

ZapasPro



Nombre del Estudiante: Rodrigo Pozo Sánchez

Tutores: Sara Villanueva Rosa, Irene Del Rincón Bello

Especialidad del Ciclo Formativo: Desarrollo de Aplicaciones Web

Institución: Universidad Europea de Madrid (Villaviciosa)

Fecha: octubre 2024

1. Introducción

Descripción del Proyecto:

ZapasPro es una plataforma de e-commerce que facilita la compra y gestión de productos en línea. El proyecto nace de la necesidad de ofrecer un sistema e-commerce intuitivo, seguro y eficiente, enfocado en ofrecer una experiencia de usuario optimizada tanto para los clientes como para los administradores de la tienda.

Objetivos Generales:

- Crear una plataforma de e-commerce con un frontend moderno utilizando Angular y Tailwind CSS.
- Establecer un backend en Node.js con TypeScript, utilizando SQLite3 como base de datos.
- Implementar autenticación de usuarios con roles diferenciados entre cliente y administrador.
- Proporcionar una interfaz de administración para gestionar productos y usuarios, con funciones de carrito de compra para clientes.

Motivación:

Dado el crecimiento del comercio electrónico, este proyecto busca resolver las demandas actuales de los consumidores con una plataforma de compras online accesible y funcional, mejorando la experiencia de usuario en cada interacción.

2. Palabras Clave

- E-commerce
- Desarrollo Web
- Angular
- Node.js
- Autenticación de usuarios
- Gestión de productos
- Carrito de compras
- API REST
- SQLite3

Índice

ZapasPro	1
1.Introducción.....	2
2.Palabras Clave	3
3.Módulos Formativos Aplicados	8
4.Herramientas y Lenguajes Utilizados.....	11
5.Componentes del equipo y aportación realizada.....	13
5.1.Planificación del Desarrollo.....	13
5.1.1.Investigación y Definición de Requerimientos.....	13
5.1.2.Diseño de la Arquitectura y Modelado de Datos.....	13
5.1.3.Desarrollo del Backend y API REST	14
5.1.4.Desarrollo del Frontend e Interfaz de Usuario	14
5.1.5.Pruebas de Usabilidad y Revisión de Errores.....	15
5.1.6.Lanzamiento y Documentación Final	15
5.2.Desarrollo de la Práctica	16
5.2.1.Funcionalidad.....	16
5.2.2.Ejecución.....	18
6.Fases del proyecto,	20
6.1.Estudio de Mercado	20
6.1.1.Analizar la temática del proyecto:	20

6.1.2.Realizar un análisis y comparativa de la competencia:	20
6.1.3.Identificar mejoras a realizar:	20
6.1.4.Estudiar el perfil del usuario objetivo:.....	21
6.2.Análisis de la Aplicación	21
6.2.1.Especificar todos los requisitos de la aplicación	21
6.2.2.Enumerar todas las funcionalidades	21
6.2.3.Tendencias del Consumidor.....	22
6.2.4.Oportunidades de Negocio.....	22
6.3.Modelo de Datos	22
6.3.1.Entidades Principales	22
6.3.1.1.Usuario.....	22
6.3.1.2.Producto.....	23
6.3.1.3.Carrito	23
6.3.1.4.Stock	23
6.3.1.5.Relaciones entre Entidades.....	24
6.3.1.6.Diagramas UML.....	25
6.4.Diseño de Interfaces.....	26
6.4.1.Estilo Visual y Estética	26
6.4.2.Estructura y Diseño del Contenedor	26
6.5.Planificación del Desarrollo.....	27
6.5.1.Diseñar un diagrama de Gantt.....	27

6.5.2. Configurar herramientas como Trello, Notion, etc.....	27
6.5.2.1. Estructura del Proyecto.....	28
6.5.2.2. Prioridades:	28
6.5.2.3. Seguimiento de Progreso:.....	28
6.5.2.4. Documentación de Detalles Técnicos:	29
6.5.2.5. Integraciones y Dependencias:	29
6.5.2.6. Optimización de UX/UI:	29
7. Conclusiones y mejoras del proyecto	30
7.1. Conclusiones	30
7.1.1. Logros alcanzados:.....	30
7.1.2. Retos superados:	30
7.1.3. Impacto potencial:.....	31
7.2. Mejoras Identificadas.....	31
7.2.1. Expansión de funcionalidades:	31
7.2.2. Optimización técnica:	31
7.2.3. Diseño y experiencia de usuario (UX/UI):	32
7.2.4. Estrategias de mercado:	32
7.2.5. Escalabilidad del sistema:	32
8. Bibliografía.....	33
9. Anexos	35
9.1. Glosario	35

9.2.Scripts de Ejecución/Instalación de ZapasPro	37
9.3.Manual de usuario	40
9.4. Manual de Administrador	50

3. Módulos Formativos Aplicados

Desarrollo Web en Entorno Cliente

- Creación de una interfaz de usuario interactiva con Angular y Tailwind que permite la navegación fluida entre diferentes secciones de la web.
- Implementación de formularios dinámicos para la gestión de cuentas de usuario, incluyendo registro e inicio de sesión.

Desarrollo Web en Entorno Servidor

- Construcción de varias API REST con Node.js y TypeScript que gestionan las operaciones de creación, lectura, actualización y eliminación (CRUD) de productos, usuarios y stock.
- Implementación de la lógica de negocio para el procesamiento de pedidos y la gestión del carrito de compra.

Despliegue de Aplicaciones Web

- Uso de plataformas de alojamiento Netlify para el Frontend y Render para el Backend
- Configuración de entornos de desarrollo y producción (Creación de entornos separados para el desarrollo y las pruebas antes del lanzamiento, lo que permite realizar cambios sin afectar a los usuarios finales).

Diseño de Interfaces Web

- Creación de un diseño visualmente atractivo que sigue principios de UX/UI para mejorar la experiencia del usuario.
- Realización de pruebas de usabilidad para optimizar la navegación y asegurarse de que los usuarios puedan encontrar fácilmente lo que buscan.

- Diseño responsivo para garantizar que la tienda sea accesible desde dispositivos móviles y tabletas, adaptándose a diferentes tamaños de pantalla.

Formación y Orientación Laboral

- Desarrollo de habilidades de organización esenciales para realizar los objetivos solicitados en el desarrollo del proyecto.
- Utilización de herramientas como Notion para la organización a la hora de priorizar tareas.

Bases de Datos

- Diseño de una base de datos relacional para almacenar información sobre productos, usuarios, stock y transacciones, utilizando SQLite.
- Implementación de consultas eficientes para la recuperación de datos, asegurando un acceso rápido a la información necesaria para la tienda.
- Mantenimiento de la integridad de los datos a través de transacciones y relaciones adecuadas entre las diferentes tablas de la base de datos.

Lenguaje de Marcas

- Uso de HTML, CSS y Tailwind para estructurar y dar estilo a la interfaz de la tienda online, creando un diseño visual atractivo y funcional.
- Implementación de estándares de accesibilidad y SEO a través del correcto uso de etiquetas y atributos, mejorando la visibilidad del sitio en buscadores.
- Integración de elementos multimedia (imágenes, carruseles) que enriquecen la presentación de los productos.
- Implementación de librerías como Bootstrap, Heroicons, Font Awesome y Material Icons para incorporar iconos y componentes estilizados, mejorando la estética y la usabilidad de la web.

Sistemas Informáticos

- Comprensión de la arquitectura de sistemas informáticos, lo que permitió optimizar la infraestructura de la tienda online para un rendimiento adecuado.
- Implementación de software y herramientas de gestión que facilitan el seguimiento de inventario y pedidos.
- Uso de sistemas operativos y servidores para el alojamiento de la aplicación, asegurando un entorno estable y seguro.

Empresa e Iniciativa Emprendedora

- Desarrollo de un plan de negocio que incluye análisis de mercado y estrategias de marketing para posicionar la tienda online en un entorno competitivo.
- Identificación de oportunidades de negocio y evaluación de riesgos asociados al e-commerce.

4. Herramientas y Lenguajes Utilizados

- Angular: Framework JavaScript para el desarrollo del frontend.
- Tailwind CSS: Framework CSS para la creación de interfaces rápidas y responsivas.
- Node.js: Plataforma para el desarrollo de backend y la creación de la API REST.
- HTML: Lenguaje de marcado utilizado para estructurar contenido en la web.
- CSS: Lenguaje de estilo que se usa para diseñar y maquetar la apariencia de un sitio web.
- TypeScript: Superset de JavaScript que añade tipado estático y otras características avanzadas al lenguaje.
- Express.js: Framework de Node.js para gestionar rutas y peticiones del backend.
- SQLite3: Base de datos ligera utilizada para almacenar la información de la plataforma.
- CORS: Configuración para permitir peticiones entre diferentes dominios. Postman: Herramienta para el testeado de la API.
- Git: Sistema de control de versiones para gestionar el código fuente.
- Visual Studio Code: Entorno de desarrollo (IDE) utilizado para escribir y depurar código.
- Notion: Herramientas para la planificación y organización del proyecto.
- Netlify: Plataforma para implementar y alojar aplicaciones web estáticas con integración continua y despliegue automatizado.
- Render: Servicio de alojamiento en la nube para aplicaciones web, APIs, bases de datos y trabajos en segundo plano, diseñado para simplificar la infraestructura y el escalado.
- Sequelize: ORM (Object-Relational Mapping) para Node.js que facilita la interacción con bases de datos SQL mediante modelos y consultas en JavaScript.

- Postman: Herramienta para probar, desarrollar y documentar APIs, permitiendo realizar solicitudes HTTP y gestionar colecciones de pruebas.
- Signals: Mecanismo reactivo para gestionar el estado y las actualizaciones en aplicaciones, comúnmente utilizado en frameworks modernos como Solid.js.
- LocalStorage: API del navegador que permite almacenar datos de manera persistente en el lado del cliente, incluso después de cerrar el navegador.

5. Componentes del equipo y aportación realizada.

El trabajo fue realizado de forma individual por Rodrigo Pozo Sánchez, quien se encargó de todas las tareas relacionadas con la investigación, desarrollo y presentación del proyecto.

5.1. Planificación del Desarrollo

La Planificación del Desarrollo en ZapasPro se estructuró en varias fases para asegurar una implementación organizada y eficiente, desde la fase de investigación hasta el lanzamiento final del e-commerce. Cada fase se describe a continuación:

5.1.1. Investigación y Definición de Requerimientos

Análisis del mercado: Realización de un estudio de mercado para entender las preferencias de los usuarios y los competidores directos, lo cual permitió definir características únicas para el proyecto.

Requerimientos funcionales: Definición de las funcionalidades clave (registro de usuarios, gestión de productos, carrito de compra, etc.) y de los aspectos técnicos necesarios para implementarlas, incluyendo el diseño de interfaces y especificaciones del backend.

Identificación de tecnologías: Selección de herramientas como Angular para el frontend y Node.js con Express para el backend, además de la integración con bases de datos relacionales mediante Sequelize.

