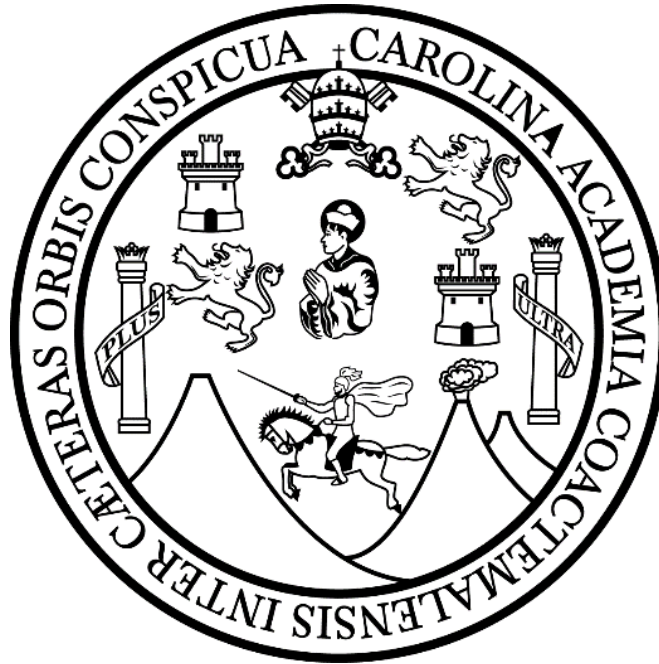


UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
CENTRO UNIVERSITARIO DE OCCIDENTE
DIVISIÓN DE CIENCIAS DE LA INGENIERÍA



DOCUMENTACIÓN: MANUAL TECNICO
ANÁLISIS Y DISEÑO 1
ING. DANIEL GONZALEZ

WILLIAM ALEXANDER MIRANDA SANTOS

201930967

LUIS FERNANDO SÁNCHEZ SANTOS

201930697

LUIS NERY CIFUENTES RODAS

202030482

MANUEL ANTONIO ROJAS PAXTOR

202030799

11 de septiembre de 2024

TABLA DE CONTENIDO

1. INTRODUCCIÓN.....	1
2. REQUERIMIENTOS.....	2
3. OBJETIVOS.....	3
Objetivo Principal.....	3
Objetivos Específicos.....	3
4. DESCRIPCIÓN DEL PROYECTO.....	4
5. ARQUITECTURA.....	4
5.1. TECNOLOGÍAS A UTILIZAR.....	4
6. FUNCIONALIDADES PRINCIPALES.....	4
6.1. VISITANTE.....	4
6.2. MÓDULO DEL ADMINISTRADOR.....	4
6.3. MÓDULO DEL AYUDANTE.....	4
6.4. MODULO DEL CLIENTE.....	4
7. DIAGRAMA DE COLECCIONES (MONGODB).....	4
8. DIAGRAMA DE CLASES.....	4
9. ESTRUCTURA DE LAS CARPETAS.....	4
10. ESTRUCTURA DE DESGLOSE DE TRABAJO.....	4
11. FLUJO DE TRABAJO CON TRELLO.....	4

1. INTRODUCCIÓN

El motivo principal del presente proyecto es crear una aplicación web para una tienda en línea que brinde una experiencia fácil de entender tanto para los clientes como para los administradores de la tienda. Los usuarios tendrán la capacidad de crear perfiles, administrar su información, elegir productos y elegir entre una variedad de opciones de entrega. Sin embargo, los administradores tendrán herramientas avanzadas para administrar productos, inventarios y pedidos, además de un sistema de roles que les permitirá distribuir tareas de manera eficiente.

