda.
	Usuarios: Administrador, Usuario

CÓDIGO	requisitos funcionales
RQF003	Nombre: Agregar y eliminar productos del carrito
	Descripción: El sistema debe permitir a los usuarios agregar y eliminar productos del carrito de compras antes de confirmar el pedido.
	Usuarios: Administrador, Usuario

CÓDIGO	requisitos funcionales
RQF004	Nombre: Verificar métodos de pago disponibles
	Descripción: El sistema debe permitir a los usuarios realizar pagos mediante tarjeta, transferencia y efectivo.
	Usuarios: Administrador, Usuario

CÓDIGO	requisitos funcionales
RQF005	Nombre: Calcular automáticamente del total del pedido
	Descripción: La aplicación debe calcular automáticamente el subtotal, impuestos y total del pedido a medida que se agregan o eliminan productos.
	Usuarios: Administrador, Usuario

RQNF001 - Tiempo de respuesta en búsquedas

Descripción: La búsqueda de productos debe ejecutarse en menos de 5 segundos para garantizar una experiencia de usuario fluida.

Usuarios: Clientes.

RQNF002 - Procesamiento de transacciones

Descripción: Las transacciones de pago deben procesarse en menos de 10 segundos, asegurando rapidez en la confirmación de pedidos.

Usuarios: Clientes.

RQNF003 - Seguridad en el almacenamiento de datos

Descripción: Todos los datos sensibles (usuarios, pedidos, productos) deben almacenarse garantizando la protección de la información.

Usuarios: Administradores, Clientes.

RQNF004 - Tiempo de envío de notificaciones

Descripción: Las notificaciones sobre el estado de los pedidos deben enviarse en menos de 5 segundos tras un cambio de estado.

Usuarios: Clientes.

7.1.2.MODULO PRODUCTOS:

CÓDIGO	requisitos funcionales
RQF001	Nombre: Registrar un nuevo producto
	Descripción: Permitir el registro de productos con datos como referencia, marca, dimensión, diseño, precio y stock disponible. Ejemplo: Se registra un producto "Xforza P1" con dimensión "235/75R17.5", diseño "tracción", marca "XBRI", precio 1.140.000, y stock inicial de 124 unidades.
	Usuarios: Administrador, Usuario

CÓDIGO	requisitos funcionales
RQF002	Nombre: Editar un producto
	Descripción: Permitir la edición de los datos de un producto ya registrado.
	Usuarios: Administrador, Usuario

CÓDIGO	requisitos funcionales
RQF003	Nombre: Consultar productos
	Descripción: Visualizar el catálogo de productos con filtros por marca, tamaño y disponibilidad.

	Usuarios: Administrador, Usuario
--	---

CÓDIGO	requisitos funcionales
RQF004	Nombre: Actualizar stock
	Descripción: Registrar cambios en el stock por nuevas compras o ventas realizadas.
	Usuarios: Administrador, Usuario

CÓDIGO	requisitos funcionales
RQF005	Nombre: Eliminar un producto
	Descripción: Eliminar productos obsoletos o discontinuados.
	Usuarios: Administrador, Usuario

CÓDIGO	requisitos funcionales
RQF006	Nombre: Generar reportes
	Descripción: Generar reportes sobre el inventario y los productos más vendidos.
	Usuarios: Administrador, Usuario

8. Stakeholders del Proyecto



Diagrama 4. Mapa de Stakeholders

9. Patrones de diseño

Existen varios patrones de diseño que pueden aplicarse al proyecto Llantas DC como un todo y también se distinguen otros mejor encaminados a los módulos de usuario y de producto, para el proyecto en general de la aplicación los patrones que se tienen en cuenta son “Factory method”, “Observer” y “Strategy”. El primer patrón “Factory method” es útil para que la aplicación pueda ser usada por distintos tipos de usuarios tales como administrador, cliente y vendedor, ya que es una misma plataforma, pero con permisos distintos y accesos diferentes según el tipo de persona dentro de la app. El patrón de “Observer” es aplicable para mantener notificado al usuario, por ejemplo, el cambio de un SKU en stock, número de existencias, etc. Incluso puede ser usado para la promoción de ofertas o demás información pertinente. Finalmente, el patrón de “Strategy” puede ser usado, pero en un paso más avanzado de la aplicación debido a que este suele ser más útil cuando se realizan los pagos dentro de la aplicación, ayuda a incluir los diferentes medios de pago sin que se pierda info valiosa.

