

Centro Universitario de Occidente

Análisis y Diseño 1

Ing. Daniel González

Segundo Semestre 2024



“Proyecto 1 AyD E-Commerce”

201930693 - Leví Isaac Hernández Sapón
201930699 - Erick Daniel Morales Xicarà
201832069 - Marcos Andrés Aguiar Bravo
201830498 - Oscar Antonio de León Urizar

Índice

Abstracción de Información	3
Introducción	5
Objetivos	6
Marco Teórico	7
Facilidad de Uso y Navegación:	11
EDT	12
Diccionario del Proyecto	17
Primera Entrega (40% del Proyecto):	18
Segunda Entrega (80% del Proyecto):	19
Tercera Entrega (100% del Proyecto):	20
Diagramas de la solución	21
Casos de uso (Diagramas)	28
Diagramas de Secuencia	32
Diagramas de Actividades	34
Organigrama	37
Diagrama de Tablas	38
Diagrama de Despliegue	39
Diagrama de Paquetes	40
Diagrama de Estructura	41
Diagrama de Comunicación	42
Cronograma	42
Estructura de Carpetas	43
Esquema de Arquitectura	44
Definición de Actividades	46
Secuencia de Actividades	47
Fase de Inicio:	47
Fase de Diseño:	48
Fase de Ejecución:	49
Fase de Pruebas:	49
Fase de Reportes:	50
Fase de Publicidad y Mantenimiento:	50
Fase de Finalización:	50
Fechas Importantes	51
Elaboración del Acta de Constitución del Proyecto:	51
Primer Entregable (40% del Proyecto):	51
Segundo Entregable (80% del Proyecto):	52
Tercer Entregable (100% del Proyecto):	52
Tareas y Fechas por Semana	52
Conclusión	54
Preguntas	55
¿Cuál es el problema?	55
¿Cuál es la solución?	55
¿Quién?	55
¿Dónde?	55
¿Cómo?	56
¿Cuándo?	56
¿Cuánto costará?	56
¿Con qué se llevará a cabo la solución?	56
¿Cuál es el alcance y bajo qué condiciones?	56
¿Dónde se implementará la solución?	57
¿Quiénes están involucrados?	57

Abstracción de Información

Problema: Las empresas que venden productos exclusivamente propios enfrentan el desafío de administrar las ventas y mantener una relación cercana con los clientes. Sin una plataforma eficiente y organizada, puede ser difícil ofrecer una experiencia de compra en línea segura y accesible, lo que limita el alcance y las oportunidades de crecimiento comercial.

Solución Propuesta: Desarrollaremos una plataforma de **E-commerce** exclusiva para la venta de productos de la empresa. Los usuarios podrán registrarse, crear un perfil, y acceder al catálogo de productos para realizar compras. El sistema utilizará **Angular** para el frontend, ofreciendo una interfaz de usuario dinámica y responsiva, mientras que el backend será gestionado mediante **Express**, garantizando un manejo eficiente de la lógica del servidor y la base de datos. La plataforma ofrecerá un entorno seguro para las transacciones, utilizando prácticas modernas de autenticación y seguridad.

Características Clave:

- **Cuentas de Usuario:** Los usuarios deberán registrarse y crear un perfil para acceder a la plataforma. Desde su perfil, podrán gestionar su información personal, ver su historial de compras y realizar transacciones de manera segura.
- **Catálogo de Productos de la Empresa:** El sistema mostrará un catálogo completo de los productos que ofrece la empresa, permitiendo a los usuarios navegar por las categorías, ver detalles de productos y añadirlos a su carrito de compras.
- **Carrito de Compras y Opciones de Entrega:** Los usuarios podrán gestionar sus productos en un carrito de compras virtual, revisar los detalles del pedido, seleccionar

la cantidad y proceder con el pago. Además, tendrán la opción de elegir entre la recogida en tienda física o la entrega a domicilio.

- **Seguridad y Gestión de Usuarios:** La plataforma garantizará la seguridad de las transacciones a través de encriptación de contraseñas y autenticación de usuarios. Los datos personales y las compras estarán protegidos bajo estándares de seguridad moderna.
- **Gestión de Productos y Pedidos:** Los administradores de la empresa podrán gestionar el inventario, revisar y procesar pedidos, así como generar reportes de ventas. Además, podrán actualizar y organizar el catálogo de productos desde el backend.

Beneficios Esperados:

- **Mejora en la Gestión de Ventas:** La empresa centralizará sus ventas en una plataforma eficiente, mejorando la experiencia del cliente y optimizando el proceso de compra.
- **Aumento de la Satisfacción del Cliente:** Los usuarios podrán navegar fácilmente por los productos y realizar compras de manera sencilla y segura, lo que aumentará la confianza y la fidelidad hacia la empresa.
- **Seguridad y Transparencia:** Todas las transacciones y datos estarán protegidos mediante tecnologías de seguridad de última generación, proporcionando una experiencia de compra confiable y transparente.
- **Escalabilidad y Crecimiento:** La plataforma permitirá a la empresa expandir su presencia en línea, gestionando un mayor volumen de usuarios y pedidos sin comprometer la eficiencia o la seguridad.

Introducción

En un entorno empresarial competitivo, las empresas necesitan optimizar sus canales de venta y mejorar la experiencia del cliente para mantenerse relevantes en el mercado. La creación de plataformas de E-commerce especializadas juega un papel clave en este proceso, permitiendo una gestión eficiente de las ventas y facilitando el acceso a los productos por parte de los consumidores. Sin embargo, muchas empresas aún enfrentan desafíos para ofrecer una plataforma que cumpla con estos objetivos de manera organizada y segura.

Para abordar esta necesidad, proponemos el desarrollo de una plataforma de E-commerce exclusiva para la venta de productos de la empresa, que facilitará las transacciones comerciales en un entorno seguro, eficiente y transparente. Los usuarios podrán registrarse, crear un perfil y realizar compras de productos ofrecidos únicamente por la empresa, centralizando la gestión de ventas en un único sistema.

La plataforma se desarrollará utilizando una arquitectura basada en Angular para el frontend, proporcionando una interfaz de usuario dinámica y responsiva, y Express para el backend, que se estructurará mediante microservicios. El uso de microservicios en Express permite dividir la lógica de negocio en servicios independientes, lo que facilita la escalabilidad, el mantenimiento y el despliegue de la plataforma. Además, esta arquitectura modular permite que cada servicio sea desarrollado y actualizado de forma autónoma, reduciendo el tiempo de inactividad y mejorando la capacidad de respuesta del sistema a cambios o actualizaciones.

Con el fin de asegurar la entrega continua de valor y la satisfacción del cliente, se adoptarán prácticas ágiles en el desarrollo del proyecto, asegurando que el sistema sea flexible y capaz de responder rápidamente a las necesidades de los usuarios y los administradores.

Objetivos

Título del Proyecto:

Desarrollo de Plataforma de Comercio Electrónico para Empresas

Objetivo General:

Desarrollar e implementar una plataforma de comercio electrónico genérica para **Error Encoders**, que permita a las empresas gestionar de manera eficiente la venta de sus productos, proporcionando una experiencia de usuario segura y personalizable.

Objetivos Específicos:

- Facilitar la compra y venta de productos de la empresa mediante una plataforma digital eficiente, con un enfoque en la seguridad y la simplicidad para el usuario final.
- Implementar un sistema de gestión de usuarios con roles diferenciados (administradores y compradores) que permita una administración eficaz de los productos, transacciones y clientes.
- Garantizar la seguridad de las transacciones mediante el uso de cifrado y autenticación robusta, asegurando la integridad de los datos y la confianza del usuario.
- Desarrollar una arquitectura modular que permita una fácil escalabilidad y personalización para que otras empresas puedan adaptar la plataforma según sus necesidades específicas.

Marco Teórico

Concepto de Actividad Económica:

La actividad económica se refiere a cualquier proceso que implica la producción, distribución y consumo de bienes y servicios para satisfacer las necesidades de los individuos. Las actividades económicas abarcan desde la manufactura y el comercio hasta la prestación de servicios especializados, constituyendo una base fundamental para el desarrollo de las sociedades modernas.

El Comercio como Actividad Económica:

El comercio es una actividad económica clave que consiste en el intercambio de bienes y servicios entre diferentes agentes económicos. Este intercambio puede realizarse tanto a nivel local como internacional y es fundamental para el crecimiento económico y la generación de valor dentro de las comunidades. En la era digital, el comercio ha evolucionado significativamente, dando lugar a nuevas formas de intercambio a través de plataformas en línea.

Evolución de las Formas de Intercambio:

Trueque: Fue una de las primeras formas de comercio, donde las personas intercambiaban bienes o servicios de manera directa. Aunque era eficiente en sociedades pequeñas, resultó ser insuficiente a medida que las economías crecían en complejidad.

Moneda: La creación de la moneda facilitó el comercio al introducir un medio común de intercambio. Permitió establecer precios y valores para los bienes, simplificando así las transacciones entre los participantes.

Comercio Electrónico: En la actualidad, el comercio electrónico ha transformado la forma en que se realizan los intercambios económicos. Las plataformas digitales permiten a las empresas y consumidores comprar y vender productos y servicios a través de internet de manera rápida y segura.

Sistemas de Comercio Electrónico:

El comercio electrónico implica la compra y venta de bienes y servicios a través de plataformas en línea. En este tipo de sistemas, la seguridad, la eficiencia y la confianza son esenciales para garantizar el éxito de las transacciones. En un contexto empresarial, las plataformas de E-commerce permiten a las compañías gestionar sus ventas de manera centralizada, mejorar la experiencia del cliente y escalar sus operaciones a nivel global.

Objetivo del Proyecto y Contenido de la Aplicación Web:

El proyecto tiene como objetivo desarrollar una plataforma de comercio electrónico genérica para Error Encoders, diseñada para ser implementada por cualquier empresa que desee un sistema robusto y flexible para gestionar la venta de productos propios. La aplicación permitirá a los usuarios crear perfiles, navegar por un catálogo de productos, agregar artículos al carrito y realizar compras. Además, ofrecerá una arquitectura modular que permitirá futuras personalizaciones según las necesidades de cada empresa.

Desarrollo Frontend (Angular):

Diseñar una interfaz de usuario intuitiva, adaptable y responsiva que permita a los usuarios navegar fácilmente por el catálogo de productos.

Implementar componentes clave para la gestión de cuentas de usuario, el carrito de compras y el proceso de pago.

Garantizar la compatibilidad con múltiples dispositivos, facilitando el acceso a la plataforma desde computadoras y móviles.

Desarrollo Backend (Express con Microservicios):

Establecer una arquitectura basada en microservicios para garantizar la escalabilidad y modularidad del sistema.