2. REQUERIMIENTOS

A raíz de un arduo análisis del enunciado proporcionado, se han logrado identificar los siguientes requerimientos para el funcionamiento de la aplicación:

- *Registro y Gestión de Perfiles de Usuario:*
 - ☐ Los usuarios se registrarán proporcionando datos personales como nombre, correo electrónico, dirección, NIT y contraseña segura.
 - ☐ Se podrá actualizar información personal y cambiar contraseñas.
- *Carrito de Compras:*
 - ☐ Búsqueda y selección de productos.
 - ☐ Carrito que muestra productos seleccionados, precio y total a pagar.
- *Carrito de Compras:*
 - ☐ Búsqueda y selección de productos.
 - ☐ Carrito que muestra productos seleccionados, precio y total a pagar.
- *Opciones de Entrega del Producto:*
 - ☐ Los usuarios pueden optar por recogida en tienda física o entrega a domicilio (por un valor adicional).
- *Gestión de Productos:*
 - ☐ Los usuarios pueden optar por recogida en tienda física o entrega a domicilio (por un valor adicional).
- *Roles:*
 - ☐ Se manejarán 3 tipos de roles (Administrador, Ayudante y Cliente).
 - ☐ Administrador (Control sobre la aplicación)
 - Gestión de Productos
 - Inventario
 - Pedidos
 - Reportes
 - ☐ Ayudante (Acceso limitado a la aplicación)
 - Permisos para gestionar productos e inventario, pero sin acceso a configuraciones críticas.
- *Seguridad:*
 - ☐ Encriptación de contraseñas en la base de datos.
 - ☐ Recuperación segura de contraseñas.

3. OBJETIVOS

Objetivo Principal

Desarrollar una aplicación web para una tienda en línea que ofrezca una experiencia de compra intuitiva, segura y flexible para los usuarios, permitiendo la gestión eficiente de productos, inventarios y pedidos, con funcionalidades avanzadas de administración, roles de usuario y generación de reportes, asegurando la escalabilidad y seguridad de la plataforma.

Objetivos Específicos

- *Implementación del Sistema de Registro y Perfil de Usuario.*
- *Desarrollar una funcionalidad de carrito de compras*
- *Diseñar un módulo de gestión de productos*
- *Crear un sistema de roles y permisos*
- *Desarrollar un dashboard administrativo y un sistema de reportes personalizados*
- *Garantizar la seguridad de la plataforma*

4. DESCRIPCIÓN DEL PROYECTO

La finalidad del proyecto es desarrollar un *eComerce* altamente personalizable que brinde una solución completa y flexible para administradores y usuarios finales. Esta plataforma está destinada a satisfacer las necesidades de los negocios de comercio electrónico que buscan administrar sus ventas en línea de manera segura, eficiente y adaptada a su propia marca y estilo.

La aplicación permite a los administradores personalizar el nombre y el logotipo de la tienda y ajustar los detalles de facturación. Además, la plataforma incluye un sistema de gestión de roles con permisos granulares que permite la creación de varios perfiles de usuario, como el de "Ayudantes", que pueden realizar tareas específicas asignadas por el administrador, como la gestión de productos o la revisión de pedidos, sin acceso a las configuraciones críticas de la tienda.

A continuación se describen las funcionalidades más importantes de la aplicación:

- *Gestión de usuarios*: Los clientes podrán registrarse, crear perfiles, actualizar sus datos y administrar sus pedidos de manera segura, con contraseñas encriptadas y recuperación de contraseñas a través de correo electrónico.
- *Carrito de compras*: Las opciones de entrega permiten que los clientes naveguen por el catálogo de productos, los agreguen a su carrito y elijan entre recoger sus compras en la tienda física o pagar contra entrega.
- *Gestión de productos e inventario*: Los administradores y ayudantes podrán administrar el catálogo de productos, organizarlos en categorías y recibir alertas cuando el inventario de un producto esté bajo.
- *Dashboard y reportes*: la aplicación proporcionará un tablero administrativo con métricas importantes de ventas y productos, así como la opción de generar reportes personalizados.

5. ARQUITECTURA

El proyecto se construyó utilizando una arquitectura full stack basada en JavaScript, con el backend desarrollado en Node.js y Express, almacenamiento de imágenes en un bucket de AWS S3, la base de datos alojada en MongoDB y un frontend interactivo desarrollado con Angular Material.