5.1.2. Diseño de la Arquitectura y Modelado de Datos

Diseño de la arquitectura: Planificación de la arquitectura MVC (Modelo-Vista-Controlador), segmentando la aplicación en frontend, backend y base de datos.

Modelo de datos: Creación de los modelos de datos necesarios para gestionar los productos, usuarios, roles, y carritos de compra. Este paso incluyó el uso de diagramas UML y la configuración de relaciones entre las entidades para asegurar la integridad y eficacia de las operaciones.

5.1.3. Desarrollo del Backend y API REST

Estructuración de endpoints: Diseño y desarrollo de los endpoints de la API REST necesarios para el funcionamiento de la aplicación. Estos endpoints permiten el manejo de los datos de productos, usuarios, stock y operaciones del carrito.

Seguridad y autenticación: Implementación de autenticación mediante roles para garantizar un acceso seguro a los recursos y operaciones sensibles para los administradores y un acceso restringido a esos recursos para los clientes.

Pruebas de API: Realización de pruebas unitarias por medio de Postman entre otras herramientas para validar el funcionamiento de cada endpoint, asegurando que las respuestas fueran las esperadas y manejando errores comunes de manera adecuada.

5.1.4. Desarrollo del Frontend e Interfaz de Usuario

Desarrollo de componentes: Creación de componentes en Angular para cada sección de la tienda online, incluyendo la página de inicio, catálogo de productos, carrito de compra, perfil de usuario entre otras...

Implementación de UX/UI: Aplicación de principios de diseño centrado en el usuario, optimizando la accesibilidad y el diseño visual. Esto incluyó el uso de Tailwind CSS para estilos y la incorporación de animaciones y transiciones para mejorar la interacción.

Validación de formularios: Implementación de validaciones en formularios para mejorar la experiencia del usuario y asegurar la entrada correcta de datos, como en el registro de usuarios y los formularios de inicio de sesión.

5.1.5. Pruebas de Usabilidad y Revisión de Errores

Pruebas de usabilidad: Realización de pruebas con usuarios para evaluar la navegabilidad, facilidad de uso y funcionalidad general del sitio.

Corrección de errores y optimización: Revisión y corrección de errores detectados durante las pruebas, optimizando el rendimiento de la aplicación y mejorando la estabilidad general.

5.1.6. Lanzamiento y Documentación Final

Preparación para el despliegue: Configuración del entorno de producción y pruebas finales de rendimiento en servidores de pruebas.

- En esta etapa, también se consideraron medidas de escalabilidad para garantizar el funcionamiento óptimo en caso de crecimiento en el tráfico.

Documentación: Elaboración de la documentación técnica y de usuario, incluyendo instrucciones para la gestión de la plataforma y guías de uso para futuros desarrolladores o administradores del sitio.

5.2. Desarrollo de la Práctica

5.2.1. Funcionalidad

El desarrollo de la funcionalidad se centró en crear una experiencia de compra en línea completa, segura y fácil de usar. Las principales funcionalidades implementadas fueron:

Sistema de Autenticación de Usuarios:

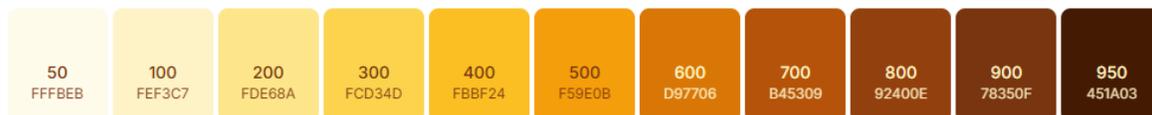
Los usuarios pueden crear cuentas, iniciar sesión y restablecer sus contraseñas de manera sencilla, protegiendo al mismo tiempo sus datos personales.

Gestión de Productos:

Desarrollo de un catálogo de productos en el que los usuarios pueden visualizar todos los artículos disponibles. Cada producto incluye detalles como nombre, descripción, precio, imágenes, colores y tallas.

Integración de filtros y búsqueda avanzada para que los usuarios puedan encontrar productos específicos según categoría, precio, marca, género o color.

Mi paleta de colores:



Fondo principal: 50 (#FFFEBB).

Fondos secundarios (secciones o tarjetas): 100 (#FEF3C7).

Texto principal: 800 (#92400E).

Texto secundario: 700 (#B45309).

Botones principales:

- Fondo: 300 (#FCD34D).
- Hover: 400 (#FBBF24).
- Texto: 900 (#78350F).

Enlaces (links):

- Normal: 500 (#F59E0B).
- Hover: 600 (#D97706).

Bordes o separadores: 700 (#B45309).

Carrito de Compra:

Creación de un sistema de carrito de compra dinámico que permite a los usuarios añadir, actualizar o eliminar productos de su carrito en tiempo real.

Cálculo automático del precio total del pedido con o sin descuento y la cantidad de artículos en el carrito.

Persistencia de datos en el carrito, de modo que los productos añadidos se mantienen incluso si el usuario recarga la página o se desconecta.

Sistema de Stock:

Control del stock de cada producto mediante el modelo de inventario. Cada vez que un producto se añade al carrito o se compra, el sistema reduce automáticamente la cantidad en stock.

Implementación de mensajes de alerta en caso de baja disponibilidad o agotamiento del producto, informando a los usuarios en tiempo real.

Perfil de Usuario:

Los usuarios pueden acceder a su perfil para visualizar su historial de compras y modificar su información personal (nombre, apellido, número de teléfono).

Panel de Administración (para usuarios con rol de administrador):

Desarrollo de un panel de administración donde los administradores pueden gestionar el inventario de productos, actualizar información y agregar o quitar productos.

Administración de usuarios y pedidos, con la posibilidad de ver detalles de cada usuario y su historial de compras, además de realizar ajustes en pedidos pendientes o pasados.

5.2.2. Ejecución

La ejecución de las funcionalidades implementadas en ZapasPro se desarrolló y verificó en entornos de prueba y simulación antes de lanzarse al entorno de producción. Esta fase incluyó pruebas exhaustivas para asegurar que cada función se desempeñara sin problemas y que la experiencia del usuario fuera óptima.

Pruebas de Registro e Inicio de Sesión:

Se realizaron pruebas en múltiples navegadores y dispositivos para confirmar que el sistema de autenticación funcionara correctamente y que los datos de usuario estuvieran seguros.

Validación de credenciales y recuperación de contraseñas, asegurando que el sistema de seguridad y privacidad cumpliera con los requisitos.

Visualización y Filtrado del Catálogo de Productos:

Ejecución de pruebas de usabilidad para asegurar que el catálogo de productos cargara rápidamente y que los filtros y la barra de búsqueda fueran efectivos y precisos.

Optimización de imágenes y contenido multimedia para mejorar los tiempos de carga y la experiencia de usuario.

Gestión del Carrito de Compra:

Pruebas funcionales para verificar la adición, modificación y eliminación de productos en el carrito.

Validación del cálculo total de la compra, asegurando precisión en el cálculo del subtotal, impuestos y envío antes de pasar a la pasarela de pago.

Control de Stock en Tiempo Real:

Pruebas para asegurar que el sistema de inventario reflejara con precisión las cantidades de productos en stock después de cada compra o ajuste.

Verificación de mensajes de advertencia por falta de stock y pruebas para que los usuarios no puedan comprar más de la cantidad disponible.

Panel de Administración:

Pruebas de permisos y accesibilidad para verificar que solo los usuarios con rol de administrador pudieran acceder al panel.

Validación de las funcionalidades de administración de productos, usuarios y pedidos, asegurando que los cambios realizados en el panel se reflejen correctamente en el sitio de cara al usuario.

6. Fases del proyecto,

6.1. Estudio de Mercado

6.1.1. Analizar la temática del proyecto:

El estudio de mercado realizado para ZapasPro tiene como objetivo entender el entorno competitivo, identificar las necesidades de los consumidores y definir las oportunidades de negocio en el sector del e-commerce de calzado deportivo.

6.1.2. Realizar un análisis y comparativa de la competencia:

El análisis de la competencia revela un mercado saturado con varias marcas consolidadas y emergentes. Algunas de las principales competidoras incluyen:

- Zalando: Amplia variedad de marcas y servicio al cliente destacado.
 - Zalando es conocida por su amplia oferta de marcas y excelente servicio al cliente
- Nike y Adidas: Innovación y calidad, líderes del mercado.
 - Nike destaca en el mercado por su innovación en diseño y calidad en productos deportivos
- Tiendas locales de calzado: Implementando plataformas e-commerce para competir regionalmente.

El mercado de calzado deportivo ha mostrado un crecimiento constante en los últimos años, impulsado por el aumento de la conciencia sobre la salud y el bienestar, así como por la creciente popularidad de estilos de vida activos. Se estima que el e-commerce representa una parte significativa de este sector, con un aumento en la compra online debido a la comodidad y la variedad que ofrece.

6.1.3. Identificar mejoras a realizar:

- Segmentación en nichos específicos.
- Estrategias de marketing digital utilizando redes sociales e influencers.

- Enfoque en personalización y experiencia del cliente.

6.1.4. Estudiar el perfil del usuario objetivo:

El público objetivo de ZapasPro incluye:

- Jóvenes y adultos (18-35 años): Interesados en tendencias de moda y tecnología deportiva, que buscan productos de calidad y diseño atractivo.
- Atletas y entusiastas del deporte: Que requieren calzado especializado para diferentes actividades, como running, entrenamiento en gimnasio y deportes de equipo.
- Padres de familia: Que buscan calzado deportivo para sus hijos, valorando la comodidad y la durabilidad de los productos.

6.2. Análisis de la Aplicación

6.2.1. Especificar todos los requisitos de la aplicación

Requisitos funcionales y no funcionales incluyen:

- Gestión de usuarios, productos, carritos de compra y stock.
- Interfaz intuitiva con filtros de búsqueda avanzada.
- Seguridad de datos, velocidad de respuesta y disponibilidad.

6.2.2. Enumerar todas las funcionalidades

- Registro e inicio de sesión de usuarios.
- Catálogo interactivo de productos con descripción detallada.
- Gestión de carritos y finalización de compras.
- Notificaciones sobre promociones y novedades.
- Panel administrativo para gestión de inventario.

6.2.3. Tendencias del Consumidor

Las tendencias actuales en el comportamiento del consumidor incluyen:

- Preferencia por compras online: La comodidad de comprar desde casa y la disponibilidad de opciones de entrega rápida están influyendo en la decisión de compra.
- Interés en productos sostenibles: Cada vez más consumidores buscan marcas que ofrezcan calzado ecológico y responsable.
- Uso de redes sociales: Las plataformas como Instagram y TikTok son fundamentales para la promoción y marketing de productos, influyendo en las decisiones de compra.