Implementar lógica de negocio que gestione las transacciones de compra, la administración de usuarios y el inventario de productos.

Crear APIs seguras y eficientes para la comunicación entre el frontend y el backend, mejorando la interacción en tiempo real entre las diferentes partes del sistema.

Base de Datos (MariaDB):

Diseñar una estructura de base de datos optimizada para almacenar información de productos, usuarios y transacciones.

Implementar consultas eficientes que mejoren el rendimiento general de la aplicación y aseguren una experiencia de usuario rápida y fluida.

Compra y Venta:

Implementar funciones que permitan la publicación de productos, su búsqueda y compra.

Integrar una pasarela de pago segura para la realización de transacciones con moneda local.

Administración y Seguridad:

Crear un panel de administración que permita gestionar usuarios, productos, inventarios y transacciones.

Implementar medidas de seguridad como la encriptación de datos, autenticación de usuarios y protección contra ataques cibernéticos.

Documentación y Soporte:

Proporcionar manuales de usuario y documentación técnica detallada.

Ofrecer soporte técnico eficiente para resolver cualquier problema que pueda surgir durante el uso de la plataforma.

Herramientas a Utilizar:

- **Frontend:** Se utilizará **Angular**, un framework mantenido por Google, para desarrollar la parte visible de la plataforma. Angular permite una experiencia de usuario interactiva y responsiva, gracias a características como la arquitectura MVVM, el enrutamiento integrado y el tipado estático mediante TypeScript.

- **Instalación**

```
sudo apt install nodejs npm  
npm install -g @angular/cli
```

- **Backend:** Para el backend, se empleará **Express** con una arquitectura basada en microservicios. Esta estructura facilita la modularidad del sistema y permite escalar las funcionalidades de manera eficiente.

- **Instalación:**

```
npm install express
```

Base de Datos: Se utilizará MariaDB para gestionar la base de datos:

```
sudo apt install mysql-server
```

Facilidad de Uso y Navegación:

Para garantizar que los usuarios de la plataforma de **E-commerce** puedan encontrar y adquirir productos de manera eficiente, es esencial que la interfaz sea intuitiva y de fácil navegación.

Algunas de las estrategias clave implementadas en el diseño incluyen:

- **Diseño Intuitivo:** La plataforma ha sido diseñada pensando en la simplicidad y claridad. Los elementos de navegación, como el menú principal y las categorías de productos, están claramente visibles, permitiendo a los usuarios acceder fácilmente a cualquier sección de la tienda sin complicaciones.
- **Organización Coherente de la Información:** Cada sección del sitio está organizada de manera lógica, con subtítulos claros y categorías de productos bien definidas. Los usuarios podrán navegar por el catálogo de productos y encontrar información relevante de manera rápida y sencilla.
- **Herramienta de Búsqueda:** Para mejorar la experiencia del usuario, se ha incorporado una herramienta de búsqueda avanzada, que permitirá a los clientes encontrar productos específicos ingresando palabras clave, lo que reduce el tiempo de búsqueda y facilita la compra.
- **Diseño Responsivo:** La plataforma ha sido desarrollada para ser completamente accesible desde cualquier dispositivo, ya sea una computadora de escritorio, laptop, tablet o teléfono móvil. Esto asegura que la experiencia de usuario sea consistente y fluida sin importar el tamaño de la pantalla.
- **Pruebas de Usabilidad:** Para garantizar que la plataforma sea fácil de usar, se realizarán pruebas de usabilidad con diversos perfiles de usuarios. Esto permitirá identificar posibles áreas de mejora y asegurar una experiencia de usuario óptima en todas las etapas del proceso de compra.

EDT

Plataforma de Comercio Electrónico de Error Encoders

El **EDT** (Estructura de Desglose del Trabajo) nos permite dividir y organizar las tareas del proyecto de manera jerárquica, identificando dependencias y facilitando la planificación. Dado que el software desarrollado para **Error Encoders** es genérico, orientado a cualquier empresa que desee implementar un sistema de comercio electrónico, el EDT está diseñado para reflejar los diferentes hitos del desarrollo.

Primer Nivel - Meta del Proyecto: Desarrollar un software de comercio electrónico escalable, flexible y seguro que permita a las empresas gestionar sus ventas y mejorar su interacción con los clientes, facilitando una experiencia de compra optimizada.

Segundo Nivel - Objetivos Clave del Proyecto:

- **Entrega del 40% del proyecto:** Implementación inicial de la plataforma, configuración básica del frontend y backend.
- **Entrega del 80% del proyecto:** Desarrollo completo de funcionalidades principales, incluyendo gestión de productos, usuarios y transacciones.
- **Entrega del 100% del proyecto:** Pruebas generales, manuales de usuario y técnico, despliegue y publicidad de la plataforma.

Presupuesto estimado: Q100,000 - Q500,000, dependiendo de la complejidad de personalización y los módulos adicionales requeridos.

El **EDT** está acompañado por un diagrama visual que muestra cómo se desglosan las tareas, tomando en cuenta:

- **Diccionario de la EDT**
- **Descripción de la tarea**
- **Responsable de la tarea**
- **Presupuesto de la tarea**
- **Fecha de finalización**
- **Estado de la tarea**



Descripción de las Tareas:

Entrega del 40% (Etapas Iniciales)

- **Definición del entorno de desarrollo (Terminada):** Configuración del entorno utilizando **Angular** para el frontend, **Express** con microservicios para el backend, y **Maria DB** para la base de datos.
- **Creación de diagramas de abstracción de información (Terminada):** Incluye el uso de diagramas UML, como el Diagrama de Clases y el Diagrama de Base de Datos, para modelar las relaciones entre las entidades del sistema.
- **Toma de Requerimientos (Terminada):** Recolección de información de los stakeholders para definir funcionalidades y expectativas clave del sistema.
- **Documentación de instalación de tecnologías (Terminada):** Creación de una guía paso a paso para configurar el entorno de desarrollo con las tecnologías seleccionadas.
- **Creación de Repositorio en GitHub (Terminada):** Configuración de un repositorio para la gestión de código fuente.
- **Creación del Proyecto en Angular (Frontend) (Terminada):** Instalación de Angular y generación de la estructura inicial del proyecto.
- **Creación del Proyecto en Express (Backend) (Terminada):** Instalación de Express y configuración del servidor para gestionar las APIs.
- **Definición de Cronograma (Terminada):** Planificación temporal detallada de todas las fases del proyecto.
- **Creación de Script para la Base de Datos (Terminada):** Definición de tablas, relaciones y datos iniciales en MySQL/PostgreSQL.

Entrega del 80% (Etapa Avanzada)

- **Implementación de Casos de Uso del Área Administrativa (Terminada):**
Desarrollo de funcionalidades como la gestión de productos, usuarios y reportes desde el área de administración.
- **Generación de Interfaz Gráfica (Terminada):** Creación de una interfaz gráfica intuitiva y responsiva para los usuarios y administradores.
- **Implementación de APIs (Terminada):** Desarrollo de los endpoints necesarios para la interacción entre el frontend y backend.
- **Testeo de Implementaciones (Terminada):** Pruebas de funcionalidades implementadas, asegurando su correcto funcionamiento.
- **Agregar Diagramas de Nuevos Casos de Uso (Terminada):** Actualización de la documentación visual con nuevos casos de uso implementados.
- **Implementación de Casos de Uso del Área de Ventas/Compra (Terminada):**
Desarrollo de funciones clave como el carrito de compras, gestión de pedidos y transacciones.
- **Establecimiento de Reglas sobre Moneda Virtual (Terminada):** Definición e implementación de políticas para la moneda virtual utilizada en la plataforma.
- **Agregar Documentación a los Manuales Técnicos (Terminada):** Actualización de los manuales con detalles sobre nuevas implementaciones.
- **Documentación del Código (Terminada):** Inclusión de comentarios y explicaciones detalladas en el código fuente.

Entrega del 100% (Etapa Final)

- **Implementación de Casos de Uso del Área de Voluntariado (Terminada):**
Desarrollo de funcionalidades relacionadas con el voluntariado y la interacción de usuarios en esta área.
- **Testeo General (Terminada):** Pruebas exhaustivas para garantizar el correcto funcionamiento del sistema bajo diferentes condiciones.
- **Elaboración de Manual Técnico (Terminada):** Creación de un manual detallado para los desarrolladores y administradores de la plataforma.
- **Elaboración de Manual de Usuario (Terminada):** Creación de un manual dirigido a los usuarios finales.
- **Implementación de Métodos de Pago (Terminada):** Integración de pasarelas de pago para procesar transacciones de forma segura.
- **Definición e Implementación de Reportes (Terminada):** Desarrollo de funcionalidades para la generación de reportes personalizados para los administradores.
- **Despliegue en Servidor (En proceso):** Configuración del entorno de producción, asegurando la correcta instalación de archivos y bases de datos.
- **Posible Mantenimiento (En proceso):** Planificación de mantenimiento a largo plazo para aplicar actualizaciones y parches.
- **Publicidad de la Aplicación (En proceso):** Estrategias de marketing para promocionar la plataforma en diferentes medios de comunicación.



Diccionario del Proyecto

Plataforma de Comercio Electrónico Error Encoders

El equipo de desarrollo estará compuesto por los siguientes roles:

- **Desarrollador Backend:** Responsable de implementar y mantener la lógica del servidor, microservicios y la conexión con la base de datos.
- **Desarrollador Frontend:** Encargado de desarrollar la interfaz de usuario utilizando **Angular**, asegurando la experiencia de usuario.
- **DBA (Administrador de Base de Datos):** Gestionará y optimizará la base de datos, asegurando su correcta estructura y rendimiento.
- **Desarrollador Full Stack:** Participará en todas las fases del proyecto, coordinando tanto el frontend como el backend y apoyando en la integración.