Backend - Node.js y Express

El backend está desarrollado sobre Node.js, una plataforma ligera y eficiente basada en el motor V8 de Chrome, que permite manejar peticiones concurrentes de forma efectiva. El framework Express.js se utiliza para la creación de APIs RESTful que manejan las operaciones de negocio principales. La arquitectura es modular, con controladores dedicados para gestionar los recursos clave, como usuarios, productos, pedidos y tarjetas de crédito.

- Rutas y Controladores: Se implementaron rutas para las distintas entidades de la aplicación (usuarios, productos, órdenes, tarjetas de crédito), siguiendo una estructura modular que permite el mantenimiento y escalabilidad del proyecto.
- Middleware de Autenticación: Se utiliza JWT (JSON Web Token) para la autenticación de usuarios, asegurando que solo los usuarios autenticados puedan acceder a ciertas rutas y recursos.
- Verificación de Permisos: El sistema de roles y permisos se gestiona mediante middlewares que verifican si un usuario tiene acceso a ciertas funcionalidades o recursos.

Base de Datos - MongoDB

Se utiliza un cluster de MongoDB como base de datos NoSQL para almacenar la información del proyecto, lo cual proporciona alta escalabilidad y flexibilidad para manejar datos estructurados y no estructurados.

- Esquemas y Colecciones: Los datos se organizan en colecciones como Users, Products, Orders y CreditCards. Los esquemas de MongoDB permiten definir la estructura de los documentos y realizar validaciones de los datos.
- Transacciones: Para operaciones críticas (como la creación de órdenes y actualización de stock), se utilizan transacciones de MongoDB que garantizan la integridad de los datos.

Almacenamiento de Imágenes - AWS S3

Las imágenes subidas por los usuarios, como las de productos o fotos de perfil, se almacenan en un bucket de AWS S3. Esto proporciona un almacenamiento escalable, seguro y de bajo costo para los archivos estáticos.

- Integración con AWS SDK: El backend se integra con AWS mediante el SDK para gestionar el upload de imágenes de manera segura y eficiente.
- Enlace de Imágenes: Los enlaces generados para las imágenes almacenadas en S3 se guardan en MongoDB junto con los documentos correspondientes, permitiendo un acceso rápido y directo desde el frontend.

Frontend - Angular Material

El frontend está desarrollado con Angular utilizando Angular Material para proporcionar una interfaz de usuario moderna y responsiva. Angular Material provee componentes estilizados que facilitan la creación de un diseño consistente y atractivo.

- Componentes y Servicios: El frontend está organizado en componentes que manejan diferentes vistas y funcionalidades de la aplicación. Los servicios de Angular consumen las APIs del backend para interactuar con los datos.
- Gestión del Estado: Se utiliza RxJS para gestionar flujos de datos y mantener el estado de la aplicación a través de servicios observables, lo que permite una mejor reactividad en la interfaz.
- Seguridad: El frontend maneja tokens de autenticación JWT para autenticar y autorizar usuarios en cada solicitud al backend.

Flujo General de la Aplicación

- Autenticación: Los usuarios se autentican mediante JWT; al iniciar sesión, reciben un token que se utiliza para proteger las rutas del backend.
- Operaciones CRUD: Los usuarios pueden realizar operaciones CRUD (Crear, Leer, Actualizar y Eliminar) sobre productos, pedidos, y otros recursos, dependiendo de sus permisos y roles.
- Gestión de Imágenes: Las imágenes de productos se suben al bucket de AWS S3 y luego se almacenan sus enlaces en MongoDB.
- Permisos y Roles: Los roles como Admin, Assistant o Customer definen qué funcionalidades puede acceder cada usuario, con verificaciones tanto en el frontend como en el backend.