6.2.4. Oportunidades de Negocio

Con base en el análisis del sector y la competencia, se identifican las siguientes oportunidades para ZapasPro:

- Segmentación de nichos de mercado: Ofrecer calzado especializado para deportes menos representados, como el senderismo o el ciclismo.
- Estrategias de marketing digital: Utilizar redes sociales y campañas de influencers para llegar a un público más amplio.
- Enfoque en la experiencia del cliente: Mejorar la atención al cliente y ofrecer un servicio de personalización para destacar en un mercado competitivo.

6.3. Modelo de Datos

El modelo de datos para ZapasPro ha sido diseñado utilizando Sequelize como ORM para gestionar eficientemente la información relacionada con los usuarios, productos, carritos de compra y stocks. A continuación, se presentan las principales entidades del modelo, sus atributos y las relaciones entre ellas.

6.3.1. Entidades Principales

6.3.1.1. Usuario

- ID: Identificador único del usuario (número entero, clave primaria).
- Nombre de Usuario: Nombre único del usuario.

- Correo Electrónico: Dirección de correo electrónico única.
- Contraseña: Contraseña del usuario, almacenada de formasegura.
- Rol: Tipo de usuario (admin o client).
- Fecha de Creación: Fecha en que se creó el registro.
- Fecha de Actualización: Fecha de la última modificación del registro.
- Otros Atributos: Incluyen fecha de nacimiento, apellido, número de teléfono, preferencias de compra, etc.

6.3.1.2. Producto

- ID: Identificador único del producto (número entero, clave primaria).
- Nombre: Nombre del producto.
- Descripción: Descripción detallada del producto.
- Precio: Precio de venta del producto.
- Imagen: URL de la imagen del producto.
- Género: Género del producto (masculino, femenino, unisex).
- Color: Color del producto (ejemplo: negro, azul, etc.).
- Marca: Marca del producto (ejemplo: Nike, Adidas, etc.).
- Fecha de Creación: Fecha en que se creó el registro.
- Fecha de Actualización: Fecha de la última modificación del registro.

6.3.1.3. Carrito

- ID: Identificador único del carrito (número entero, clave primaria).
- ID de Usuario: Referencia al usuario asociado al carrito.
- ID de Producto: Referencia al producto en el carrito.
- Cantidad: Cantidad de productos en el carrito.

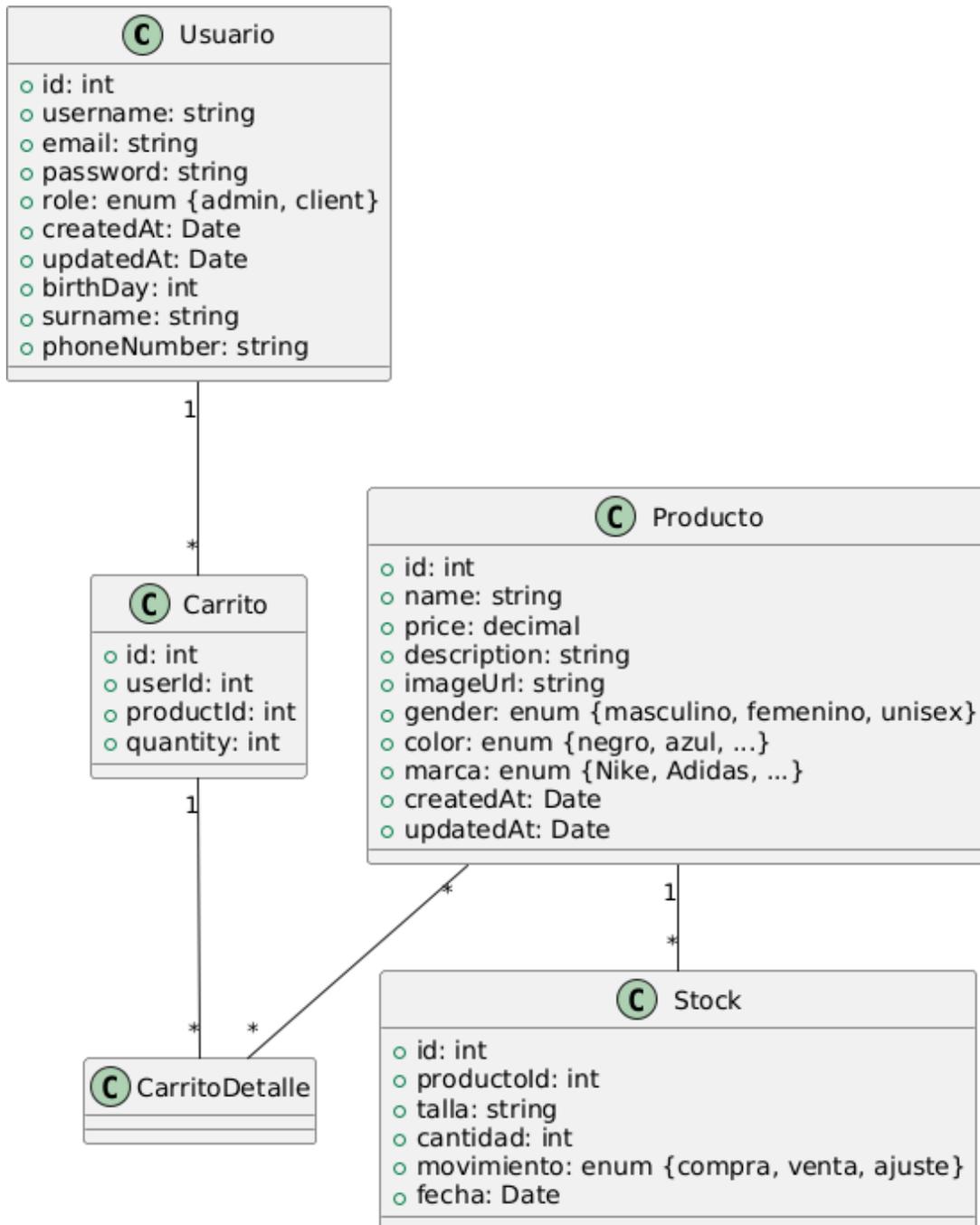
6.3.1.4. Stock

- ID: Identificador único del registro de stock (número entero, clave primaria).
- ID de Producto: Referencia al producto al que pertenece el stock.
- Talla: Talla del producto (ejemplo: 38, 39, 40).
- Cantidad: Cantidad disponible de esa talla.
- Movimiento: Tipo de movimiento de stock (compra, venta, ajuste).
- Fecha: Fecha del movimiento de stock.

6.3.1.5. Relaciones entre Entidades

- Un Usuario puede tener múltiples Carritos (relación uno a muchos).
- Un Carrito puede contener múltiples Productos, y un Producto puede estar en múltiples Carritos (relación muchos a muchos).
- Un Producto puede tener múltiples registros en Stock (relación uno a muchos).
- Cada Stock está asociado a un único Producto (relación muchos a uno).

6.3.1.6. Diagramas UML



6.4. Diseño de Interfaces

El diseño de interfaces en ZapasPro busca proporcionar una experiencia de usuario atractiva y funcional, combinando una estética moderna con una estructura clara. Los elementos clave del diseño, basados en el código proporcionado, se describen a continuación:

6.4.1. Estilo Visual y Estética

Paleta de Colores: Se ha elegido una paleta de colores cálidos, con tonos crema y amarillo predominantes. El fondo del contenedor es de color crema claro (#f3e8d5), lo cual crea una atmósfera amigable y profesional que da la bienvenida a los usuarios.

Tipografía: Se utiliza tipografía de gran tamaño y peso en el encabezado ("Iniciar Sesión") en color amarillo oscuro, lo que asegura que este mensaje clave sea claramente visible y llame la atención del usuario.

6.4.2. Estructura y Diseño del Contenedor

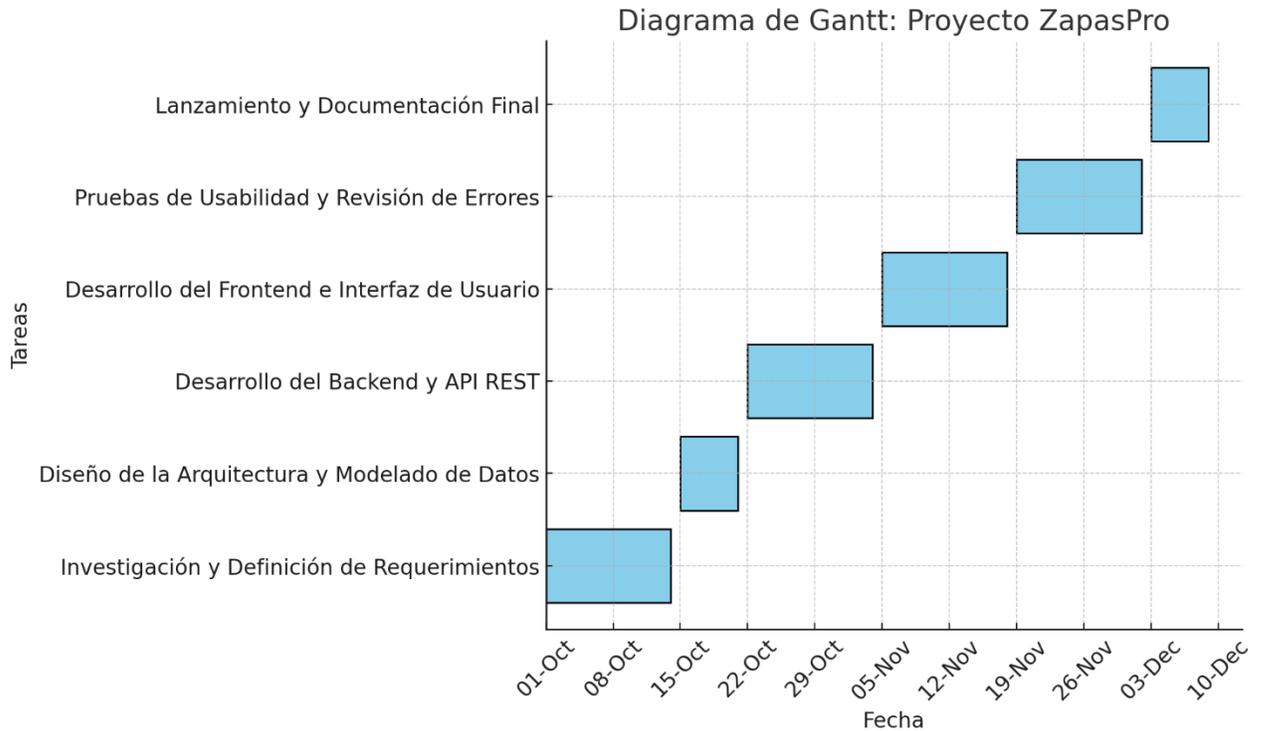
Contenedor Centralizado: La interfaz está centrada en la pantalla, con un contenedor principal de fondo blanco y esquinas redondeadas que se eleva sobre el fondo. Este diseño limpio y minimalista permite que el área de contenido principal destaque, facilitando la atención del usuario en la acción deseada.