Primera Entrega (40% del Proyecto):

Nombre	A cargo	Presupuest	o	Tiempo
Definición de entorno de desarrollo	Todos los programadores	Q.0.00		2-3 Días hábiles
Creación de diagramas para abstracción	Todos los programadores	Q.2,500		5 Días hábiles
Toma de Requerimientos	Desarrollador Stack	Full Q.0.00		2 Días (pueden ser en distintas fechas)
Documentar instalación de tecnologías	Todos los programadores	Q.2,500		1 Día hábil
Creación de Repositorio en GitHub	Desarrollador Stack	Full Q.0.00		1 Día hábil
Creación Proyecto en Angular (Frontend)	Desarrollador Frontend	Q.0.00		1 Día hábil
Creación Proyecto en Express (Backend)	Desarrollador Backend	Q.0.00		1 Día hábil
Definición de Cronograma	Desarrollador Stack	Full Q.1,000		1-2 Días hábiles
Creación de Script para la Base de Datos	DBA	Q.5,000		3 Días hábiles

Segunda Entrega (80% del Proyecto):

Nombre	A cargo	Presupuest o	Tiempo
Implementación de casos de uso (área admin)	Desarrollador Frontend/Backend	Q.10,000	7-9 Días hábiles
Generación de interfaz gráfica	Desarrollador Frontend	Q.5,000	7-9 Días hábiles
Implementación de API	Desarrollador Backend	Q.7,000	5 Días hábiles
Testeo de implementaciones	Desarrollador Full Stack	Q.5,000	3 Días hábiles
Agregar diagramas de nuevos casos de uso	Desarrollador Full Stack	Q.0.00	3 Días hábiles
Implementación de casos de uso (ventas/compra)	Desarrollador Frontend/Backend/DBA	Q.30,000	9-12 Días hábiles
Implementacion	Desarrollador Full Stack	Q.0.00	1 Día
Creación de microservicios de acuerdo a casos de uso	Desarrollador Full Stack	Q.0.00	1 Día
Agregar documentación a los manuales técnicos	Todos los programadores	Q.0.00	3 Días hábiles
Documentación de código	Todos los programadores	Q.0.00	Durante todo el proyecto

Tercera Entrega (100% del Proyecto):

Nombre	A cargo	Presupuesto	Tiempo	
Implementación de casos de uso area de venta/carrito	Desarrollador	Q.8,000	3-5	Días
	Frontend/Backend/DBA		hábiles	
Testeo General	Desarrollador Full Stack	Q.12,000	5-7	Días
			hábiles	
Elaboración de Manual Técnico	Todos los programadores	Q.0.00	3-5	Días
			hábiles	
Elaboración de Manual de Usuario	Desarrollador Full Stack	Q.1,000	2-4	Días
			hábiles	
Implementación del método de pago	Desarrollador	Q.5,000	2-4	Días
	Frontend/Backend/DBA		hábiles	
Definición de reportes de la app	Desarrollador Full Stack	Q.2,500	1-2	Días
			hábiles	
Implementación de reportes para el administrador	Desarrollador	Q.15,000	3-5	Días
	Frontend/Backend/DBA		hábiles	
Despliegue en servidor	Desarrollador Full Stack	Q.5,000	2-3	Días
			hábiles	
Posible Mantenimiento	Desarrollador Full Stack	Q.25,000	7-9	Días
			hábiles	
Manejo de excepciones	Todos los programadores	Q.5,000	3-5	Días
			hábiles	

Presupuesto total:

Q.146,500, sujeto a cambios según los ajustes aprobados durante el desarrollo del proyecto.

Diagramas de la solución

Casos de uso Alto Nivel

Caso de uso	Registrar Negocio	CU 001
Actor(es)	Admin	
Poscondición	El admin registra datos de la tienda virtual	
Resumen	El admin, añadirá los datos de la empresa, como logo y nombre de la tienda que se mostraran en toda la aplicación incluyendo las facturas generadas	
Tipo	Primario	

Caso de uso	Registrarse	CU 002
Actor(es)	Cliente	
Poscondición	El cliente se registra en el sistema	
Resumen	El Cliente ingresa sus credenciales para ser guardadas en el sistema.	
Tipo	Primario	

Caso de uso	Inicio de Sesión	CU 003
Actor(es)	Cliente, Administrador, Ayudante	
Poscondición	El usuario ingresa al sistema	
Resumen	El usuario(ya sea Cliente, Administrador o Ayudante) accede al sistema a través de la página por medio de su usuario y contraseña, y será redirigido a su pantalla determinada.	
Tipo	Primario	

Caso de uso	Ver artículos sin ser usuario	CU 004
Actor(es)	Usuario Visitante	
Poscondición	El usuario visitante ve algunas publicaciones, pero si desea ver más, deberá iniciar sesión o registrarse y seguidamente iniciar sesión.	
Resumen	El usuario al ingresar al sistema, podrá ver algunas publicaciones sin embargo si quiere comprar, debe iniciar sesión	
Tipo	Primario	

Caso de uso	Crear usuario	CU 005
Actor(es)	Administrador	
Poscondición	El administrador creará nuevos usuarios	
Resumen	El administrador podrá crear usuarios y asignarle un rol, ya sea Administrador o Ayudante	
Tipo	Primario	

Caso de uso	Ver publicaciones	CU 006
Actor(es)	Cliente	
Poscondición	El cliente podrá ver el listado de productos	
Resumen	El cliente verá “todas” las publicaciones activas	
Tipo	Primario	

Caso de uso	Buscar Producto	CU 007
Actor(es)	Cliente	
Poscondición	El cliente podrá buscar un producto	
Resumen	El cliente podrá buscar un producto por nombre del producto o por categoría de producto	
Tipo	Secundario	

Caso de uso	Filtrar Producto	CU 008
Actor(es)	Cliente	
Poscondición	El cliente podrá filtrar los productos	
Resumen	El cliente podrá filtrar productos por categoría, precio, disponibilidad y otro atributo	
Tipo	Secundario	

Caso de uso	Ver Producto	CU 009
Actor(es)	Cliente	
Poscondición	El cliente podrá ver un producto	
Resumen	El cliente podrá ver un producto en específico, el cual mostrará toda su información como descripción, precio, entre otros	
Tipo	Primario	

Caso de uso	Subir Imagen/es	CU 010
Actor(es)	Administrador, Ayudante	
Poscondición	EL usuario podrá subir imágenes de los productos	
Resumen	El usuario podrá subir imagen / imágenes de los productos para seguidamente registrar el ingreso del producto	
Tipo	Primario	

Caso de uso	Guardar Ingreso Producto	CU 011
Actor(es)	Administrador, Ayudante	
Poscondición	El usuario guarda el ingreso de producto	
Resumen	El usuario luego de subir imagen/imágenes, definen el precio unitario, disponibilidad y descripciones del producto	
Tipo	Primario	

Caso de uso	Actualizar Inventario	CU 012
Actor(es)	Administrador, Ayudante	
Poscondición	El usuario podrá gestionar el inventario	
Resumen	El usuario podrá actualizar la cantidad disponible de cada producto	
Tipo	Primario	

Caso de uso	Gestionar Permisos	CU 013
Actor(es)	Administrador	
Poscondición	El usuario podrá gestionar los permisos	
Resumen	El usuario podrá limitar los permisos de los ayudantes, según el desee (Crear, Editar, gestionar inventario, visualizar pedidos)	
Tipo	Primario	

Caso de uso	Recuperar Contraseña	CU 014
Actor(es)	Cliente	
Poscondición	El usuario podrá restablecer su contraseña	
Resumen	El usuario podrá restablecer su contraseña en caso la haya olvidado, mediante su correo electrónico	
Tipo	Primario	

Caso de uso	Actualizar Perfil	CU 015
Actor(es)	Cliente, Administrador, Ayudante	
Poscondición	El usuario podrá actualizar su perfil	
Resumen	El usuario podrá actualizar su perfil luego de registrarse y iniciar sesion, información como dirección, preferencias de pago y contraseñas	
Tipo	Primario	

Caso de uso	Añadir a Carrito	CU 016
Actor(es)	Cliente	
Poscondición	El usuario podrá añadir productos a su carrito de compras	
Resumen	El usuario navegará por los productos disponibles en tienda, podrá seleccionarlos y agregar a su carrito de compras	
Tipo	Primario	

Caso de uso	Ver Carrito	CU 017
Actor(es)	Cliente	
Poscondición	El usuario podrá ver su carrito de compras	
Resumen	El carrito de compras mostrará todos los productos seleccionados, su precio unitario, cantidad y el total a pagar	
Tipo	Primario	

Caso de uso	Modificar Carrito	CU 018
Actor(es)	Cliente	
Poscondición	El usuario podrá modificar su carrito de compras	
Resumen	El usuario luego de acceder a su carrito, podrá modificar el contenido antes de proceder a comprar, como cambiar cantidades o eliminar productos	
Tipo	Primario	

Caso de uso	Escoger Entrega	CU 019
Actor(es)	Cliente	
Poscondición	El usuario podrá escoger el tipo de entrega	
Resumen	El usuario podrá escoger el tipo de entrega para su producto, ya sea recoger en tienda física o solicitar entrega a domicilio (deberá activar opción de proporcionar dirección de entrega detallada).	
Tipo	Primario	

Caso de uso	Escoger Tipo Pago	CU 020
Actor(es)	Cliente	
Poscondición	El usuario podrá escoger el tipo de pago	
Resumen	El usuario podrá escoger el tipo de pago para su producto, si anteriormente escogió entrega a domicilio, tendrá la opción de pagar con tarjeta o pago contra entrega, si es para recoger en tienda solo tendra la opcion de pagar con tarjeta	
Tipo	Primario	

Caso de uso	Realizar Pago	CU 021
Actor(es)	Cliente	
Poscondición	El usuario podrá realizar el pago	
Resumen	Luego de rellenar los datos según criterio, se deberá realizar el pedido y el pedido entrará en un estado de “Pendiente”	
Tipo	Primario	