6. FUNCIONALIDADES PRINCIPALES

6.1. VISITANTE

El visitante es cualquier usuario que accede a la aplicación sin estar registrado o autenticado. Sus funcionalidades están limitadas principalmente a la exploración de la tienda.

Funcionalidades:

- *Registrarse:* El visitante puede crear una cuenta proporcionando su información personal (nombre, correo electrónico, dirección, etc.) y establecer una contraseña segura.
- *Iniciar sesión:* Si el visitante ya está registrado, puede autenticarse en el sistema para acceder a más funcionalidades.
- *Ver productos:* Puede navegar por el catálogo completo de productos, viendo detalles como precios, descripciones e imágenes.
- *Buscar y filtrar productos:* El visitante puede utilizar herramientas de búsqueda y filtros (categoría, precio, disponibilidad) para encontrar productos de manera más eficiente.

6.2. MÓDULO DEL ADMINISTRADOR

El administrador tiene acceso completo a todas las funcionalidades de la aplicación, siendo responsable de la gestión global de la tienda, incluyendo usuarios, productos, categorías y pedidos.

Funcionalidades:

- *Control de usuarios:* El administrador puede gestionar las cuentas de los usuarios, incluyendo la creación de nuevos usuarios (ayudantes o administradores), la edición de permisos, la suspensión de cuentas o la eliminación de usuarios.
- *Generación de reportes:* El administrador tiene acceso a un panel avanzado de reportes donde puede generar y exportar informes sobre ventas, inventarios, pedidos y rendimiento de productos.
- *Control de pedidos:* Puede revisar todos los pedidos realizados en la tienda, actualizar estados (en proceso, enviado, cancelado), y gestionar devoluciones o problemas con las entregas.
- *Control de productos:* Tiene la capacidad de crear, editar, eliminar y organizar productos dentro del sistema, gestionando todos los detalles asociados a ellos (descripción, precio, disponibilidad, imágenes, etc.).

- *Control de categorías:* El administrador puede crear y gestionar categorías, organizando los productos de manera que los clientes puedan encontrarlos fácilmente.
- *Asignación de permisos:* El administrador puede crear perfiles de ayudante y definir qué funcionalidades pueden o no realizar, controlando sus niveles de acceso en el sistema.

6.3. MÓDULO DEL AYUDANTE

El ayudante es un usuario que tiene acceso a funcionalidades específicas dentro del sistema, dependiendo de los permisos que le haya otorgado el administrador. Este rol es útil para delegar tareas operativas sin conceder acceso a configuraciones críticas.

Funcionalidades:

- *Gestión de productos:* Dependiendo de los permisos, el ayudante puede crear, editar o eliminar productos dentro del sistema. También puede agregar descripciones, imágenes, precios y gestionar el inventario.
- *Gestión de categorías:* El ayudante puede crear y organizar las categorías de productos, permitiendo una mejor clasificación para facilitar la navegación del cliente.
- *Gestión de pedidos:* Puede acceder al módulo de pedidos para visualizar, procesar, actualizar estados y gestionar los pedidos recibidos por la tienda.

Tomar en Cuenta: Los permisos del ayudante son configurados por el administrador, es decir, puede tener acceso a una o varias funcionalidades dependiendo del administrador.

6.4. MODULO DEL CLIENTE

El cliente es un usuario que ha completado el proceso de registro y ha iniciado sesión. Tiene acceso a una variedad de funcionalidades orientadas a la gestión de su perfil y la realización de compras.

Funcionalidades:

- *Ver y editar perfil:* El cliente puede acceder a su perfil, modificar su información personal, como nombre, dirección, correo electrónico y preferencias de pago.
- *Administrar direcciones:* Puede agregar, eliminar y modificar direcciones para facturación y entrega de productos.
- *Administrar tarjetas de crédito:* Tiene la opción de agregar, eliminar y gestionar las tarjetas de crédito que utilizará para sus compras.
- *Generar pedidos/compras:* El cliente puede añadir productos al carrito de compras, revisar el total a pagar y completar el proceso de compra. También puede elegir la opción de recoger en tienda o entrega a domicilio, con la posibilidad de pago contra entrega.
- *Control de pedidos:* Los clientes pueden acceder a un historial de pedidos realizados, ver el estado actual de sus compras (pendiente, enviado, entregado, cancelado) y solicitar devoluciones o aclaraciones si es necesario.