Este diseño de interfaces en ZapasPro utiliza principios visuales y de usabilidad para crear una experiencia atractiva y eficiente. El uso de Tailwind CSS y la integración de efectos de transformación permiten una navegación intuitiva y fluida, logrando una interfaz que invita a los usuarios a interactuar cómodamente con la plataforma.

6.5. Planificación del Desarrollo

6.5.1. Diseñar un diagrama de Gantt

Diagrama detallado que divide las tareas en fases específicas: investigación, diseño, desarrollo, pruebas y despliegue.



6.5.2. Configurar herramientas como Trello, Notion, etc.

La planificación de desarrollo fue realizada con Notion para el proyecto ZapasPro se enfocó en estructurar, priorizar y documentar las tareas necesarias para llevar a cabo el desarrollo del proyecto.

La herramienta Notion se utilizó como un sistema centralizado para organizar el flujo de trabajo y garantizar que todas las partes del proyecto estuvieran claramente definidas.

Aquí están los aspectos clave de la planificación:

6.5.2.1. Estructura del Proyecto

Las tareas se categorizaron y agruparon según funcionalidades específicas o componentes clave del sistema, como:

- Responsividad del diseño.
- Gestión de stock.
- Panel de administración.
- Carrito de compras y funcionalidades relacionadas.
- Historial y gestión de compras.
- Perfil del usuario y configuración.
- Estilización y mejoras visuales.

6.5.2.2. Prioridades:

Cada tarea se identificó con un propósito claro y se ordenó según su nivel de importancia o dependencia. Por ejemplo:

Tareas críticas: "Actualizar stock cuando se hacen compras", "Poder devolver un producto", "Revisar funcionamiento de descuentos en carrito".

Tareas de usabilidad: "Hacer responsive", "Mostrar en recibos el precio con y sin descuento", "Añadir paginación cada 18 productos".

Tareas de diseño y detalles: "Hover para el botón de corazón", "Mejorar estilos", "Mantener el color del corazón si está en favoritos".

6.5.2.3. Seguimiento de Progreso:

Cada tarea incluía un enlace único, permitiendo detallar el estado, responsables y las acciones pendientes. Esto aseguraba una fácil trazabilidad y actualización en tiempo real.

Las etiquetas como "completado", "en progreso" o "pendiente" indicaban el estado de desarrollo de cada actividad.

6.5.2.4. Documentación de Detalles Técnicos:

Se definieron los requisitos específicos de cada funcionalidad. Por ejemplo:

Carrito: "Tendrá dos botones, uno para pasar por caja y otro para ir a la cesta".

Perfil: "Crear todas las secciones de perfil con opciones como configuración de cuenta y devolución de productos".

Paginación: "Mostrar todas las páginas disponibles para ir directamente a una específica".

6.5.2.5. Integraciones y Dependencias:

Tareas relacionadas estaban vinculadas entre sí para reflejar dependencias. Por ejemplo:

"Modificar el stock después de hacer la compra" está relacionado con "Actualizar stock cuando se hacen compras".

"Mandar cambios a la base de datos" está vinculado a tareas de actualización de datos en tiempo real.

6.5.2.6. Optimización de UX/UI:

Varias tareas específicas se orientaron a mejorar la experiencia del usuario final, como:

Diseño responsive: Mencionado en múltiples entradas para garantizar accesibilidad en distintos dispositivos.

Interactividad: "Mostrar desplegable al añadir al carrito", "Añadir opción de seleccionar talla para añadir al carrito".

7. Conclusiones y mejoras del proyecto

El desarrollo de ZapasPro ha permitido crear una solución sólida y funcional en el ámbito del comercio electrónico, logrando cumplir con la mayoría de los objetivos planteados al inicio del proyecto. A través de este trabajo, se han obtenido importantes aprendizajes tanto técnicos como estratégicos, así como la identificación de áreas con un alto potencial de mejora.

7.1. Conclusiones

7.1.1. Logros alcanzados:

Implementación técnica exitosa: ZapasPro cuenta con una arquitectura basada en Angular para el frontend y Node.js para el backend, lo que ha permitido un desempeño eficiente y una experiencia de usuario fluida. Además, el uso de Tailwind CSS ha contribuido a un diseño atractivo y responsivo.

Gestión integral de funcionalidades: Se han desarrollado módulos clave como el registro de usuarios, el sistema de carrito de compras, la gestión de productos y el control de stock, lo que asegura una experiencia completa tanto para los clientes como para los administradores.

Foco en la experiencia de usuario: La aplicación ha sido diseñada con principios de UX/UI modernos, priorizando la navegación intuitiva, la rapidez de respuesta y la accesibilidad desde dispositivos móviles.

Despliegue eficiente: La plataforma ha sido implementada en Netlify (frontend) y Render (backend), utilizando entornos separados de desarrollo y producción, lo que facilita futuras actualizaciones sin interrupciones en el servicio.

7.1.2. Retos superados:

Se lograron resolver problemas relacionados con la configuración de la base de datos y la optimización de las consultas, utilizando SQLite3 como solución inicial.

La integración de validaciones robustas en formularios y sistemas de autenticación asegura la integridad de los datos y la protección de la privacidad del usuario.

7.1.3. Impacto potencial:

ZapasPro tiene el potencial de competir en el sector del e-commerce gracias a su enfoque en la personalización y su capacidad de adaptarse a las necesidades cambiantes de los consumidores.

7.2. Mejoras Identificadas

Aunque el proyecto cumple con sus objetivos fundamentales, se han identificado áreas clave para su evolución y mejora, orientadas a fortalecer su competitividad en el mercado y mejorar la experiencia del usuario.

7.2.1. Expansión de funcionalidades:

Sistema de recomendaciones: Implementar algoritmos de aprendizaje automático que sugieran productos basados en el historial de compras y las preferencias del usuario.

Promociones y descuentos personalizados: Incluir notificaciones automáticas que informen a los clientes sobre promociones específicas según su perfil de compra.

Revisión de productos: Integrar una sección que permita a los usuarios valorar y comentar productos, incrementando la confianza de futuros compradores.

7.2.2. Optimización técnica:

Base de datos más robusta: Migrar de SQLite3 a una solución más escalable como PostgreSQL o MongoDB para manejar mayores volúmenes de datos y consultas complejas.

Carga más eficiente: Aplicar técnicas avanzadas de compresión y caching para reducir los tiempos de carga, especialmente en redes lentas.

Seguridad avanzada: Incorporar herramientas para análisis continuo de vulnerabilidades, detección de intrusiones y encriptación más robusta de datos sensibles.

7.2.3. Diseño y experiencia de usuario (UX/UI):

Interfaz más interactiva: Mejorar el diseño visual con animaciones suaves y elementos interactivos que hagan más atractiva la navegación.

Accesibilidad mejorada: Incorporar opciones para personas con discapacidades visuales o auditivas, como la compatibilidad con lectores de pantalla y controles por teclado.

Diseño de filtros: Ampliar los filtros de búsqueda para incluir parámetros adicionales como talla, marca y disponibilidad en tiempo real.

7.2.4. Estrategias de mercado:

Internacionalización: Implementar soporte para múltiples idiomas y monedas, lo que facilitará la expansión a mercados internacionales.

Marketing digital: Diseñar campañas en redes sociales y colaboraciones con influencers para aumentar la visibilidad de ZapasPro entre públicos jóvenes.

Sostenibilidad: Desarrollar una línea de productos eco-friendly para captar consumidores interesados en opciones sostenibles.

7.2.5. Escalabilidad del sistema:

Soporte para mayor tráfico: Optimizar la infraestructura actual para soportar un aumento en el número de usuarios y transacciones simultáneas.

Servicios complementarios: Incluir funcionalidades como seguimiento de pedidos, devoluciones automatizadas y un sistema de puntos de fidelidad para la retención de clientes.

8. Bibliografía

- Angular. (s.f.). Angular - The modern web developer's platform. Obtenido de <https://angular.io/>
- Tailwind Labs. (s.f.). Tailwind CSS Documentation. Obtenido de <https://tailwindcss.com/docs>
- Node.js. (s.f.). Node.js. Obtenido de <https://nodejs.org/>
- Express.js. (s.f.). Express - Node.js web application framework. Obtenido de <https://expressjs.com/>
- SQLite. (s.f.). SQLite Documentation. Obtenido de <https://www.sqlite.org/docs.html>
- Sequelize. (s.f.). Sequelize ORM for Node.js. Obtenido de <https://sequelize.org/>
- Postman. (s.f.). Postman API Platform. Obtenido de <https://www.postman.com/>
- Netlify. (s.f.). Netlify: Develop & deploy the best web experiences. Obtenido de <https://www.netlify.com/>
- Render. (s.f.). Render: Cloud Hosting for Developers. Obtenido de <https://render.com/>
- Git. (s.f.). Git. Obtenido de <https://git-scm.com/>
- Visual Studio Code. (s.f.). Code Editing. Redefined. Obtenido de <https://code.visualstudio.com/>
- GitHub. (s.f.). GitHub: Where the world builds software. Obtenido de <https://github.com/>
- Bootstrap. (s.f.). Bootstrap: The most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites. Obtenido de <https://getbootstrap.com/>
- Heroicons. (s.f.). Heroicons - Free, open-source icons by the makers of Tailwind CSS. Obtenido de <https://heroicons.com/>

- Font Awesome. (s.f.). Font Awesome: The world's most popular icon set and toolkit. Obtenido de <https://fontawesome.com/>
- Material Icons. (s.f.). Material Icons. Obtenido de <https://material.io/resources/icons/>
- Notion. (s.f.). Notion: The all-in-one workspace for your notes, tasks, wikis, and databases. Obtenido de <https://www.notion.so/>
- CORS. (s.f.). CORS - Cross-Origin Resource Sharing. Obtenido de <https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>
- Google Fonts. (s.f.). Google Fonts: Free web fonts. Obtenido de <https://fonts.google.com/>
- Google Analytics. (s.f.). Google Analytics. Obtenido de <https://analytics.google.com/>
- UX Design. (s.f.). UX Design - Understanding user experience. Obtenido de <https://uxdesign.cc/>
- Zalando. (2024). Acerca de Zalando. <https://www.zalando.com>
- Nike. (2024). Historia de Nike. <https://www.nike.com>

9. Anexos

9.1. Glosario

- Ecommerce: Comercio electrónico, permite realizar compras y ventas a través de internet.
- Autenticación de usuarios: Proceso de verificar la identidad de los usuarios.
- Roles de usuario: Diferentes permisos asignados a los usuarios según perfil, como cliente o administrador.
- Administración de productos: Gestión de los productos, incluyendo su creación, actualización, y eliminación.
- Carrito de compras: Herramienta que permite a los usuarios añadir y gestionar productos antes de la compra.
- Angular: Framework frontend para construir aplicaciones web dinámicas.
- Tailwind CSS: Framework CSS para crear diseños responsivos y personalizables.
- Node.js: Entorno de servidor que permite ejecutar JavaScript en el backend.
- SQLite3: Base de datos ligera para almacenar y gestionar datos localmente.
- API REST: Interfaz de comunicación que permite el intercambio de datos entre cliente y servidor.
- CRUD: Conjunto de operaciones básicas en bases de datos: Crear, Leer, Actualizar y Eliminar.
- Frontend: Parte visible de la aplicación con la que interactúan los usuarios.
- Backend: Parte de la aplicación donde se maneja la lógica y gestión de datos.
- Interfaz de usuario: Diseño visual a través del cual los usuarios interactúan con la aplicación.
- UX/UI: Diseño de experiencia y usabilidad, enfocado en optimizar la interacción del usuario.
- Autenticación: Verificación de identidad del usuario para acceso seguro.
- Seguridad: Medidas para proteger la aplicación y los datos del usuario. Base de datos: Almacenamiento organizado de información utilizada por la aplicación.
- MVC: Modelo de arquitectura que separa la aplicación en Modelo, Vista y Controlador.
- Express.js: Framework de Node.js para la creación de servidores y gestión de rutas.