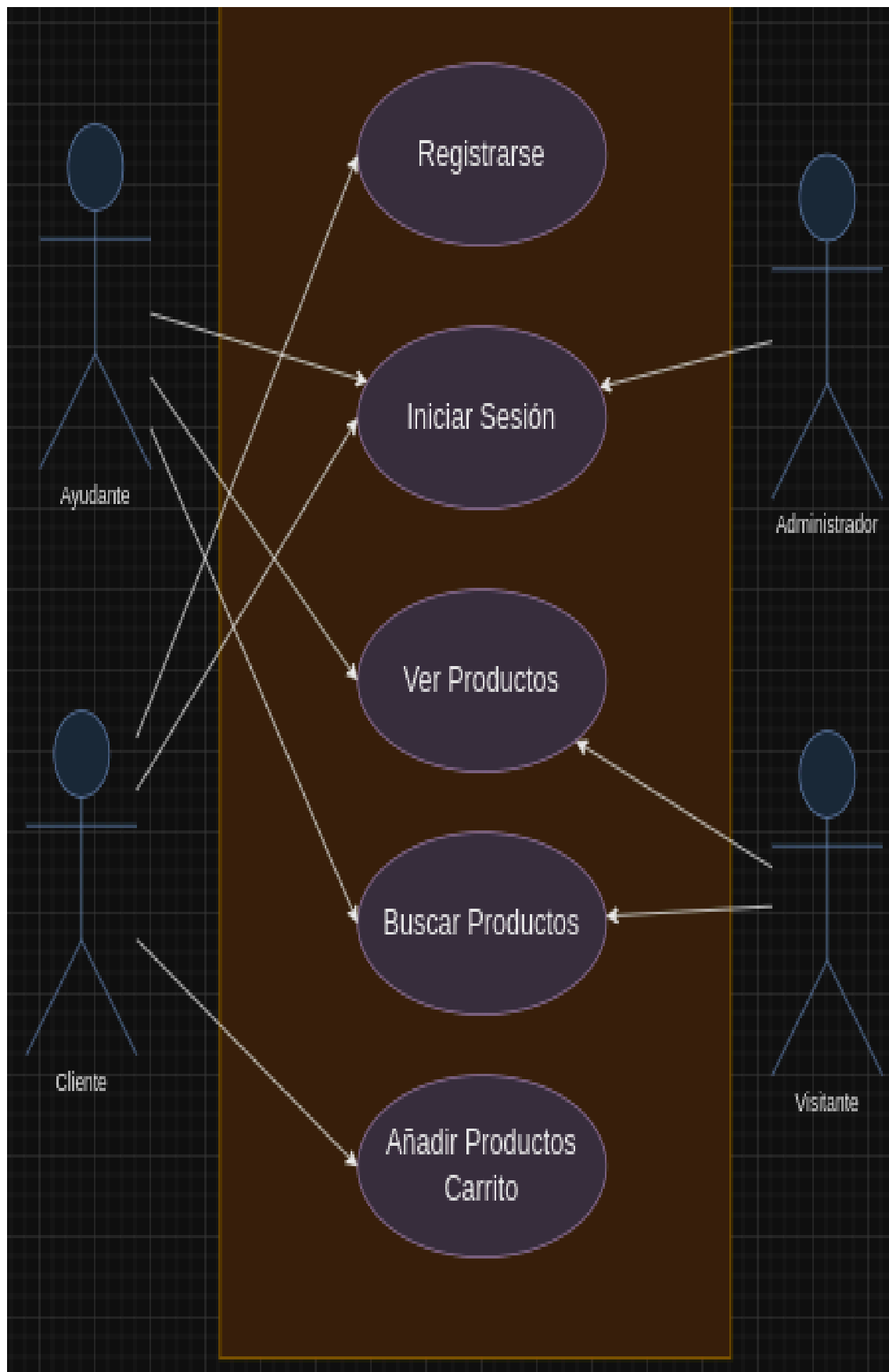
Caso de uso	Procesar pedido	CU 022
Actor(es)	Administrador, Ayudante	
Poscondición	El usuario podrá cambiar el estado del pedido	
Resumen	El usuario cambiará el estado del pedido de “Pendiente” a “Entregado”	
Tipo	Primario	

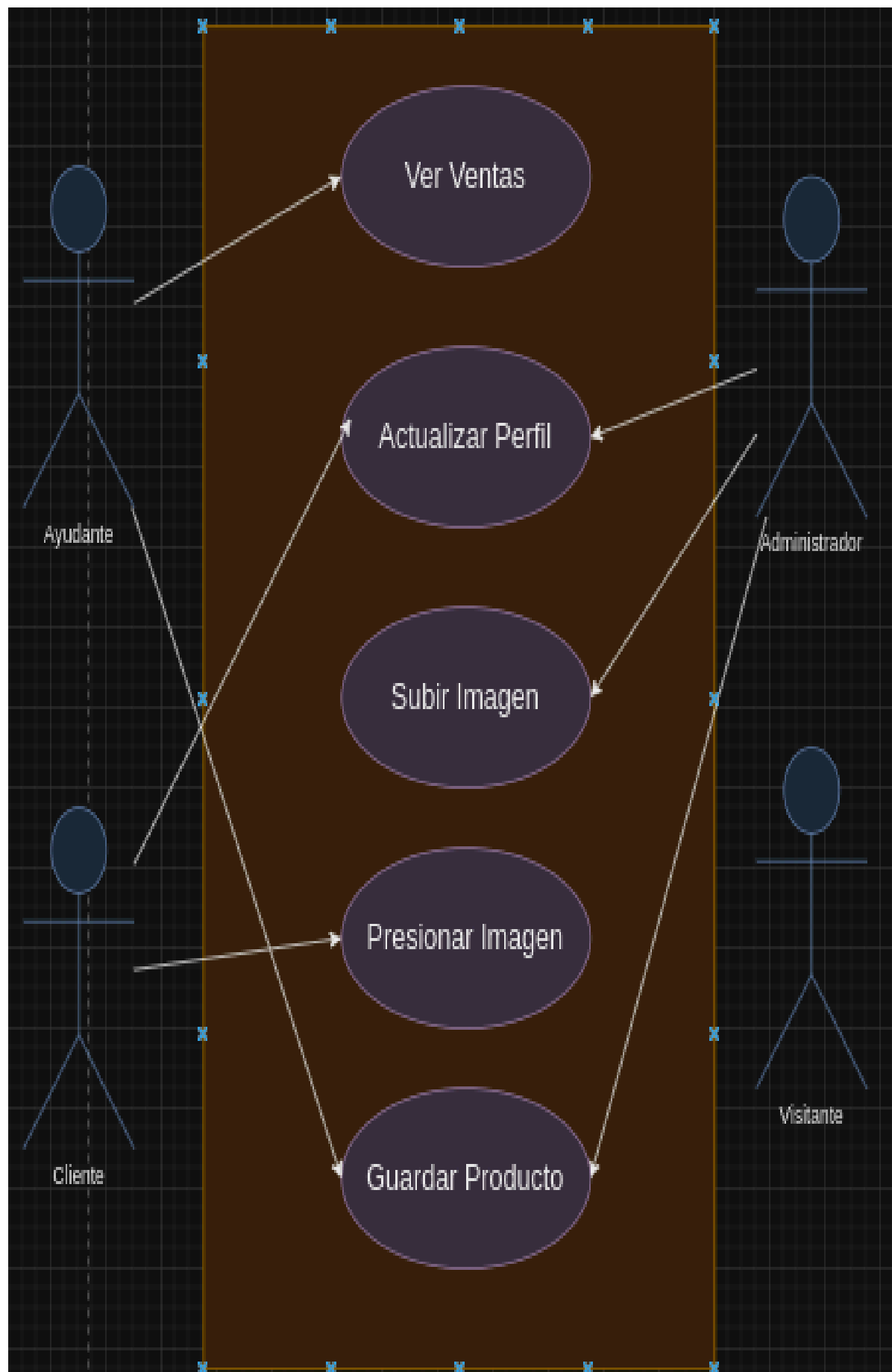
Caso de uso	Notificar pocas unidades	CU 023
Actor(es)	Sistema	
Poscondición	El sistema notificará al Administrador, Ayudante	
Resumen	EL sistema notificará cuando un producto esté cerca de agotarse (inventario cercano a 0)	
Tipo	Primario	

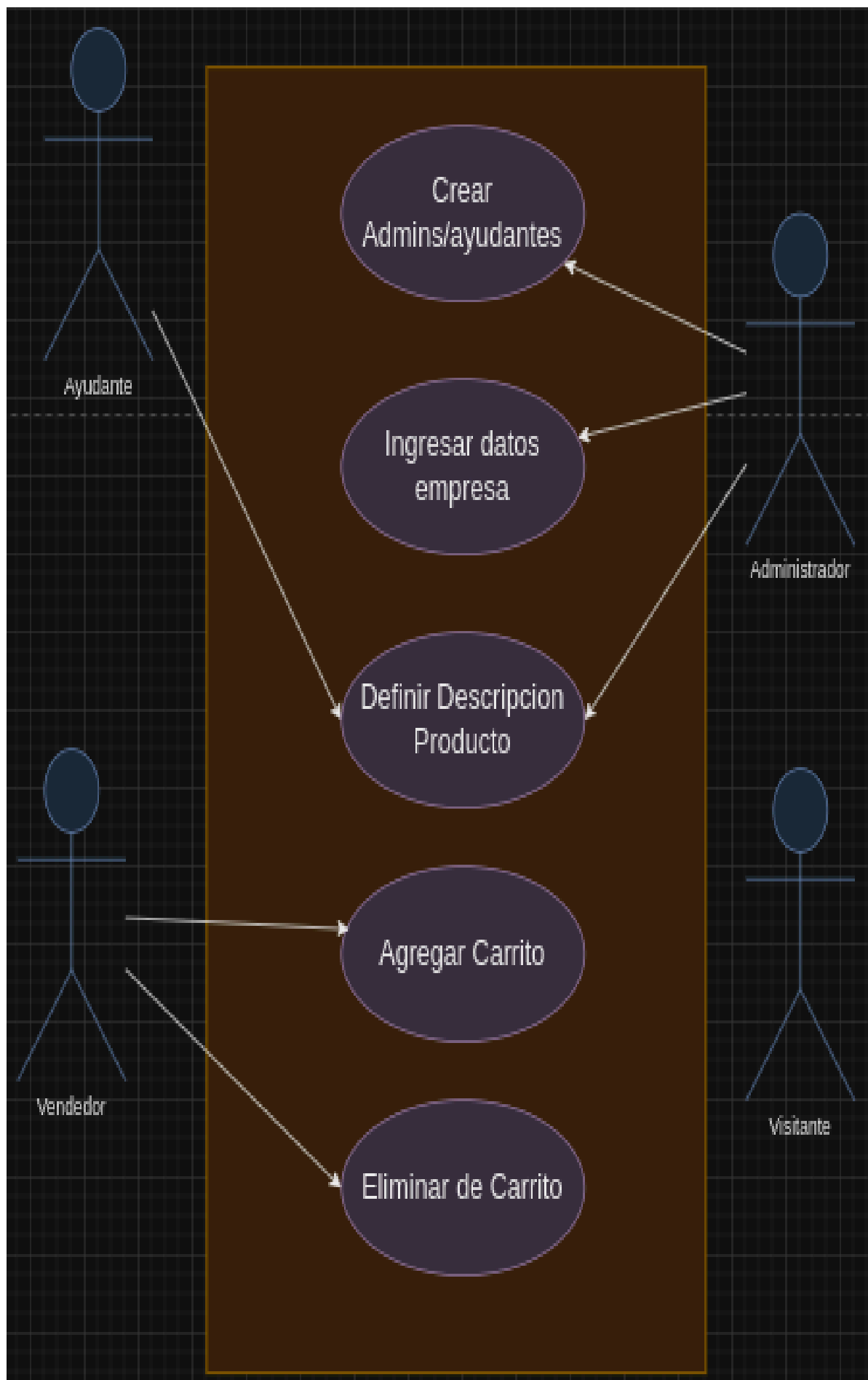
Caso de uso	Reportes	CU 024
Actor(es)	Administrador	
Poscondición	El Administrador podrá ver reportes	
Resumen	El Administrador podrá visualizar gráficas sobre ventas, rendimiento de productos y otras métricas clave	
Tipo	Primario	