Ahora bien, para los módulos que se van a implementar existen algunos patrones aplicables tales como lo son nuevamente “Factory method” y “Strategy”, que en este caso se usarían para crear los distintos roles y enlazarlos con inicios de sesión y métodos de autenticación como correo, o 2fa, etc. Para el módulo de productos se encontraron patrones como lo son “Repository” que se usa para acceder a una base de datos con facilidad y “Decorator” que puede usarse para darle una personalización a la llanta si esta lo requiere y tenemos también

“Specification” que puede aplicarse para la búsqueda dinámica de productos, por ejemplo, filtrado inteligente.

Para añadir mencionamos mejores prácticas para la resolución de problemas durante el desarrollo del diseño de software como lo son MVC, que nos permite tener una organización más nítida del proyecto y permitir que este pueda ser escalable a futuro. Otra practica para tener en cuenta es SRP que, según lo investigado, la separación de responsabilidades independiza cada módulo y obliga que cada módulo o clase realice una sola función. En resumen los patrones a usar son “Observer”, “Factory method” y “Specification”, debido a que estos son los que pueden ayudarnos a desarrollar el proyecto de la mejor manera como está planeado.

10. Estilos arquitectónicos

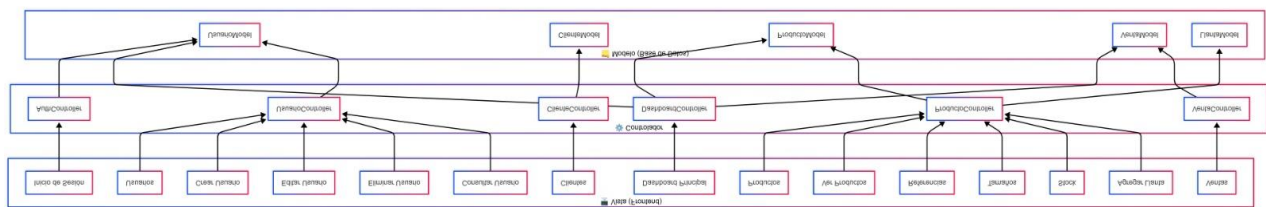


Diagrama 5. Diagrama arquitectónico

11. Diagrama de componentes

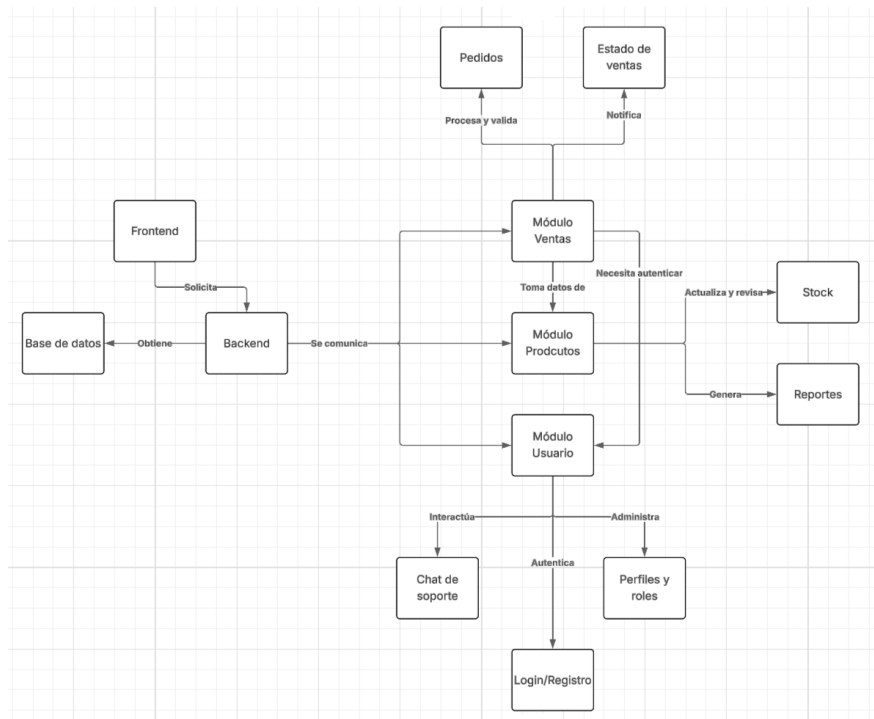


Diagrama 6. Diagrama de componentes

12. Diagrama de clases

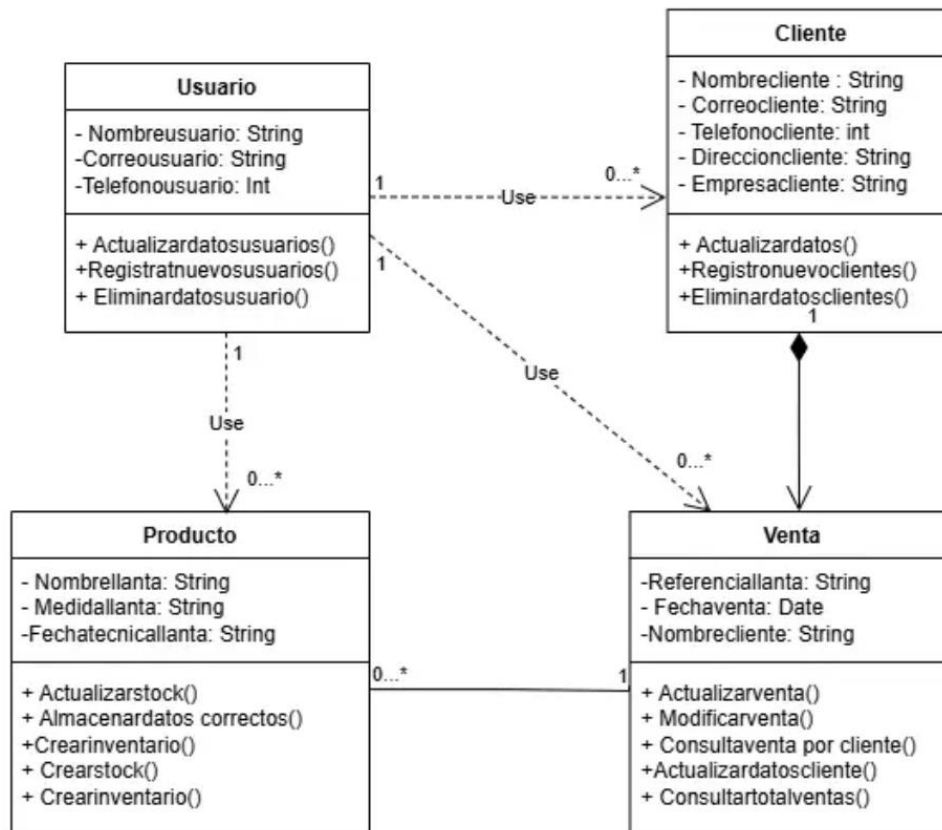


Diagrama 7. Diagrama de clases

Vemos que el sistema gira en torno a cuatro clases principales las cuales son: Usuario que es definido por sus atributos “nombreUsuario” (String), “correoUsuario” (String) y “telefonoUsuario”(Int) y es el eje central del sistema. También está Producto que es caracterizado por “nombreLlanta” (String), “medidaLlanta” (String) y “fechaTecnicaLlanta” (String) y su función es gestionar el inventario de llantas. Como tercera clase tenemos la clase Cliente cuyos atributos son “nombreCliente” (String), “correoCliente” (String), “telefonoCliente” (Int), “direccionCliente” (String) y “empresaCliente” (String) el cual almacena toda la información de los compradores. Finalmente tenemos la clase Venta que registra transacciones mediante “referenciaLlanta” (String), “fechaVenta” (Date) y “nombreCliente” (String).