- CORS: Mecanismo de seguridad que permite o restringe solicitudes entre dominios.
- JSON: Formato ligero de intercambio de datos entre cliente y servidor.
- Control de versiones: Sistema que permite gestionar y mantener versiones del código.
- Git: Herramienta para el control de versiones distribuido del código fuente.
- Visual Studio Code: Editor de texto y entorno de desarrollo para escribir y depurar código.
- Responsividad: Capacidad de la aplicación de adaptarse a distintos dispositivos y pantallas.
- JavaScript: Lenguaje de programación utilizado en el frontend y backend.
- Framework: Conjunto de herramientas y librerías para facilitar el desarrollo de software.
- Middleware: Funciones intermedias que procesan peticiones en una aplicación.
- Desarrollo ágil: Metodología de desarrollo basada en la colaboración y entrega continua.
- Componentes: Partes reutilizables del código que forman la estructura de la interfaz.
- Modularización: Organización del código en módulos para su mejor mantenimiento.
- Cliente-servidor: Modelo de arquitectura donde el cliente solicita y el servidor responde.
- SPA: Aplicación de una sola página que carga dinámicamente sus contenidos.
- Bootstrap: Framework CSS para diseñar interfaces responsivas y atractivas.
- ORM: Herramienta que facilita el acceso y manipulación de bases de datos.
- SQLite: Sistema de gestión de bases de datos SQL, usado para almacenamiento local.
- Mockups: Representación visual del diseño de la interfaz de usuario.
- Prototipado: Creación de modelos preliminares de una aplicación para pruebas y mejoras.
- Diseño de interfaces: Creación de pantallas y elementos visuales para la interacción.
- Routing: Proceso de navegación entre las diferentes secciones de una aplicación.
- HTTP: Protocolo de comunicación para el intercambio de datos en la web.
- Postman: Herramienta para testear y verificar APIs.
- Gestión de inventario: Control y administración de los productos en la tienda.
- Sistema de autenticación: Proceso para asegurar la identidad del usuario.
- Roles de administrador y cliente: Diferentes niveles de acceso según el tipo de usuario.
- API de compras: Conjunto de funciones que permiten el proceso de compra en línea.

- JSON Web Token (JWT): Sistema de autenticación basado en tokens encriptados.
- Sesiones: Información temporal de la actividad del usuario durante su uso de la aplicación.
- Heroku: Plataforma en la nube para el despliegue de aplicaciones.

9.2. Scripts de Ejecución/Instalación de ZapasPro

1. Node.js: Asegúrate de tener instalado Node.js (versión LTS recomendada).
2. Angular CLI: Instala Angular CLI de manera global si no lo tienes:
npm install -g @angular/cli
3. Git: Para clonar los repositorios, asegúrate de tener Git instalado.
4. SQLite3: Verifica que tienes SQLite3 instalado o usa un archivo .db incluido en el proyecto.

Repositorios

Front-End: ZapasPro_Front (https://github.com/rodrigopozodev/ZapasPro_Front)

Back-End: ZapasPro_Back (https://github.com/rodrigopozodev/ZapasPro_Back)

Configuración del Repositorio Back-End

- Clonación del Repositorio:

Para clonar usamos **git clone** mas la url del repositorio

git clone https://github.com/rodrigopozodev/ZapasPro_Back

- Instalación de Dependencias

Entramos en la carpeta y ejecutamos **npm install** para instalar todo lo necesario

1. Revisa el archivo .env o crea uno si no existe. Define las variables necesarias, como la ruta de la base de datos, puerto, etc. Un ejemplo:

```
PORT=3000
```

```
DATABASE=./database/zapaspro.db
```

2. Si necesitas migrar la base de datos, utiliza un script **npm run migrate**

Ejecución

Para iniciar el servidor utilizamos **npm run start**

Esto levantará el servidor en el puerto especificado en .env

Configuración del Repositorio Front-End

Clonación del Repositorio

Utilizamos git clone https://github.com/rodrigopozodev/ZapasPro_Front

- Instalación de Dependencias

Entramos en la carpeta y ejecutamos **npm install** para instalar todo lo necesario

- Configuración

Asegúrate de configurar la URL del servidor Back-End en el archivo de entorno de Angular (src/environments/environment.ts).

- Ejecución

Para iniciar la aplicación en modo de desarrollo: `ng serve`

Por defecto, Angular servirá la aplicación en `http://localhost:4200`.

Conexión entre Front-End y Back-End

1. Asegúrate de que ambos servicios están corriendo:

Front-End en <http://localhost:4200>

Back-End en <http://localhost:3000>

2. Navega al Front-End y prueba las funcionalidades, asegurándote de que puede comunicarse con el servidor Back-End.

3. Scripts Adicionales

Back-End

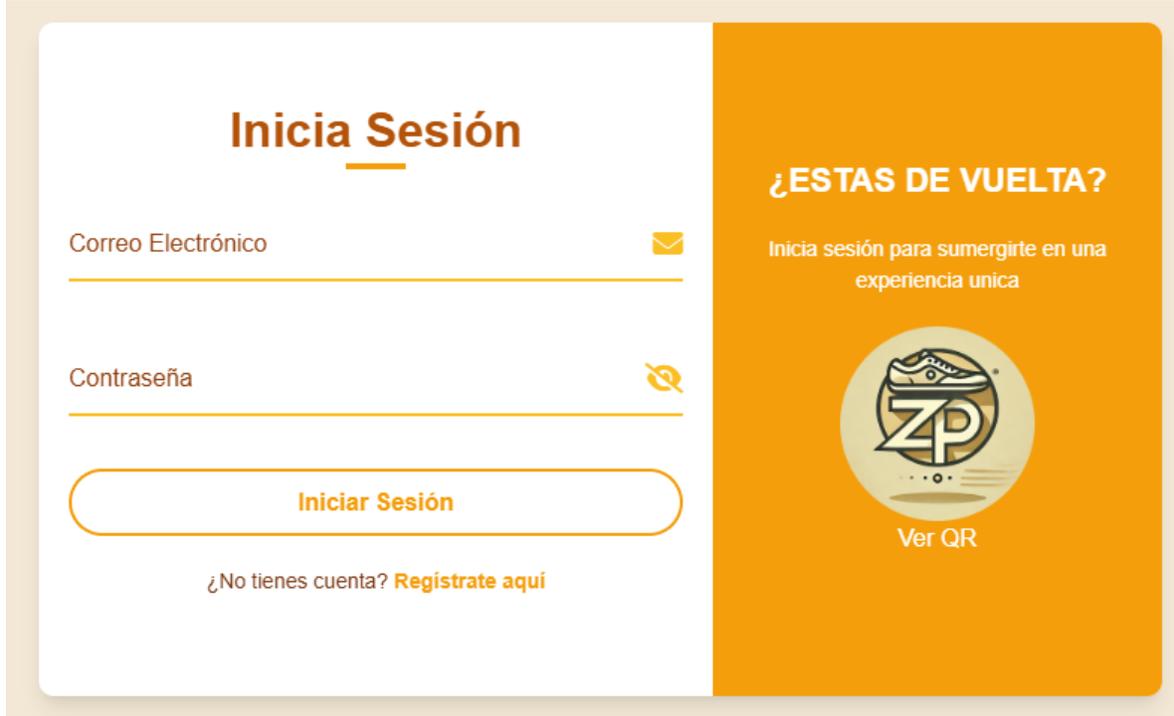
- Linting y corrección de código **npm run lint**
- Ejecución en modo desarrollo (con nodemon) **npm run dev**
- Pruebas unitarias **npm test**

Front-End

- Build para producción **ng build --prod**
- Linting **ng lint**

9.3. Manual de usuario

Primero entramos en <https://zapaspro.netlify.app> y nos carga el formulario para iniciar sesión en la web



Inicia Sesión

Correo Electrónico 

Contraseña 

Iniciar Sesión

¿No tienes cuenta? [Regístrate aquí](#)

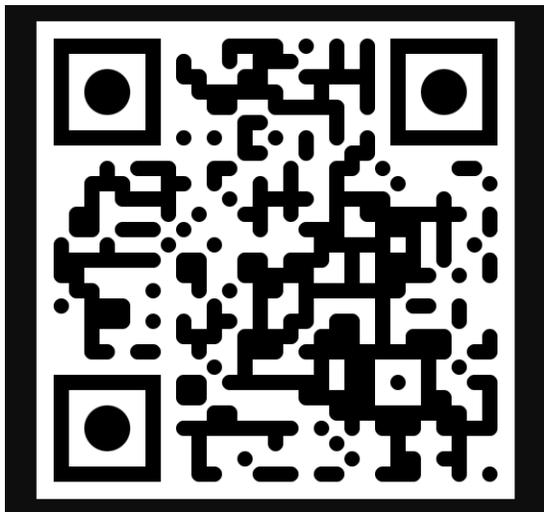
¿ESTAS DE VUELTA?