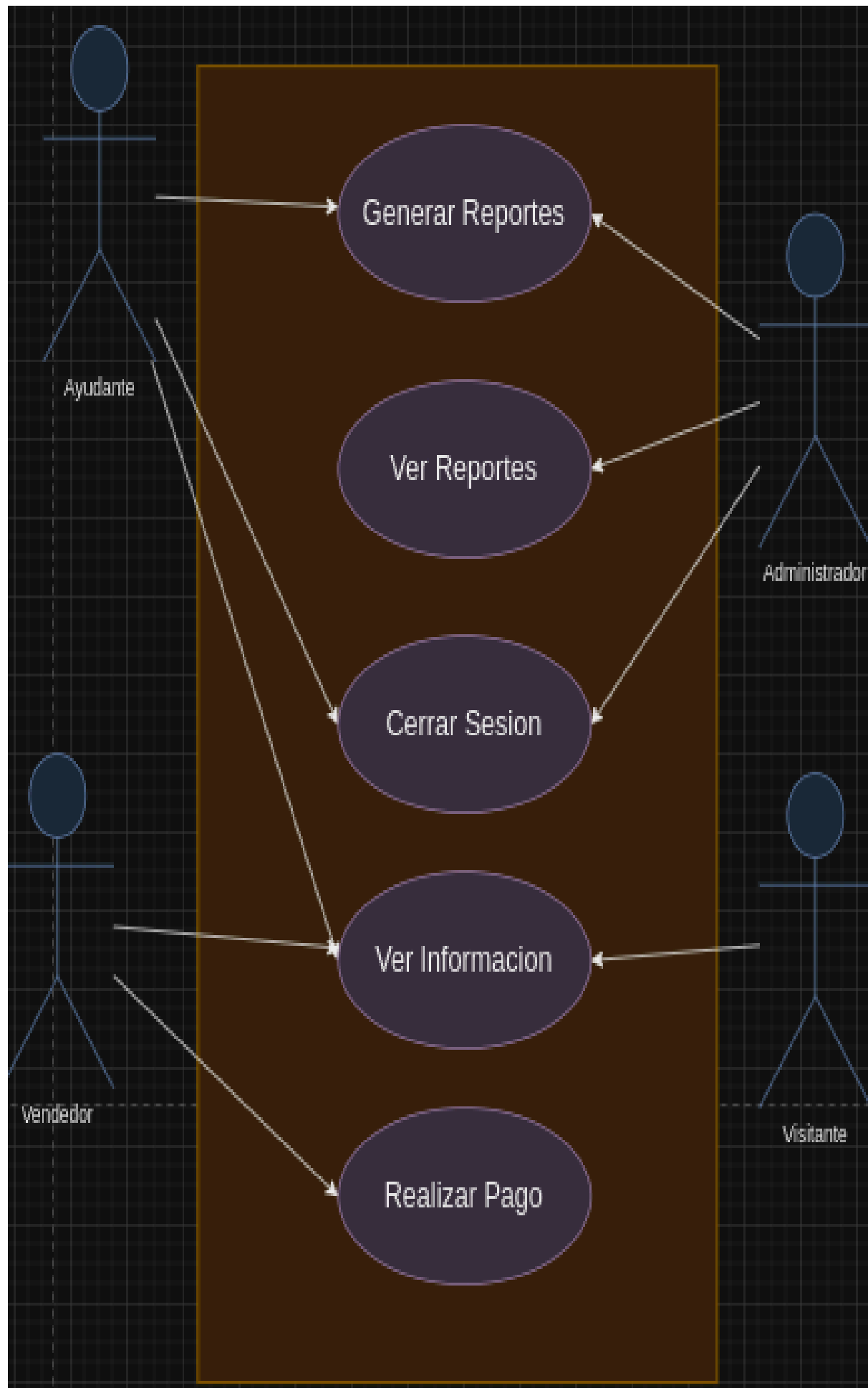
Caso de uso	Generar Reportes	CU 025
Actor(es)	Administrador, Ayudantes (con permisos)	
Poscondición	El usuario podrá generar reportes personalizados	
Resumen	El Administrador podrá generar reportes personalizados sobre ventas, inventario y pedidos	
Tipo	Primario	

Casos de uso (Diagramas)

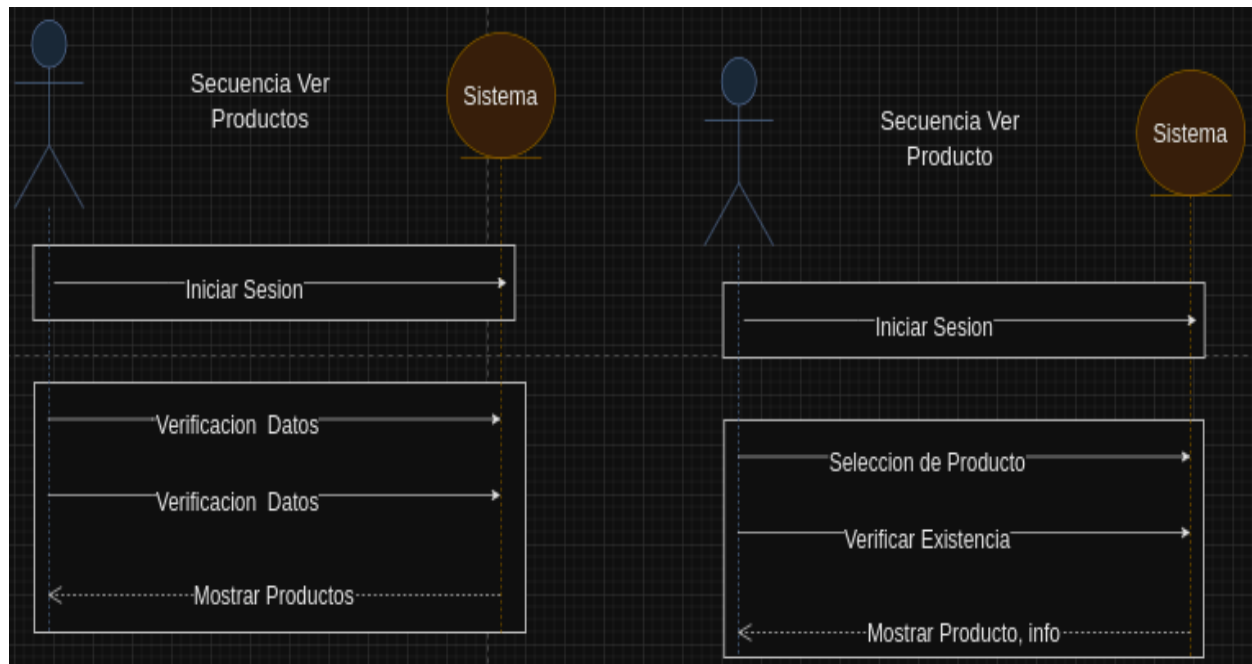
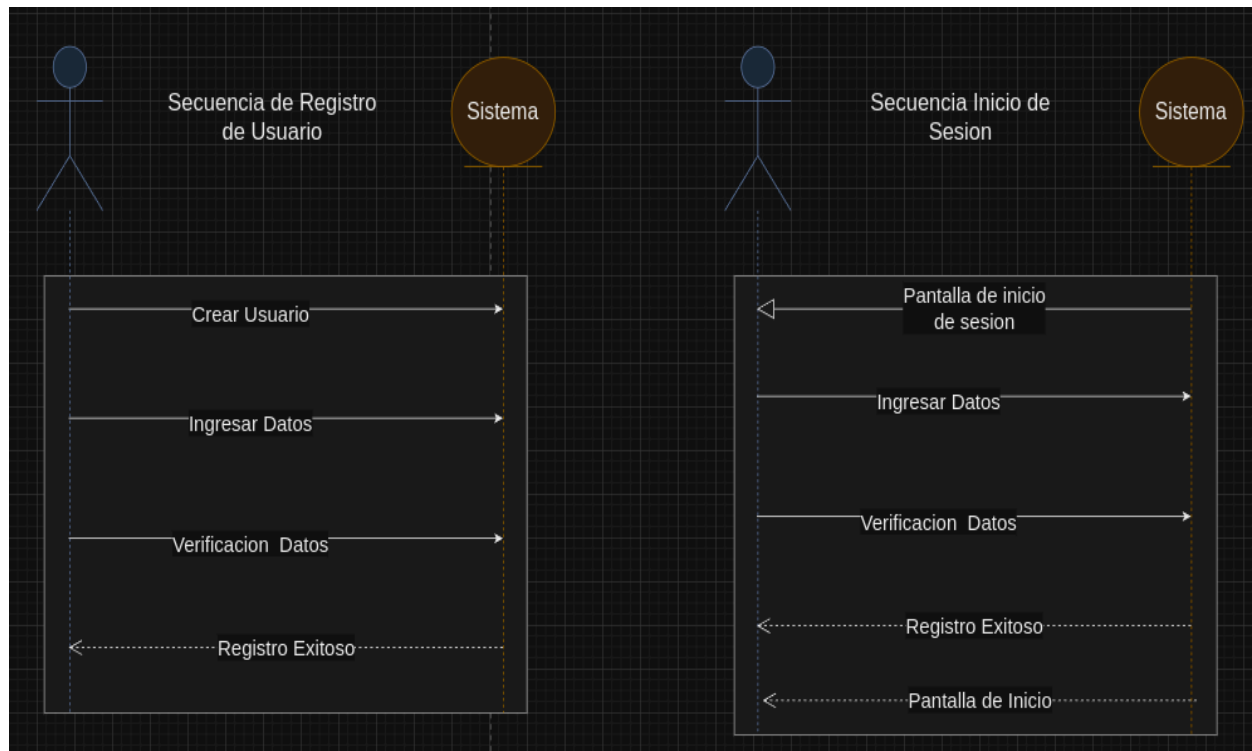