Distribución de Roles y Funcionalidades:

Funcionalidad	Visitante	Cliente	Ayudante	Administrador
Registrarse	✓			
Iniciar sesión	✓			
Ver productos	✓	✓	✓	✓
Buscar/filtrar productos	✓	✓	✓	✓
Ver/editar perfil		✓		

Administrar direcciones		✓		
Administrar tarjetas		✓		
Generar pedidos/compras		✓		
Control de pedidos (personales)		✓		
Gestión de productos			✓	✓
Gestión de categorías			✓	✓
Gestión de pedidos (global)			✓	✓
Control de usuarios				✓
Generar reportes				✓
Asignación de permisos				✓

7. CASOS DE USO

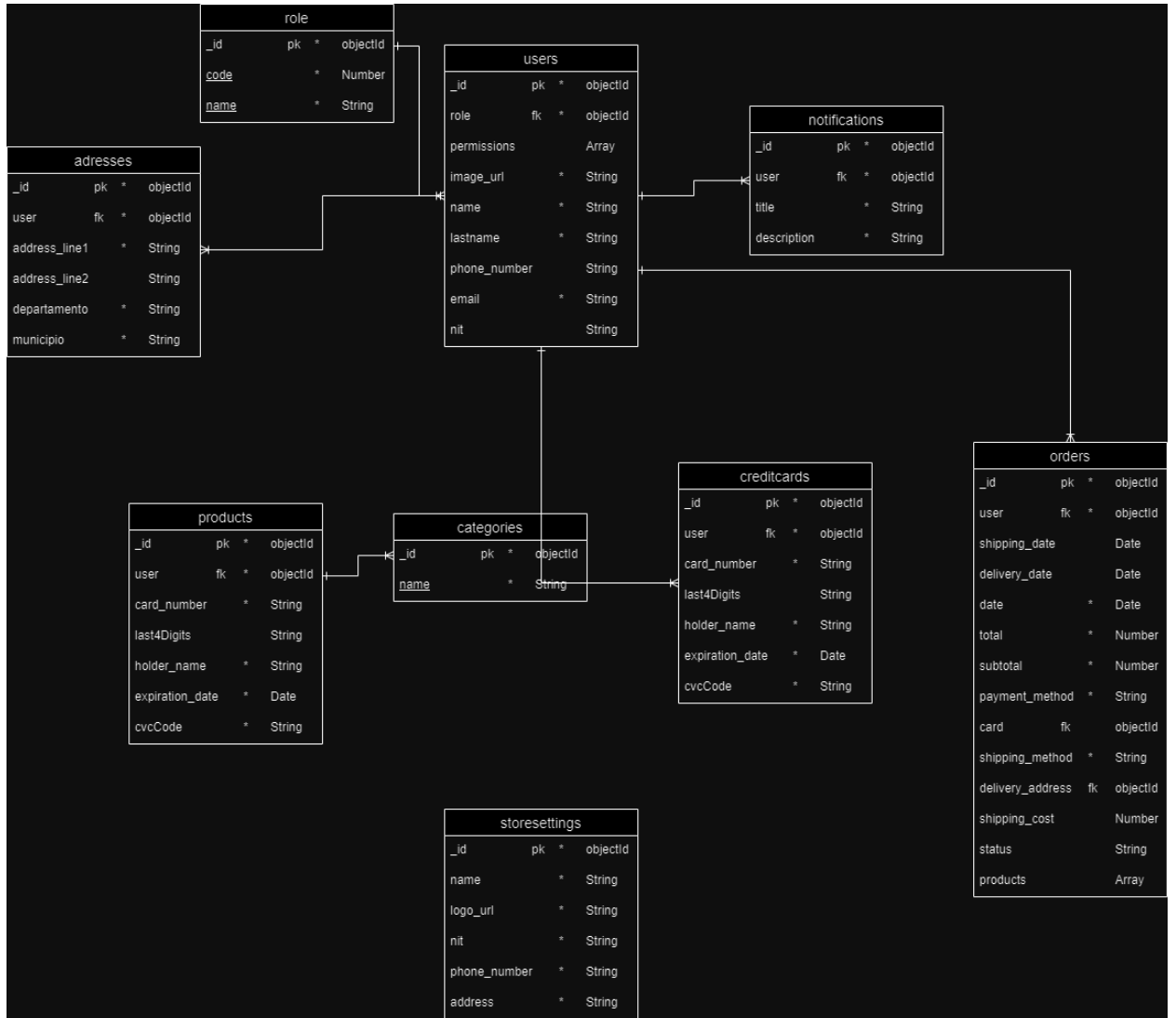
Iniciar sesión

Caso de uso	Iniciar sesión
Actor principal	Visitante
Personal involucrado e intereses	Usuario que desea acceder a su cuenta y realizar acciones como usuario registrado. Interesado en iniciar sesión para acceder a las funcionalidades exclusivas para usuarios registrados.
Precondiciones	El visitante tiene una cuenta registrada en la plataforma.
Postcondiciones	El visitante ha iniciado sesión con éxito y puede acceder a su perfil de usuario y otras funcionalidades exclusivas para usuarios registrados.