Inicia sesión para sumergirte en una experiencia unica

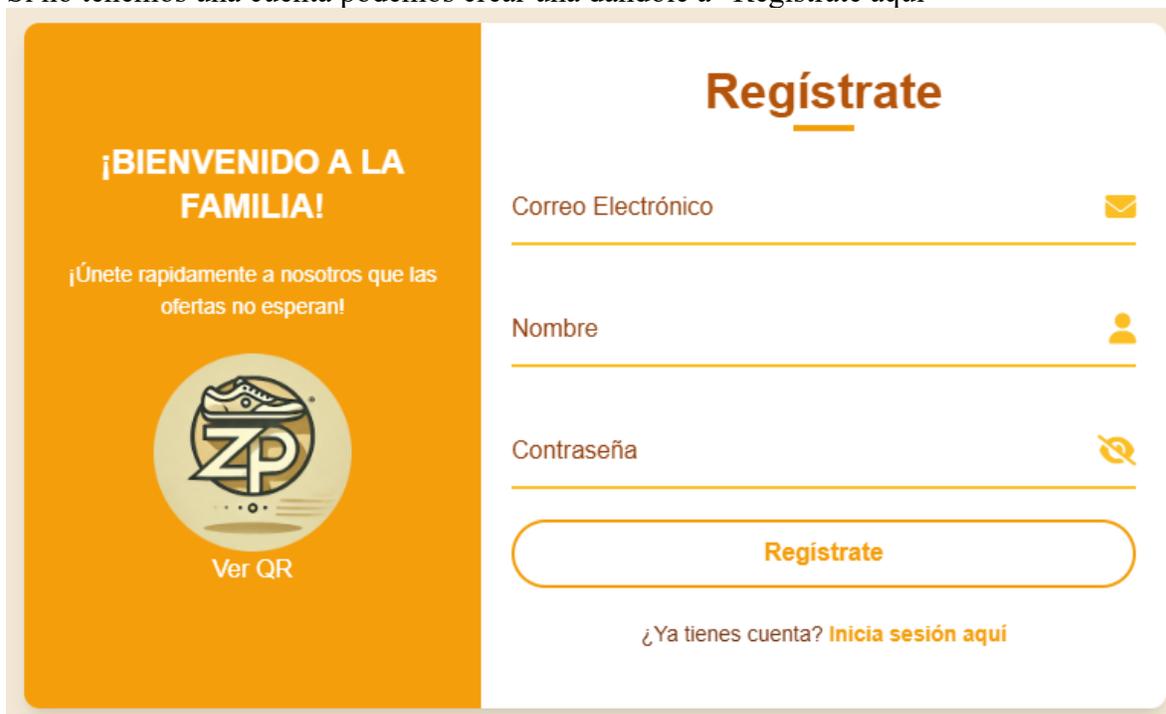


[Ver QR](#)

Si queremos compartir el enlace de la web podemos dando donde pone “Ver QR” y mostrara este QR para que lo puedan escanear otros posibles nuevos clientes.



Si no tenemos una cuenta podemos crear una dándole a “Regístrate aquí”



The registration form is split into two main sections. On the left, an orange background contains the text "¡BIENVENIDO A LA FAMILIA!" and "¡Únete rápidamente a nosotros que las ofertas no esperan!". Below this is a circular QR code with the ZP logo and the text "Ver QR". On the right, a white background features the heading "Regístrate" in orange. Below the heading are three input fields: "Correo Electrónico" with an envelope icon, "Nombre" with a person icon, and "Contraseña" with an eye icon. A large orange button labeled "Regístrate" is positioned below the fields. At the bottom, there is a link: "¿Ya tienes cuenta? [Inicia sesión aquí](#)".

Una vez ya iniciada sesión apareceremos en la tienda



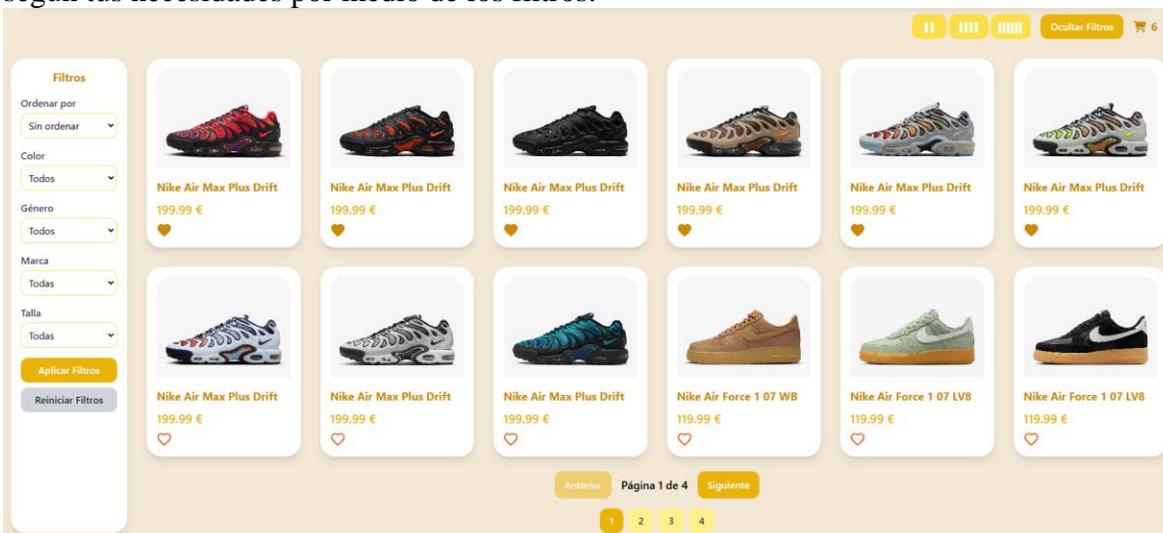
Esta página se divide en 3 secciones:

La barra de navegación la cual estará a partir de ahora en todas las páginas. Nos permite viajar entre la tienda, el carrito, el perfil. También nos permite desloguearnos.

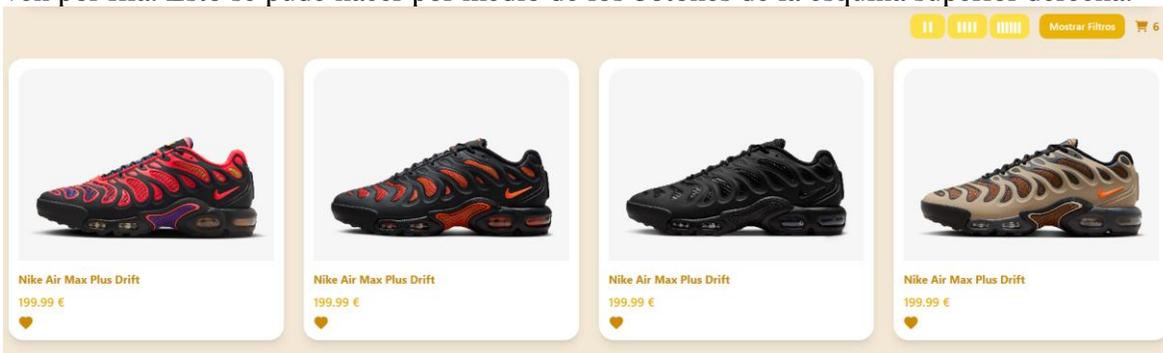
La segunda sección es un carrusel el cual nos mostrara productos, descuentos , etc..



y la última sección son las tarjetas de productos las cuales te permiten filtrar los productos según tus necesidades por medio de los filtros.

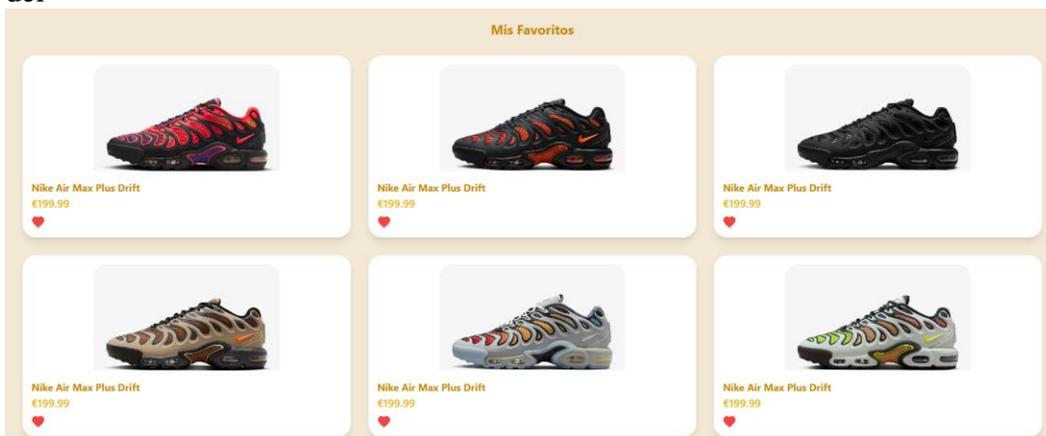


También tenemos la opción de ocultar los filtros y de cambiar el numero de zapatillas que se ven por fila. Esto se pude hacer por medio de los botones de la esquina superior derecha.

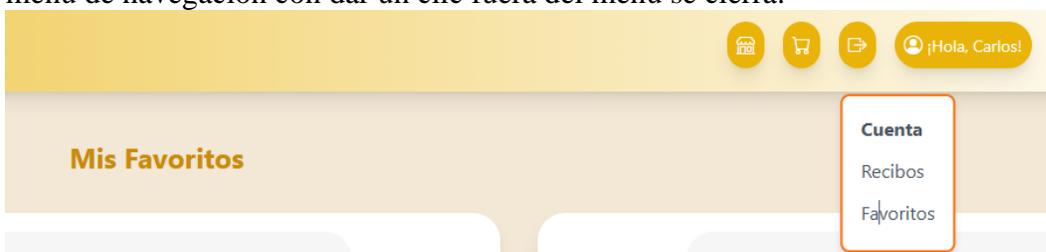


Al lado de mostrar/ocultar filtros tenemos un icono que nos indicara cuantos productos tenemos en el carrito.

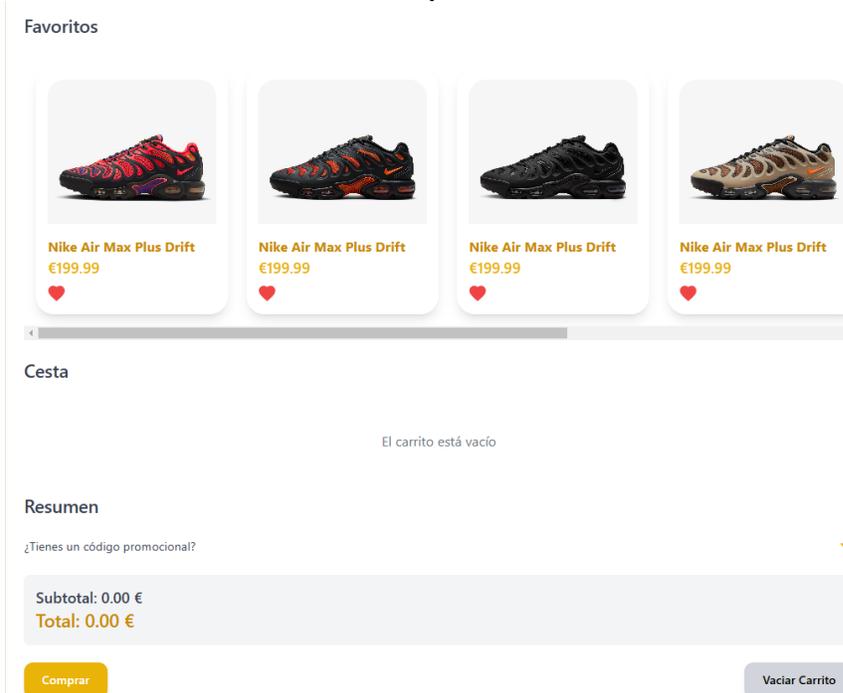
Los productos favoritos podremos encontrarlos una vez añadidos en la sección de favoritos del perfil



Para acceder a esta sección tenemos que darle a el icono de perfil con nuestro nombre. En este caso “¡Hola, Carlos!” y nos aparecerá un botón que pone cuenta y le damos y entonces saldrán las secciones del perfil que son favoritos y recibos. Si queremos que desaparezca el menú de navegación con dar un clic fuera del menú se cierra.