Las relaciones que se tienen de estas clases son: un Cliente puede tener múltiples Ventas (agregación), mostrando que las ventas pertenecen al cliente pero pueden gestionarse independientemente mientras los Productos se asocian a las Ventas de manera flexible (asociación), permitiendo que un mismo producto participe en múltiples ventas y viceversa. El Usuario interactúa con todas las clases para coordinar el flujo completo del sistema.

13. Diagrama de secuencias

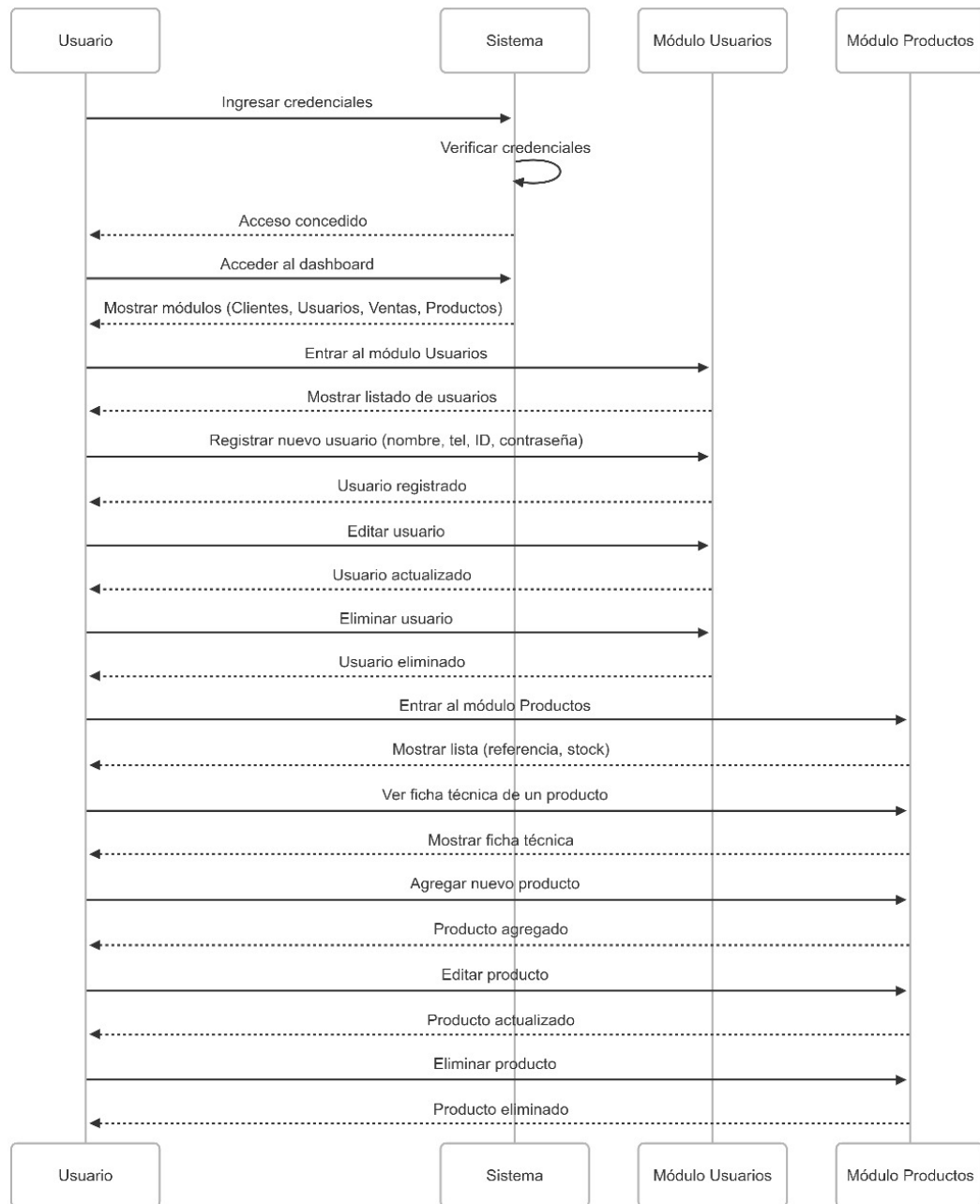


Diagrama 8. Diagrama de secuencias

14. Diagrama de casos de uso