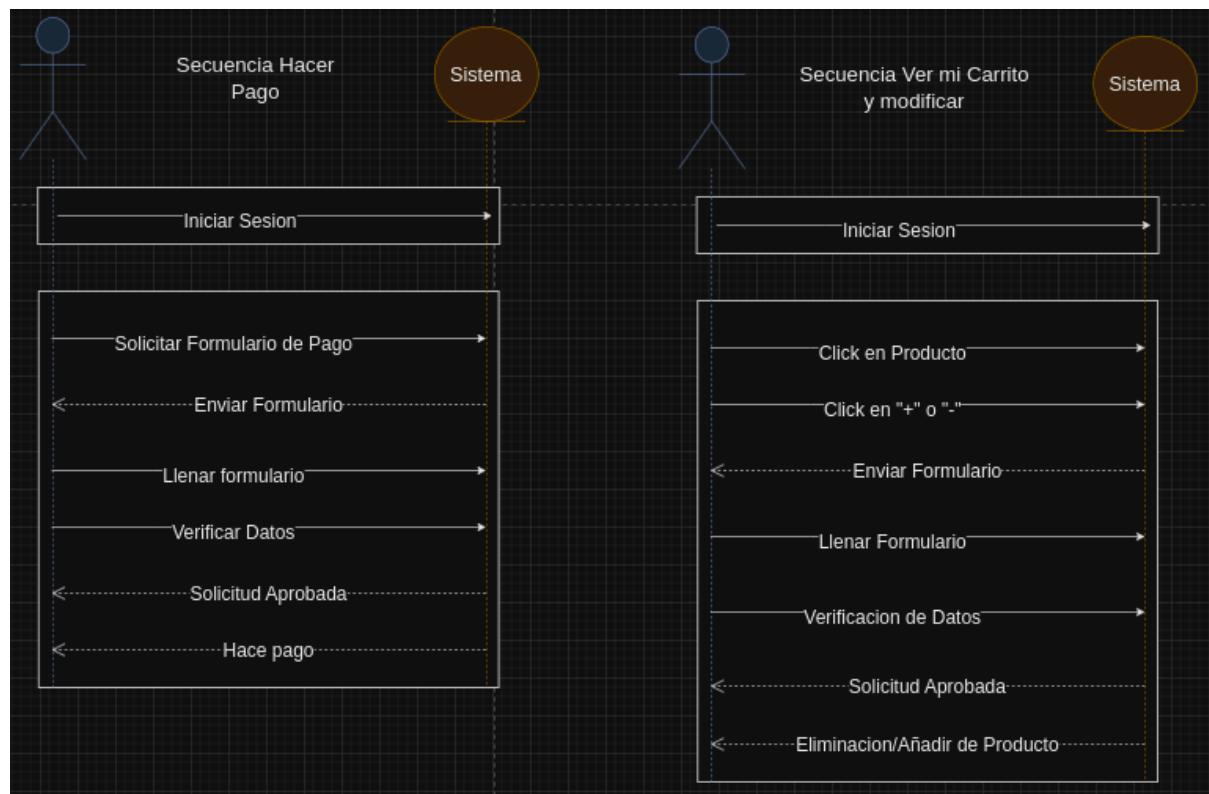
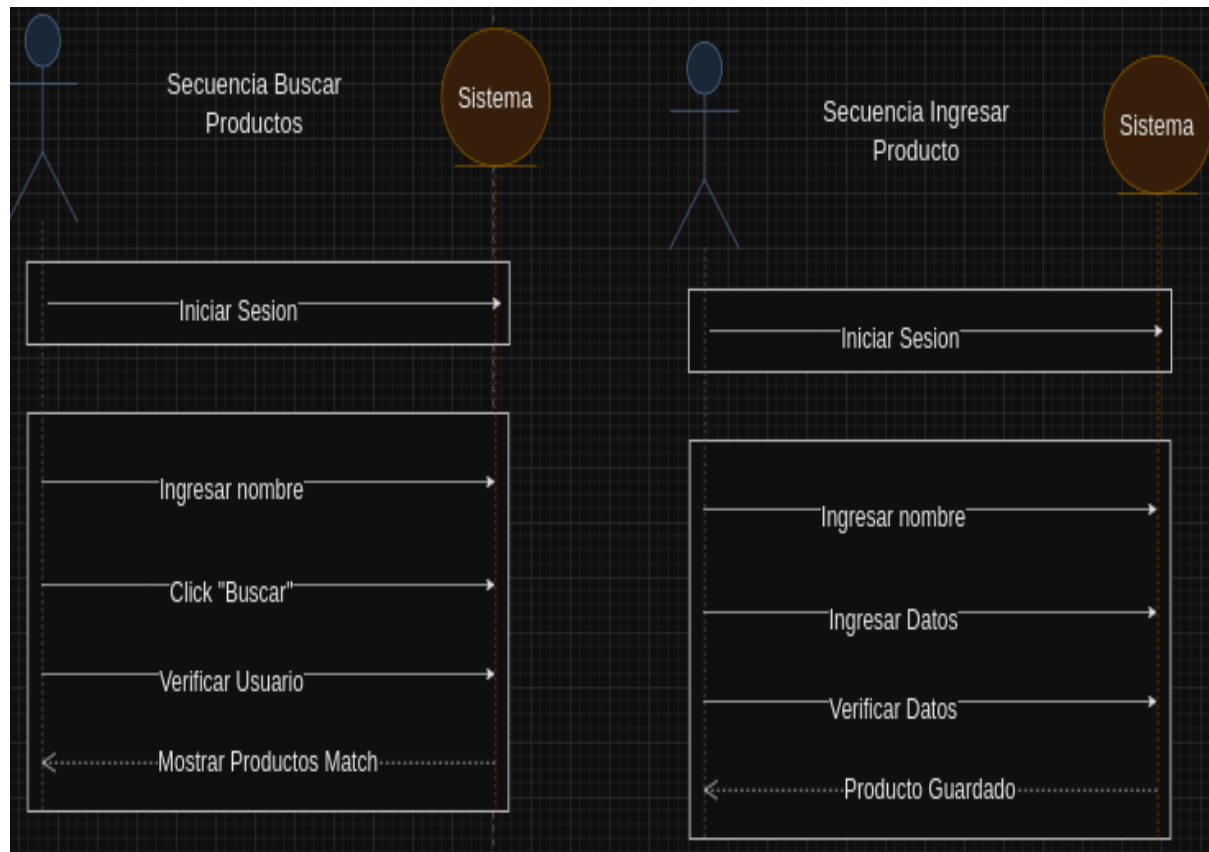