Curso principal	<ol style="list-style-type: none">1. El visitante accede a la página de inicio de sesión.2. El sistema presenta el formulario de inicio de sesión.3. El visitante completa el formulario con su nombre de usuario y contraseña.4. El visitante envía el formulario.5. El sistema verifica las credenciales proporcionadas.6. El sistema autentica al visitante y redirige a la página principal como usuario registrado.
Curso alternativo	<p>5.1 Si el nombre de usuario o la contraseña proporcionados son incorrectos, el sistema muestra un mensaje de error y solicita al visitante que vuelva a intentarlo.</p> <p>5.2 Si es recurrente puede reiniciar su contraseña.</p>
Requisitos especiales	El sistema debe contar con un mecanismo de autenticación seguro para proteger las credenciales de inicio de sesión del usuario.
Lista de tecnología y variaciones de datos	Tecnología: Laravel, React, MariaDB Variaciones de Datos: usuario, contraseña.
Frecuencia	Cada vez que un usuario registrado desea acceder a su cuenta en la plataforma.
Temas abiertos	<ul style="list-style-type: none">- Seguridad: Garantizar que el proceso de inicio de sesión sea seguro para proteger la información del usuario.- Usabilidad: Asegurar que el formulario de inicio de sesión sea fácil de usar y accesible para los usuarios.