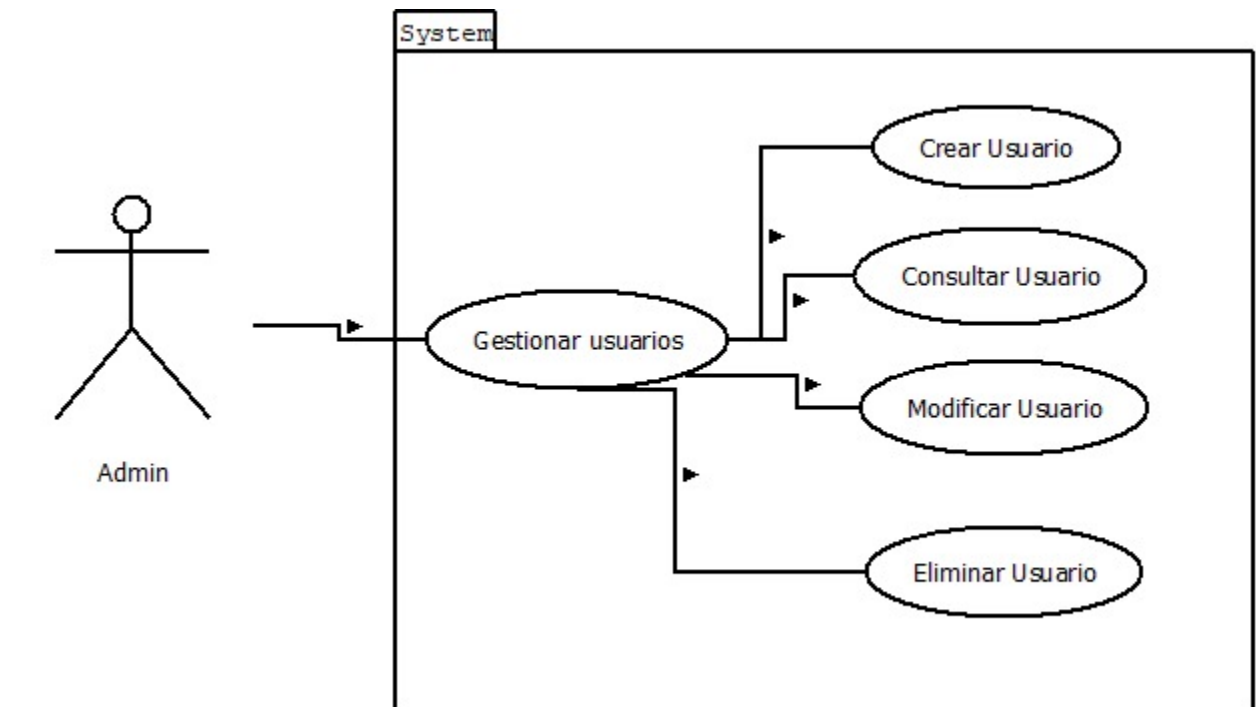


Diagrama 9.1 Diagrama de casos de uso modulo usuario

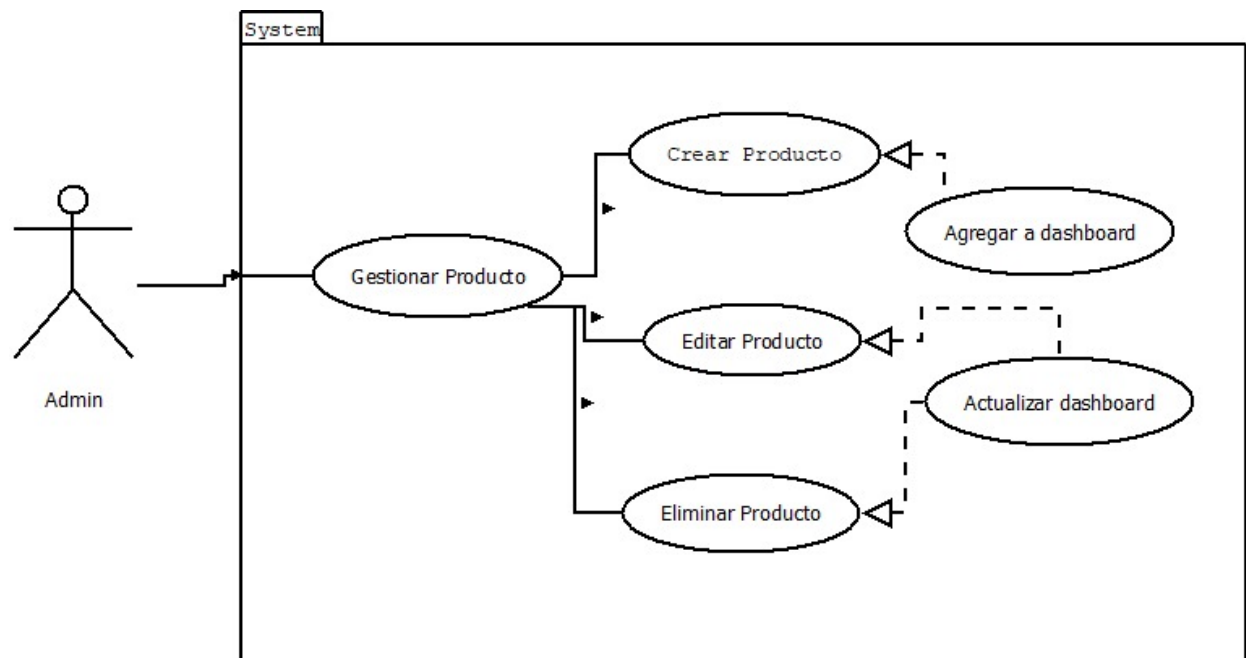


Diagrama 9.2 Diagrama de casos de uso modulo productos

15. Diseño de interfaces y prototipos

Prototipo de baja fidelidad Storyboard:

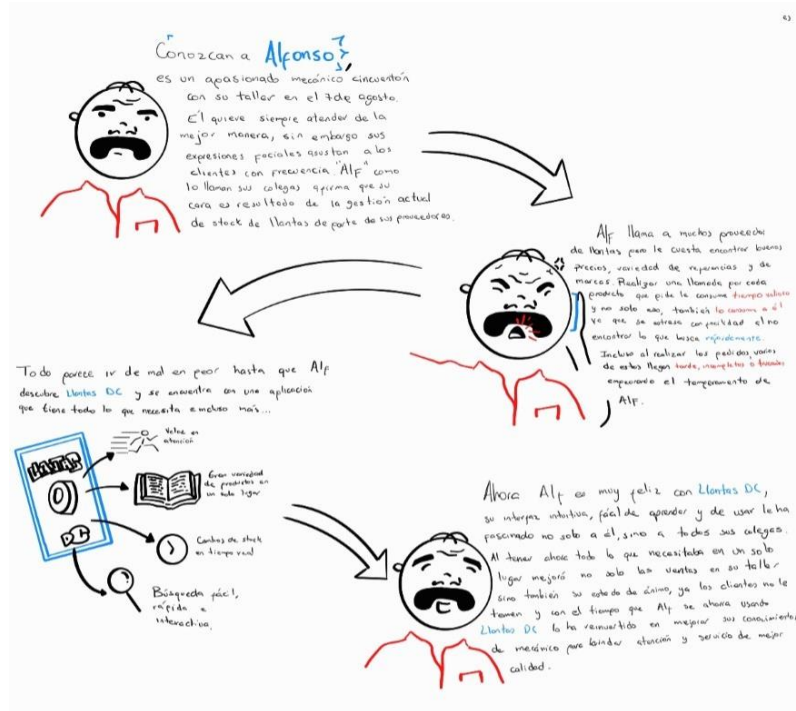


Diagrama 10.1 Storyboard 1

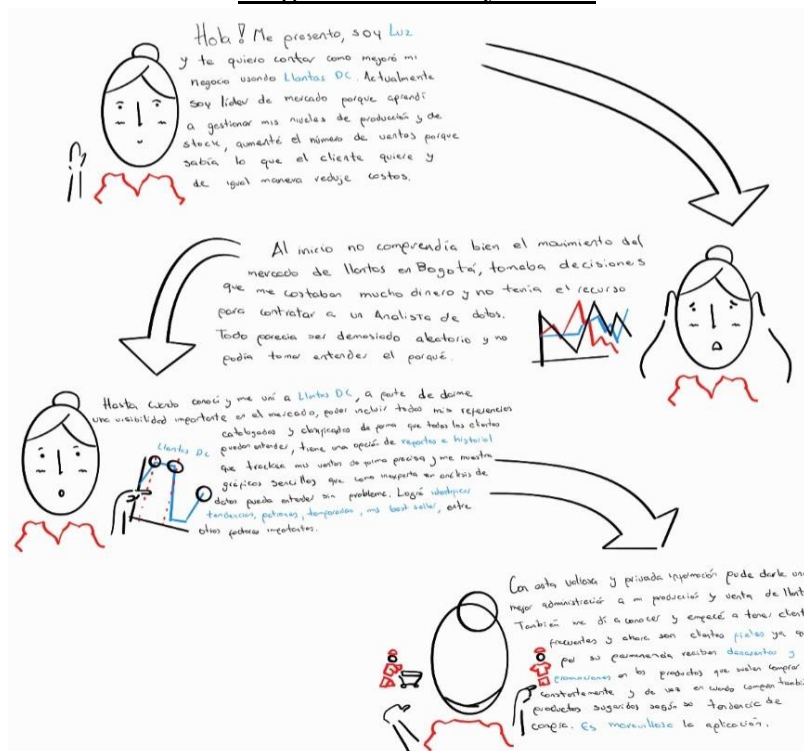


Diagrama 10.2 Storyboard 2

Interfaz de baja fidelidad :