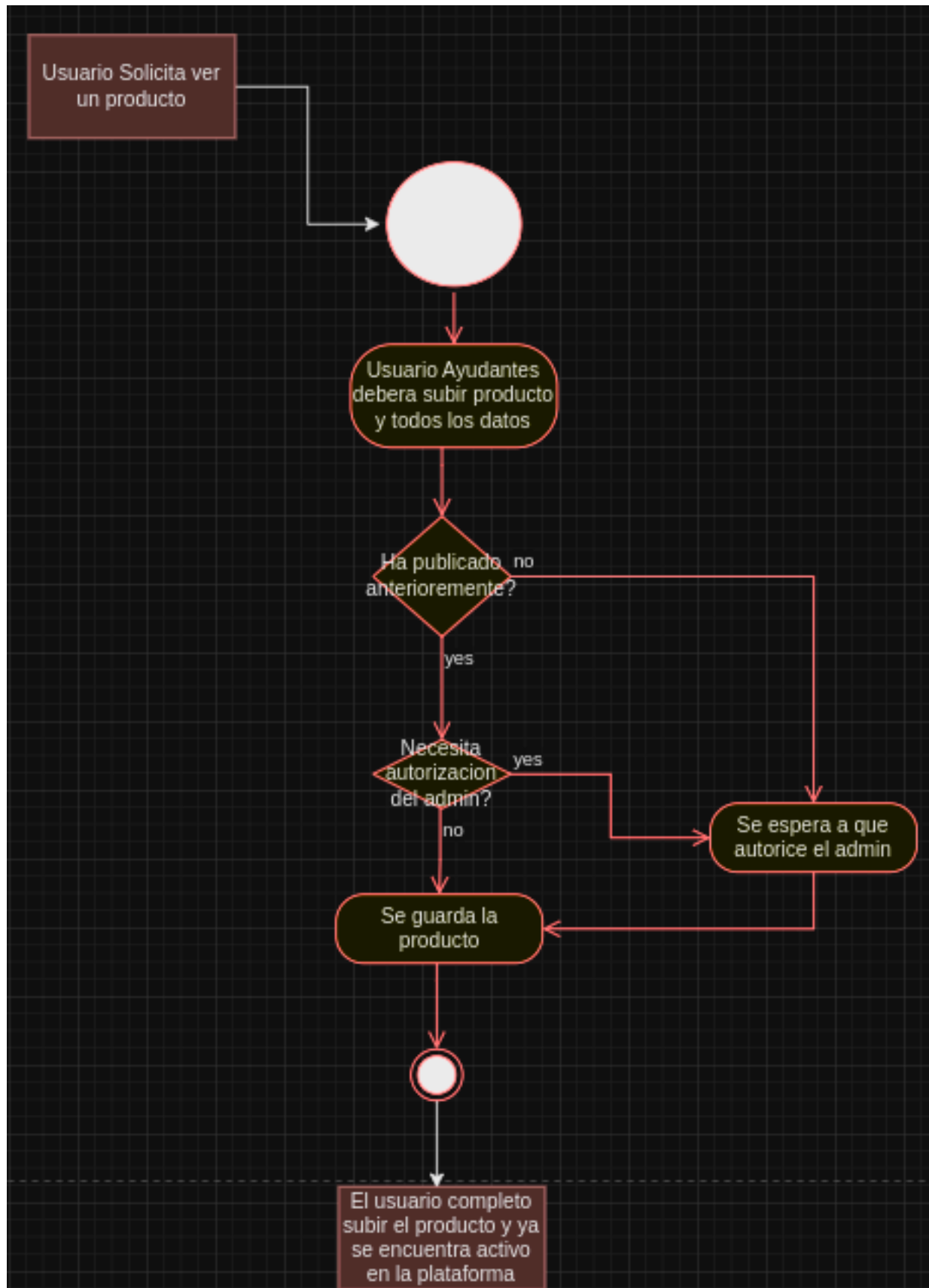


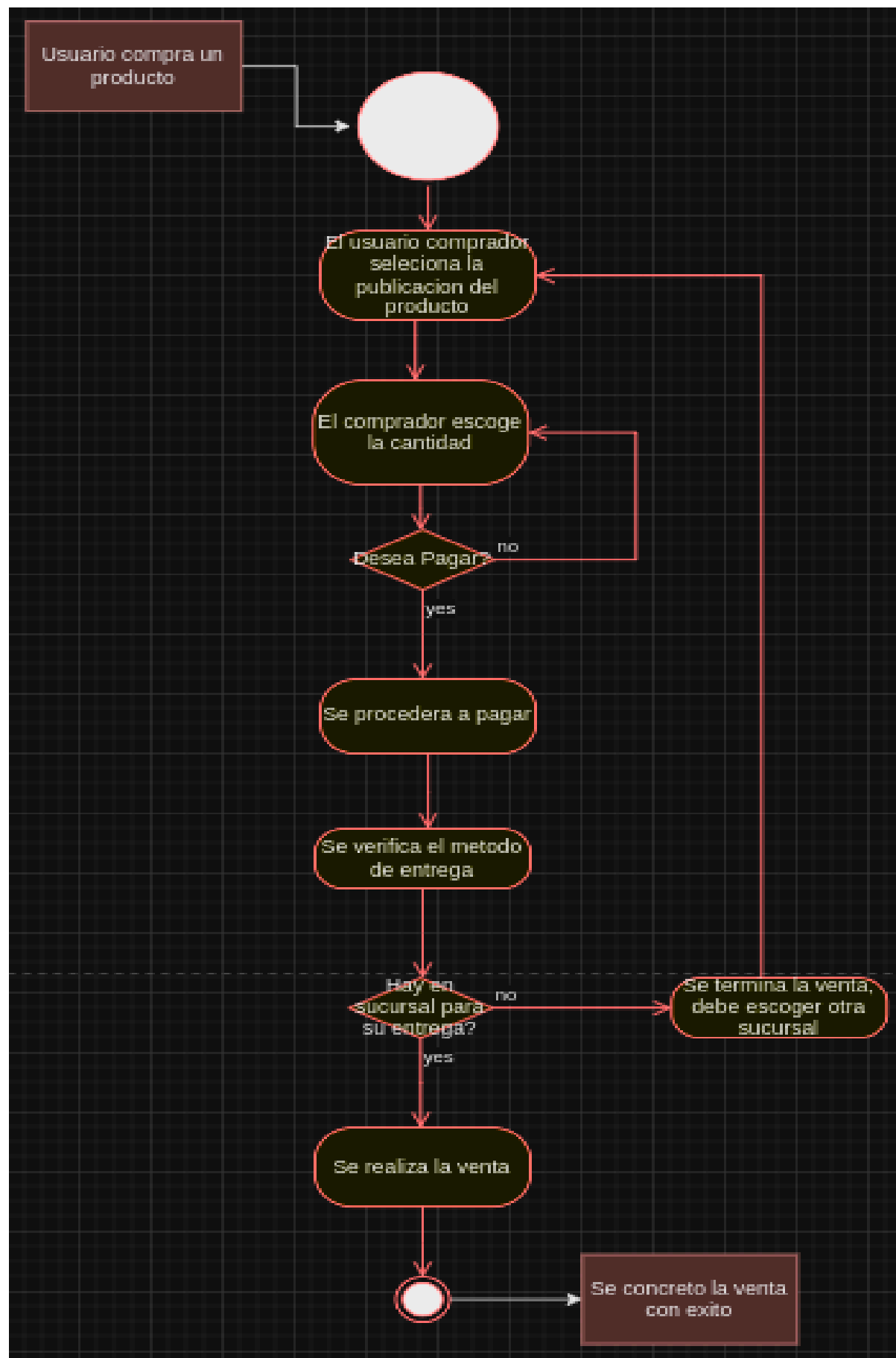
Diagramas de Secuencia





Diagramas de Actividades





Usuario selecciona una publicacion



Producto activo

EL usuario seleccciona para saber mas sobre el producto

Se le presentan los datos de la publicacion

Añade a carrito?