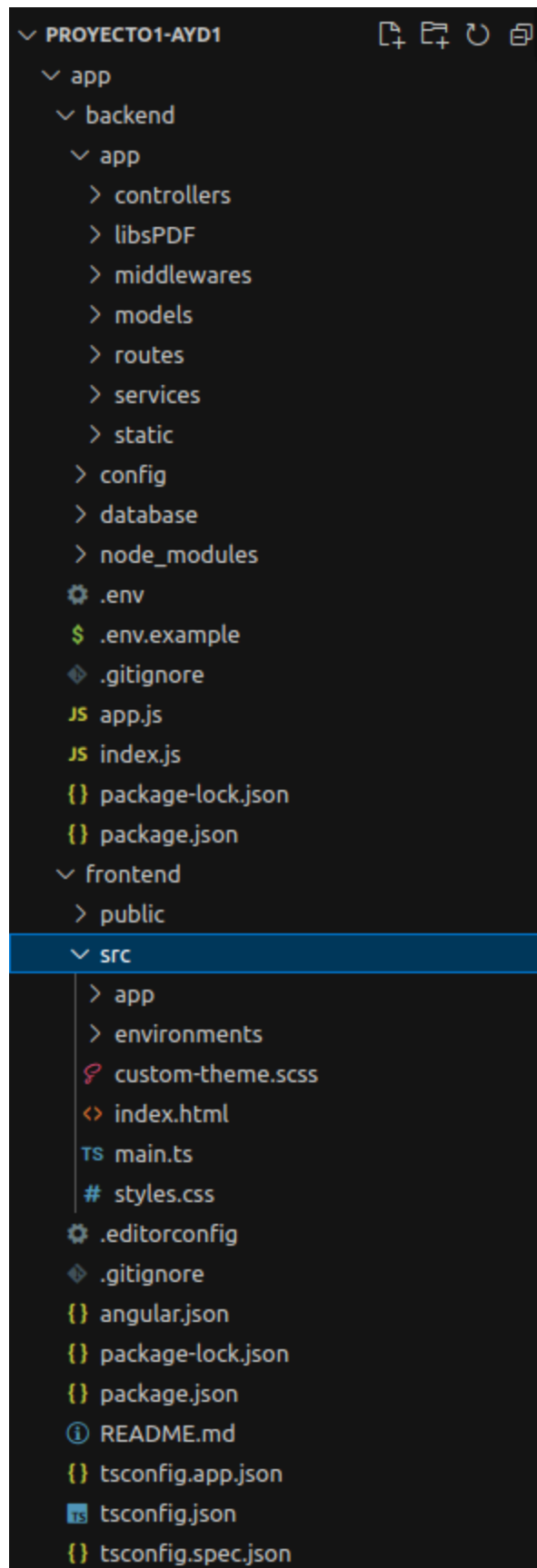
Registrarse

Caso de uso	Registrarse
Actor principal	Visitante
Personal involucrado e intereses	Usuario que desea acceder a características adicionales de la plataforma. Interesado en registrarse para poder realizar compras o utilizar otras funcionalidades exclusivas para usuarios registrados.
Precondiciones	El visitante no tiene una cuenta activa en la plataforma.
Postcondiciones	El visitante ha creado una cuenta y puede acceder a su perfil de usuario y otras funcionalidades exclusivas para usuarios registrados.
Curso principal	<ol style="list-style-type: none">1. El visitante accede a la página de registro.2. El sistema presenta el formulario de registro.3. El visitante completa el formulario con su información personal.4. El visitante envía el formulario.5. El sistema verifica la información proporcionada.6. El sistema crea una cuenta para el visitante.7. El sistema redirige al visitante a la página de inicio de sesión.
Curso alternativo	<p>5.1 Si la dirección de correo electrónico proporcionada ya está en uso, el sistema muestra un mensaje de error y solicita al visitante que proporcione una dirección de correo electrónico diferente.</p> <p>5.2 Si el nombre de usuario proporcionado ya está en uso, el sistema muestra un mensaje de error y solicita al visitante que elija un nombre de usuario diferente.</p>
Requisitos especiales	El sistema debe contar con un mecanismo para verificar la autenticidad de la dirección de correo electrónico proporcionada por el visitante.
Lista de tecnología y variaciones de datos	Tecnología: Laravel, React, MariaDB Variaciones de Datos: Nombre, usuario, contraseña, correo electrónico.
Frecuencia	Cada vez que un visitante desea convertirse en usuario registrado de la plataforma.
Temas abiertos	<ul style="list-style-type: none">- Seguridad: Garantizar que la información personal del usuario esté protegida durante el proceso de registro.- Usabilidad: Asegurar que el formulario de registro sea intuitivo y fácil de completar para los visitantes.