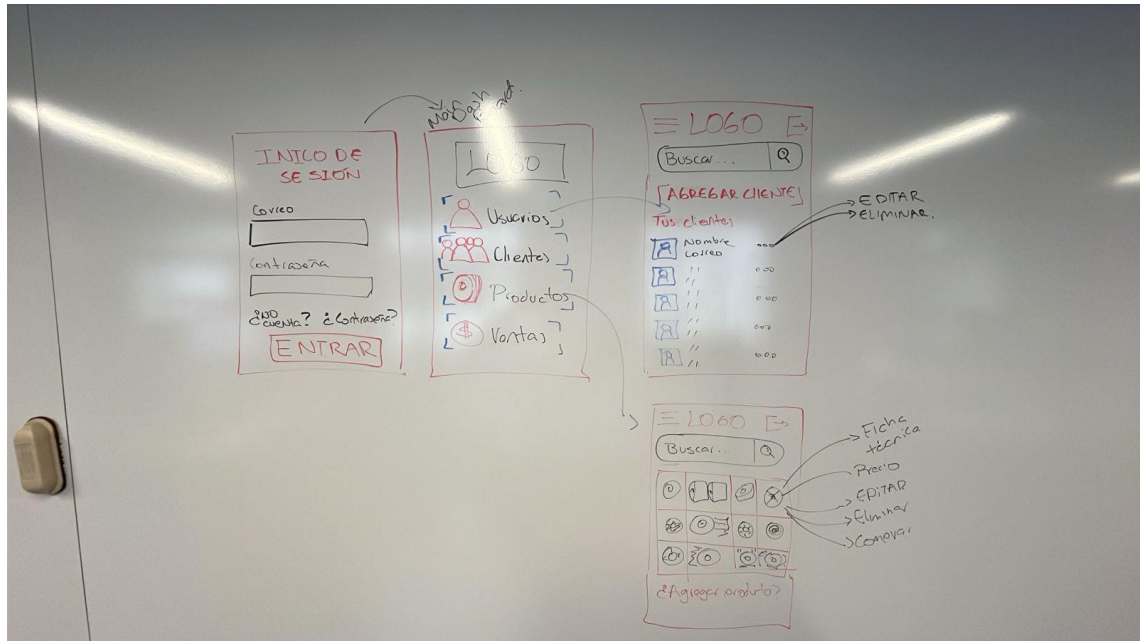


Imagen 1. Prototipo de baja fidelidad

Prototipo e interfaz de alta fidelidad:

<https://www.figma.com/design/7fO28FVE1JIEEqgjkz5BeF/Llantas-dc?node-id=52-250&t=Pw7s4Ctdvb3o2qRS-0>

16. Repositorio del Proyecto:

<https://github.com/MariaValeriano/Dise-o-Software.git>

17. Enlace del tablero Kanban para seguimiento del Proyecto:

<https://trello.com/invite/b/67bff3fe67a0656758d3d170/ATTI07636873f592fa681b61057c60c7b37e66813628/dise-no-llantas-dc>

18. Documentación en caso de uso

1 IDENTIFICACIÓN DE CASO DE USO			
1.1 Id Caso	CUD 001	1.2 Nombre	Agregar y eliminar productos del carrito
2 HISTÓRICO DE CASO DE USO			
2.1 Autor	Nicolás Morantes		
2.2 Fecha Creación	27/02/2025	2.3 Fecha última Actualización	Na

2.4 Actualizado por	Na	2.5 Versión	1.0
----------------------------	----	--------------------	-----

3 DEFINICIÓN DE UN CASO DE USO

3.1 DESCRIPCIÓN

El sistema debe permitir a los usuarios agregar y eliminar productos del carrito de compras antes de confirmar el pedido.

3.2 ACTORES

Administrador, Usuario

3.3 PRECONDICIONES

El usuario debe contar con datos de acceso al sistema como usuario y contraseña

3.4 FLUJO NORMAL

Contando con las precondiciones el flujo normal será el siguiente:

Paso	Actor	Sistema
1	Darle click encima del producto escogido	El sistema detalla el producto y permite la opción del carrito
2	Clickear el símbolo del carrito y elegir cantidad del producto	El sistema agrega al carrito el numero de productos seleccionados con un máximo los que hay en stock y habilita la opción de confirmar pedido, agregar mas productos o eliminar productos
3	Clickear en la opción de eliminar productos	El sistema separa los productos individualmente (si pide 5 llantas de la misma referencia, aparece 5 veces esa llanta), y aparece una (x) al costado y un botón para guardar cambios
4	Clickear la (x) en el producto seleccionado	El sistema automáticamente elimina el producto seleccionado

3.5 FLUJO ALTERNATIVO

Si existe otra forma de acceder al caso de uso, describir los pasos.

Paso	Actor	Sistema
------	-------	---------

3.6 FLUJO EXCEPCIONAL		
Paso	Actor	Sistema
2.a	Clickear el símbolo del carrito y elegir cantidad del producto	El producto no tiene stock en el momento el sistema deshabilita la opción de agregar ese producto al carrito, habilita la opción de volver al panel de productos
3.7 POS CONDICIONES		
El sistema permite confirmar pedido o volver al panel de productos		
3.8 FRECUENCIA		
Alta		

Diagrama 11. Documentación de caso de uso 001

El resto de la documentación en el Github.