Los favoritos también estarán disponibles desde el carrito.



Desde la tienda, sección de favoritos o el carrito podemos quitar los productos de favoritos.

volviendo a la tienda se le damos click en cualquier foto de un producto de las tarjetas de producto podemos viajar a la sección del producto.



Esta es la sección del producto



Nike Dunk Low Retro

Siempre se puede contar con un clásico. Este diseño estilo Color Block combina piel y ante con un acolchado suave para ofrecer una comodidad revolucionaria en todo momento. Las posibilidades son infinitas. ¿Cómo vas a llevar tus Dunk?

Precio: 119.99€



Selecciona tu talla

36	37	38	39
40	41	42	43
44	45		

Añadir al carrito



En esta sección tenemos dividido el contenido en tres columnas:

En la primera columna se encuentra una galería de imágenes que al seleccionarla o pasar el ratón por encima mostrara las fotos de la zapatilla en la columna dos



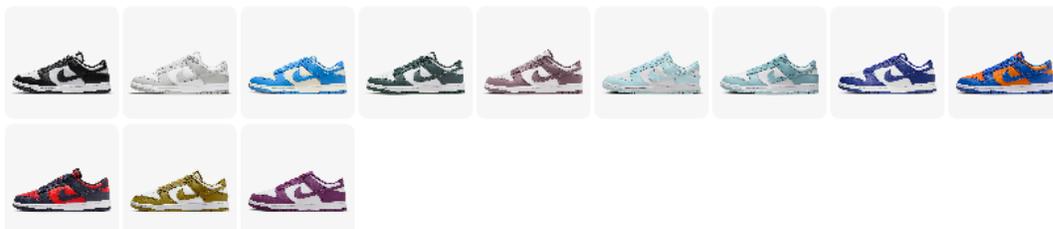
La columna dos siempre mostrara la foto genérica de la zapatilla a no ser que se cambien seleccionando otra de la columna uno.

En la columna tres tenemos el nombre de la zapatilla, descripción, precio, un menú para poder viajar entre otros colores de la zapatilla, podemos seleccionar la talla y añadirla al carrito y podemos añadir o quitar el producto de favoritos.

Nike Dunk Low Retro

Siempre se puede contar con un clásico. Este diseño estilo Color Block combina piel y ante con un acolchado suave para ofrecer una comodidad revolucionaria en todo momento. Las posibilidades son infinitas. ¿Cómo vas a llevar tus Dunk?

Precio: 119.99€



Selecciona tu talla

36	37	38	39
40	41	42	43
44	45		

Añadir al carrito



Al seleccionar la talla se mostrará la seleccionada encima del botón de añadir al carrito

Selecciona tu talla

36	37	38	39
40	41	42	43
44	45		

Talla seleccionada: EU 40

Añadir al carrito

Una vez le demos añadir al carrito aparecerá en pantalla este menú el cual te permite añadir más cantidad de ese producto, eliminarlo o disminuir la cantidad si la hemos aumentado. Podemos realizar la compra directamente desde este menú o podemos ir a la cesta. Si queremos añadir más productos le daremos a cerrar.

Precio: 119.99€

Carrito



Nike Dunk Low Retro
Talla 40
119.99 €

 **1** 

Cantidad: 1

Subtotal: 119.99 €
Total: 119.99 €

[Comprar](#) [Ir a la cesta](#) [Cerrar](#)

Podemos añadir el mismo producto con otras tallas.

Carrito



Talla 40
119.99 €

- 5 +
Cantidad: 5



Nike Dunk Low Retro
Talla 42
119.99 €

 1 +
Cantidad: 1



Nike Dunk Low Retro
Talla 44
119.99 €

Subtotal: 839.93 €

Total: 839.93 €

Comprar

Ir a la cesta

Cerrar

sí le damos a ir a la cesta podemos ver que si escribimos el código promocional ZapasProMola que hay en el carrusel tenemos un 15% de descuento.
Cesta

 <p>Nike Dunk Low Retro Talla 40 119.99 € 101.99 €</p> <p> 1  Cantidad: 1</p>	 <p>Nike Dunk Low Retro Talla 41 119.99 € 101.99 €</p> <p> 2  Cantidad: 2</p>	 <p>Nike Dunk Low Retro Talla 42 119.99 € 101.99 €</p> <p> 1  Cantidad: 1</p>	 <p>Nike Dunk Low Retro Talla 42 119.99 € 101.99 €</p> <p> 1  Cantidad: 1</p>
--	--	--	--

Resumen

¿Tienes un código promocional?

Aplicar Descuento

Subtotal: 599.95 €

Descuento aplicado: -89.99 €

Total: 509.96 €

Comprar

Vaciar Carrito

Desde el carrito también se puede añadir, eliminar, o decrementar cualquier producto. Además, podemos vaciar el carrito o realizar la compra que mandara nuestro recibo a la sección de recibos en el perfil.

Recibos de compras

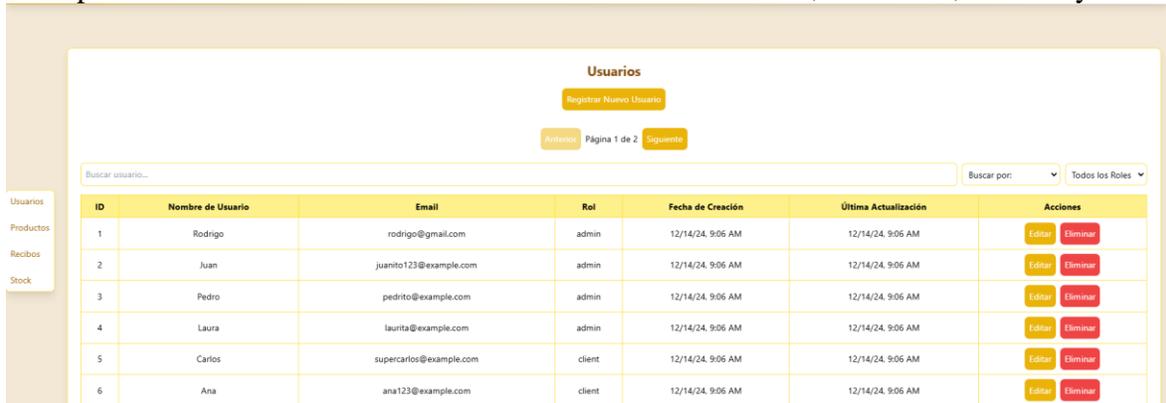
Usuario: Rodrigo
Fecha: 2/12/2024 - Hora: 8:46:24

 <p>Nike Dunk Low Retro Talla: 47 Precio: 119.99€ Cantidad: 1</p>	 <p>Nike Dunk Low Retro Talla: 48 Precio: 119.99€ Cantidad: 5</p>
---	---

Total: 719.9399999999999€

9.4. Manual de Administrador

En el panel de administrador tenemos cuatro secciones Usuarios, Productos, Recibos y Stock:



The screenshot shows the 'Usuarios' section of the administrator panel. At the top, there is a 'Registrar Nuevo Usuario' button and a pagination indicator 'Página 1 de 2'. Below this is a search bar with the text 'Buscar usuario...' and a dropdown menu for 'Buscar por' set to 'Todos los Roles'. The main content is a table with the following data:

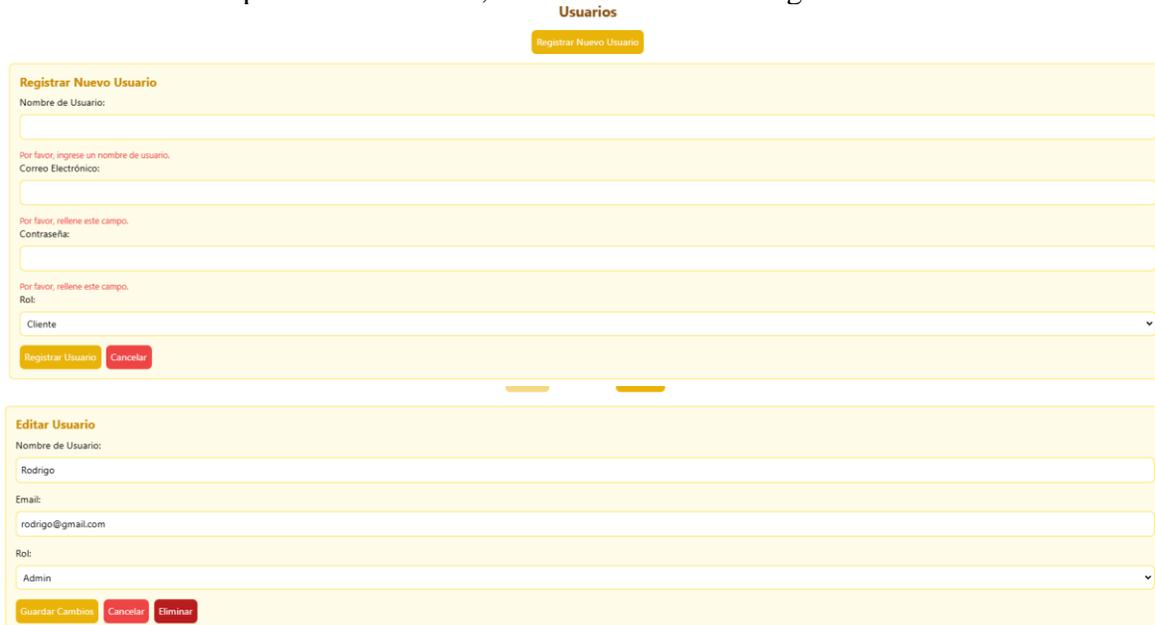
ID	Nombre de Usuario	Email	Rol	Fecha de Creación	Última Actualización	Acciones
1	Rodrigo	rodrigo@gmail.com	admin	12/14/24, 9:06 AM	12/14/24, 9:06 AM	Editar Eliminar
2	Juan	juanito123@example.com	admin	12/14/24, 9:06 AM	12/14/24, 9:06 AM	Editar Eliminar
3	Pedro	pedrito@example.com	admin	12/14/24, 9:06 AM	12/14/24, 9:06 AM	Editar Eliminar
4	Laura	laurita@example.com	admin	12/14/24, 9:06 AM	12/14/24, 9:06 AM	Editar Eliminar
5	Carlos	supercarlos@example.com	client	12/14/24, 9:06 AM	12/14/24, 9:06 AM	Editar Eliminar
6	Ana	ana123@example.com	client	12/14/24, 9:06 AM	12/14/24, 9:06 AM	Editar Eliminar

On the left side, there is a sidebar menu with the following items: Usuarios, Productos, Recibos, and Stock.

podemos navegar entre secciones por medio de este menú



En usuarios podemos editar, eliminar o registrar nuevos usuarios.