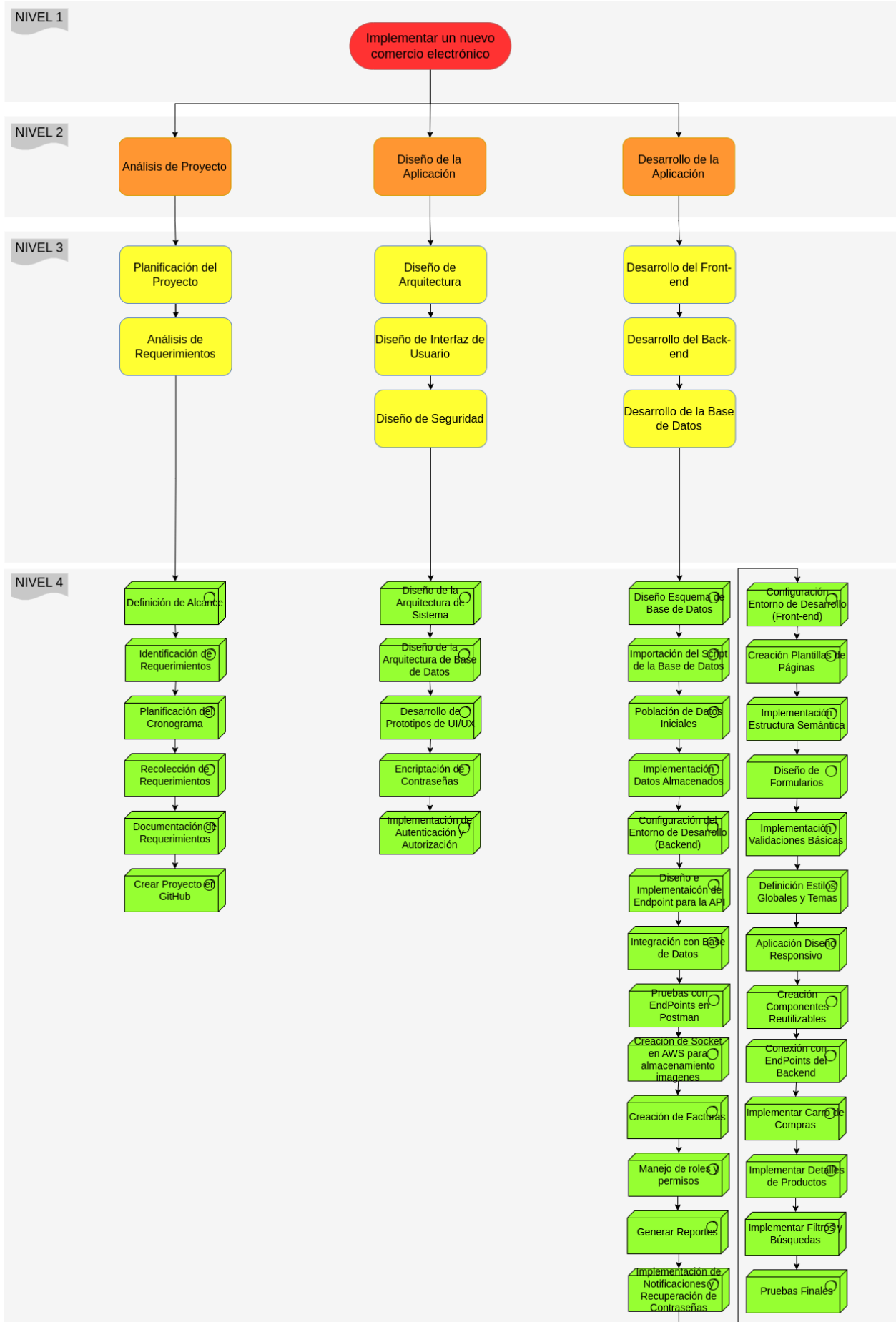
8. DIAGRAMA DE COLECCIONES (MONGODB)



9. ESTRUCTURA DE LAS CARPETAS



10. ESTRUCTURA DE DESGLOSE DE TRABAJO



11. FLUJO DE TRABAJO CON TRELLO