The image shows two screenshots of the user management forms. The first is the 'Registrar Nuevo Usuario' form, which has the following fields: 'Nombre de Usuario:', 'Correo Electrónico:', 'Contraseña:', and 'Rol:' (with a dropdown menu set to 'Cliente'). There are 'Registrar Usuario' and 'Cancelar' buttons at the bottom. The second is the 'Editar Usuario' form, which has the following fields: 'Nombre de Usuario:' (pre-filled with 'Rodrigo'), 'Email:' (pre-filled with 'rodrigo@gmail.com'), and 'Rol:' (with a dropdown menu set to 'Admin'). There are 'Guardar Cambios', 'Cancelar', and 'Eliminar' buttons at the bottom.

En Productos podemos Crear, eliminar o editar cualquier producto:

Productos

[Registrar Nuevo Producto](#)

[Anterior](#) [Página 1 de 8](#) [Siguiente](#)

Buscar producto...

Todos los Géneros ▼ Todos los colores ▼ Todas las marcas ▼

ID	Nombre	Descripción	Imagen	Precio	Género	Color	Marca	Acciones
1	Nike Air Max Plus Drift	Deja que tu estilo se eleve con las Air Max Plus Drift, una experiencia Tuned Air actualizada que ofrece estabilidad y amortiguación premium. Con una malla ventilada, colores degradados y líneas de diseño onduladas originales, rinde homenaje a su estilo desafiante.		199.99 €	unisex	rojo	Nike	Editar Eliminar
2	Nike Air Max Plus Drift	Deja que tu estilo se eleve con las Air Max Plus Drift, una experiencia Tuned Air actualizada que ofrece estabilidad y amortiguación premium. Con una malla ventilada, colores degradados y líneas de diseño onduladas originales, rinde homenaje a su estilo desafiante.		199.99 €	unisex	negro	Nike	Editar Eliminar
3	Nike Air Max Plus Drift	Deja que tu estilo se eleve con las Air Max Plus Drift, una experiencia Tuned Air actualizada que ofrece estabilidad y amortiguación premium. Con una malla ventilada, colores degradados y líneas de diseño onduladas originales, rinde homenaje a su estilo desafiante.		199.99 €	unisex	negro	Nike	Editar Eliminar
4	Nike Air Max Plus Drift	Deja que tu estilo se eleve con las Air Max Plus Drift, una experiencia Tuned Air actualizada que ofrece estabilidad y amortiguación premium. Con una malla ventilada, colores degradados y líneas de diseño onduladas originales, rinde homenaje a su estilo desafiante.		199.99 €	unisex	marron	Nike	Editar Eliminar
5	Nike Air Max Plus Drift	Deja que tu estilo se eleve con las Air Max Plus Drift, una experiencia Tuned Air actualizada que ofrece estabilidad y amortiguación premium. Con una malla ventilada, colores degradados y líneas de diseño onduladas originales, rinde homenaje a su estilo desafiante.		199.99 €	unisex	gris	Nike	Editar Eliminar

[Anterior](#) [Página 1 de 8](#) [Siguiente](#)

En el formulario para registrar o editar un productos podemos añadir una nueva marca o un nuevo color.

Productos

[Registrar Nuevo Producto](#)

Registrar Nuevo Producto

Nombre del Producto:

Descripción del Producto:

Precio del Producto:

Género:

Color:

Marca:

Escribir nueva marca:

Escribir nuevo color:

[Utilizar Imagen](#) [Subir Imagen](#)

[Registrar producto](#) [Cancelar](#)

también podemos añadir una imagen por medio de url o png

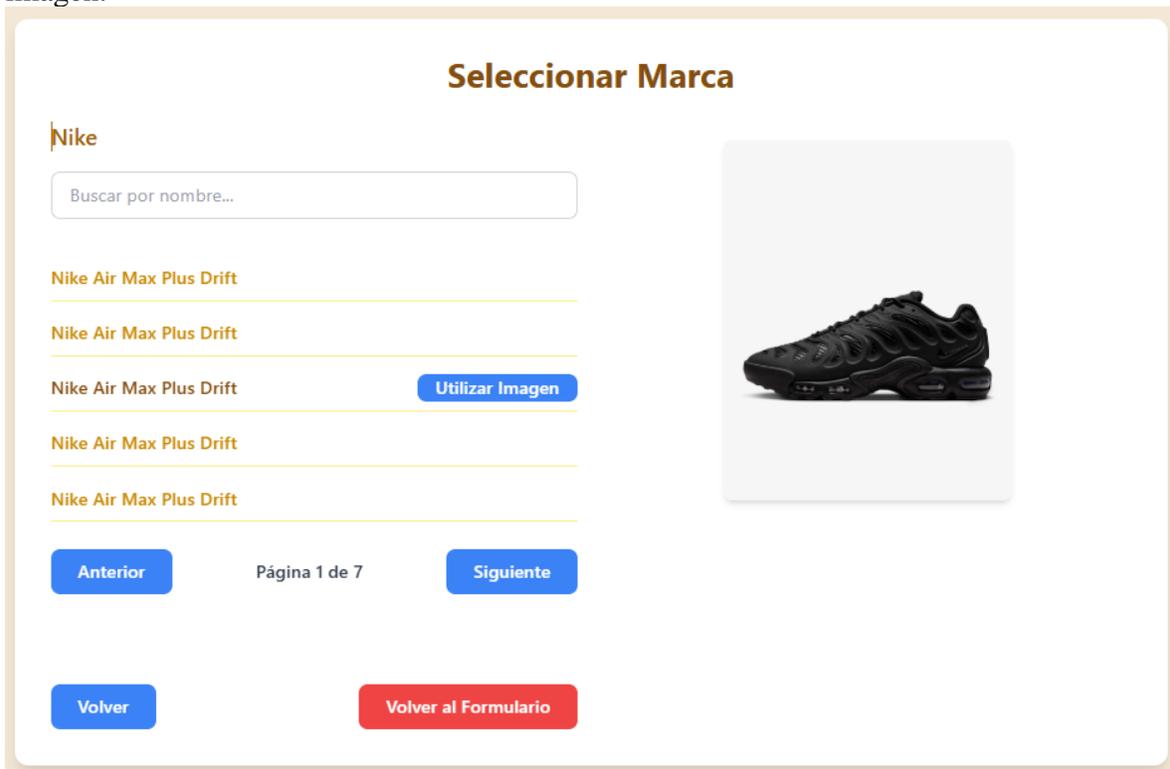
[Utilizar Imagen](#) [Subir Imagen](#)

[Usar URL de imagen](#) [Subir Imagen \(PNG sin fondo\)](#)

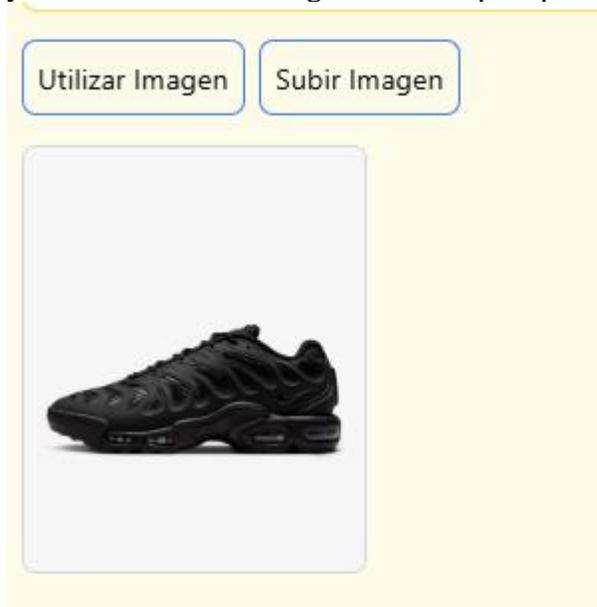
Esas imágenes se añaden al servidor de imagen. También podemos usar otras imágenes del servidor de imágenes dándole utilizar imagen



en el servidor de imágenes podemos seleccionar de que marca queremos seleccionar la imagen.



y dándole a utilizar imagen se añade para poder crear o editar un producto.



En la sección de recibos podemos ver los recibos de todos los usuarios:

<p>Usuario: Rodrigo Fecha: 15/12/2024 - Hora: 11:11:47</p> <p> Nike Air Force 1 07 LV8 Talla: 42 Precio: 119.99€ Cantidad: 1</p> <p> Nike Air Force 1 07 LV8 Talla: 44 Precio: 119.99€ Cantidad: 1</p> <p> PUMA x ONE PIECE Suede Luffy Sombrero de paja Talla: 45 Precio: 100€ Cantidad: 1</p> <p style="text-align: right;">Total: 339.98€</p>
<p>Usuario: Pedro Fecha: 15/12/2024 - Hora: 11:12:49</p> <p> Nike Air Max Plus Drift Talla: 41 Precio: 199.99€ Cantidad: 1</p> <p style="text-align: right;">Total: 199.99€</p>

Y por último tenemos la sección de stock

Stock
Registrar Movimiento

Anterior Página 1 de 4 Siguiente

Talla v

ID	Fecha	Talla	Cantidad	Movimiento	Acciones
1	12/15/24. 11:08 AM	36	8360	compra	Editar Eliminar
2	12/15/24. 11:08 AM	36	4565	compra	Editar Eliminar
3	12/15/24. 11:08 AM	36	1863	compra	Editar Eliminar
4	12/15/24. 11:08 AM	36	1961	compra	Editar Eliminar
5	12/15/24. 11:08 AM	36	1079	compra	Editar Eliminar
6	12/15/24. 11:08 AM	36	6552	compra	Editar Eliminar
7	12/15/24. 11:08 AM	36	1634	compra	Editar Eliminar
8	12/15/24. 11:08 AM	36	6226	compra	Editar Eliminar
9	12/15/24. 11:08 AM	36	5675	compra	Editar Eliminar
10	12/15/24. 11:08 AM	36	1038	compra	Editar Eliminar

Anterior Página 1 de 4 Siguiente

En la cual podemos editar, crear o eliminar cualquier stock de cualquier producto.

Stock
Ocultar Formulario

Registrar Nuevo Movimiento

Fecha

Producto ID

Talla

Cantidad

Movimiento

Registrar Movimiento Cancelar

Anterior Página 1 de 4 Siguiente

Stock
Registrar Movimiento

Editar Stock

Fecha

Producto ID

Talla

Cantidad

Movimiento

Guardar Cambios Cancelar