no

yes

Se añade a su carro para poder comprar



Se concreto el añadirlo exito

Organigrama



ORGANIZACION DE LA COMPAÑIA

Marcos Aguiar
DESORRALLOR FULL STACK



Erick Morales
Backend



Oscar de León
Frontend



Levi Hernandez
DBA

Diagrama de Tablas

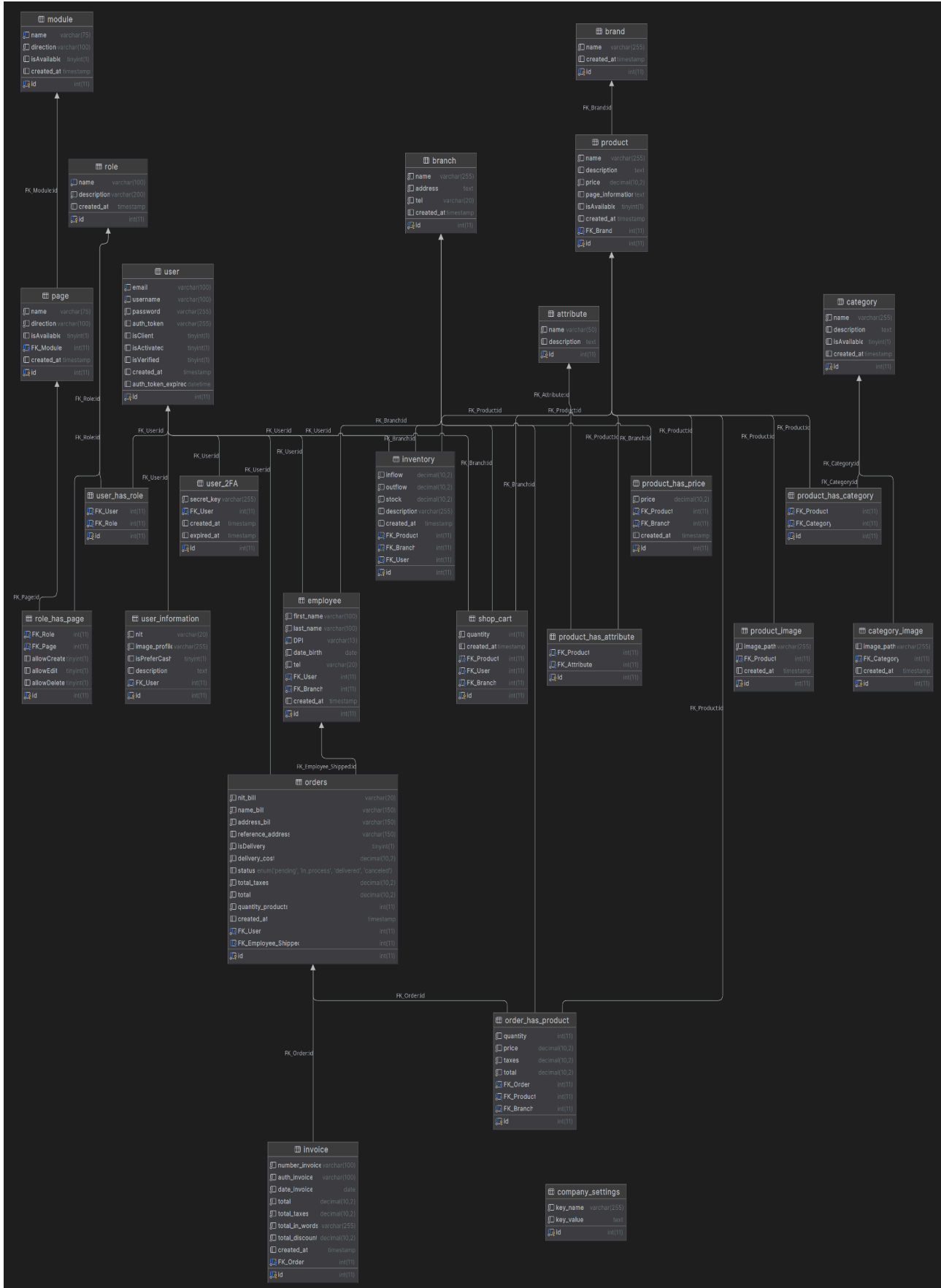


Diagrama de Despliegue

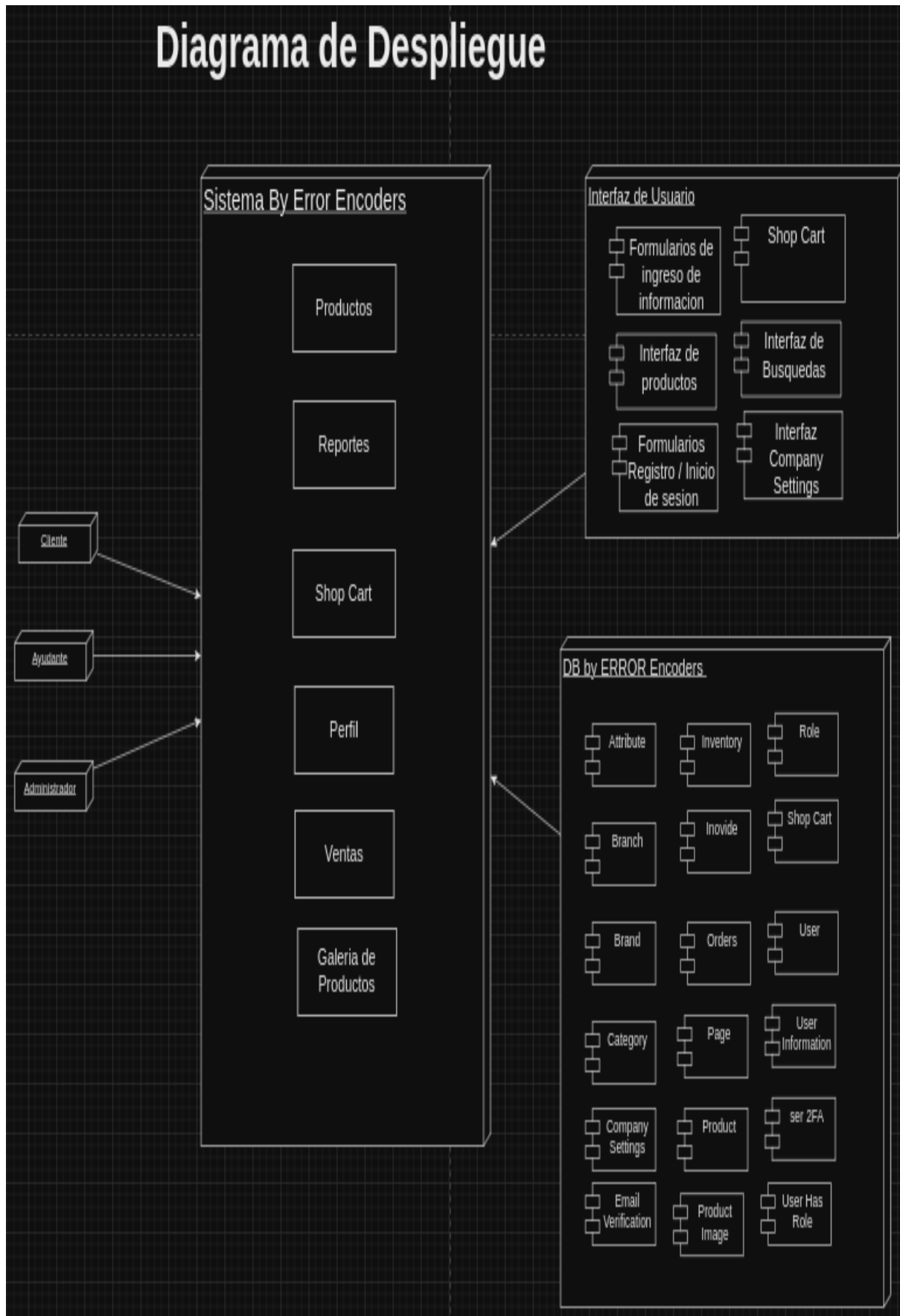


Diagrama de Paquetes

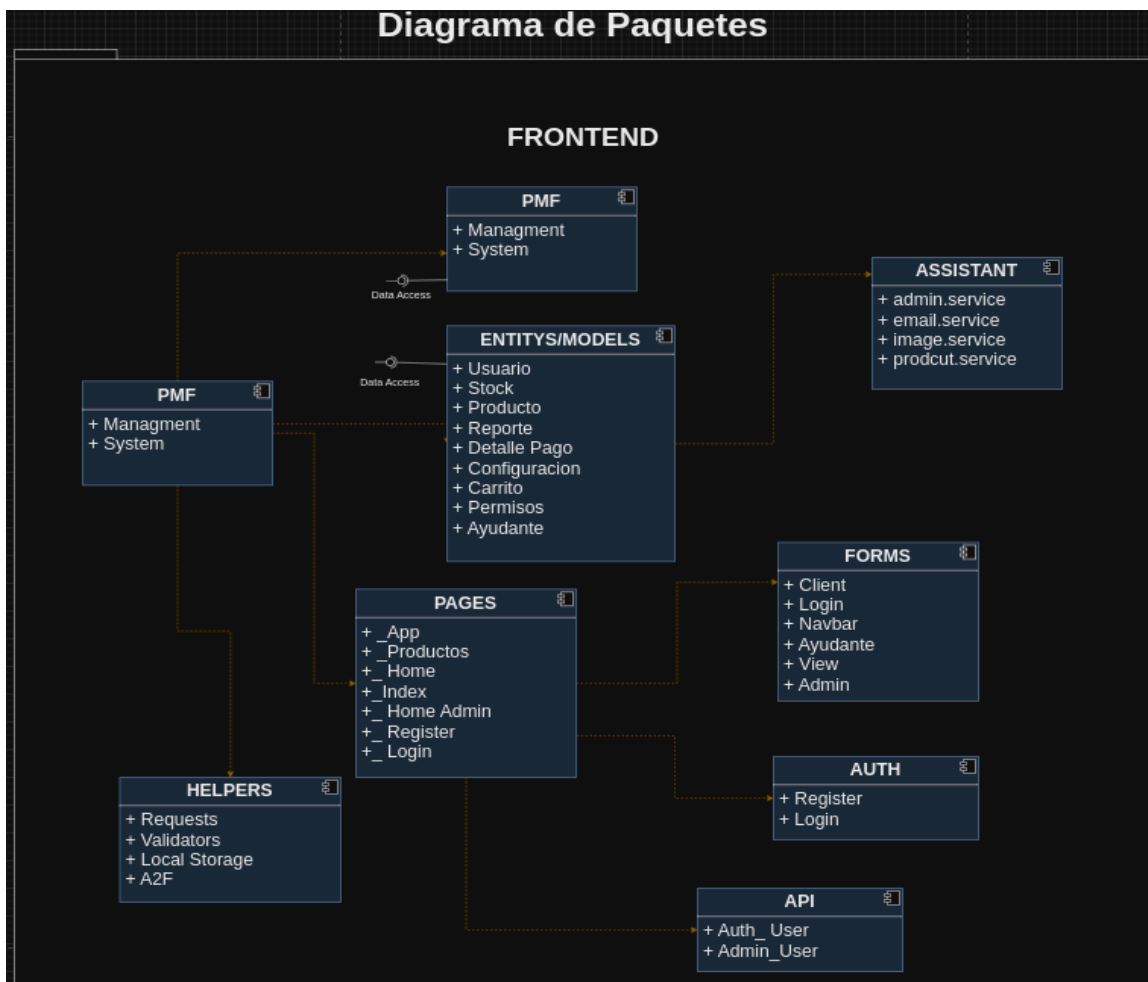
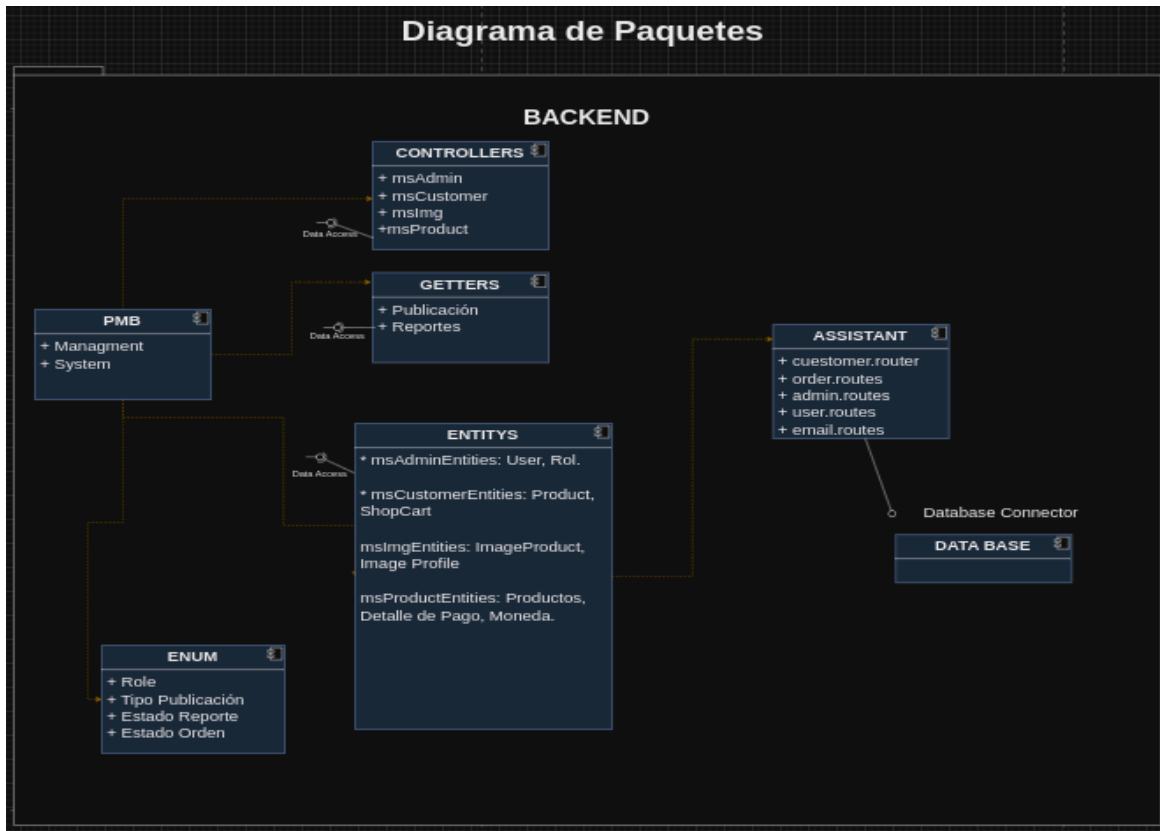
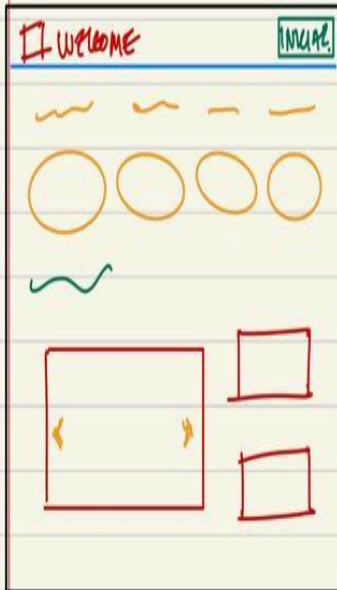


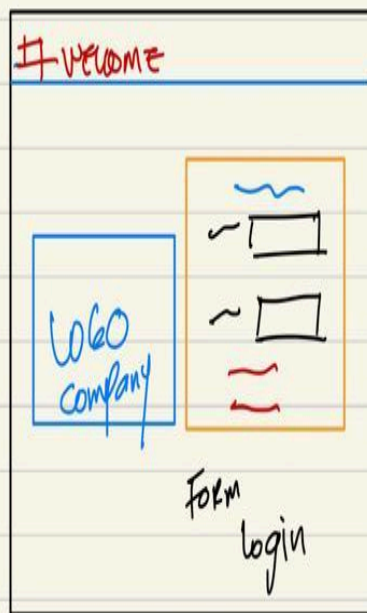
Diagrama de Estructura

Posibles Vistas

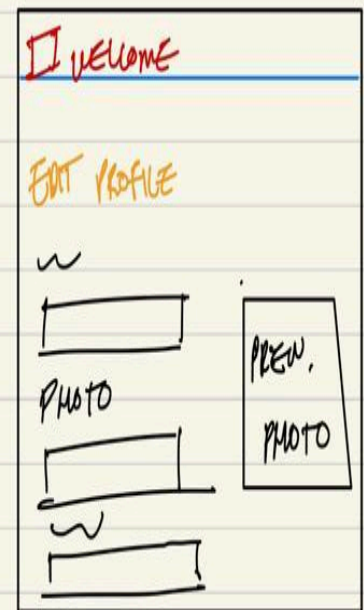
PANTALLA PRINCIPAL



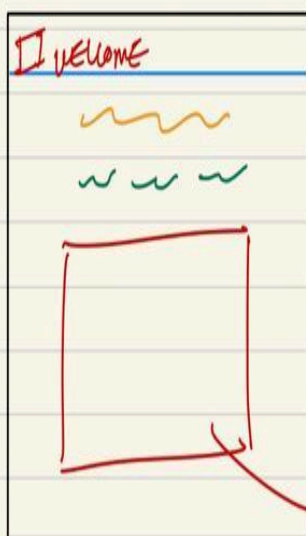
LOGIN



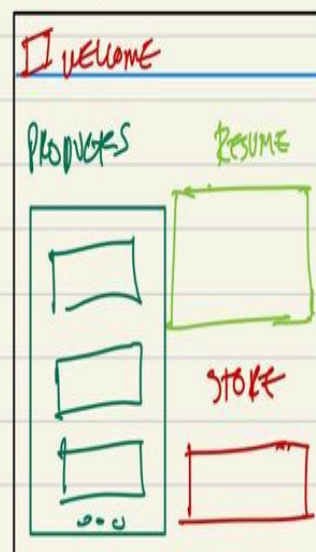
MY PROFILE



COMPANY SETTINGS

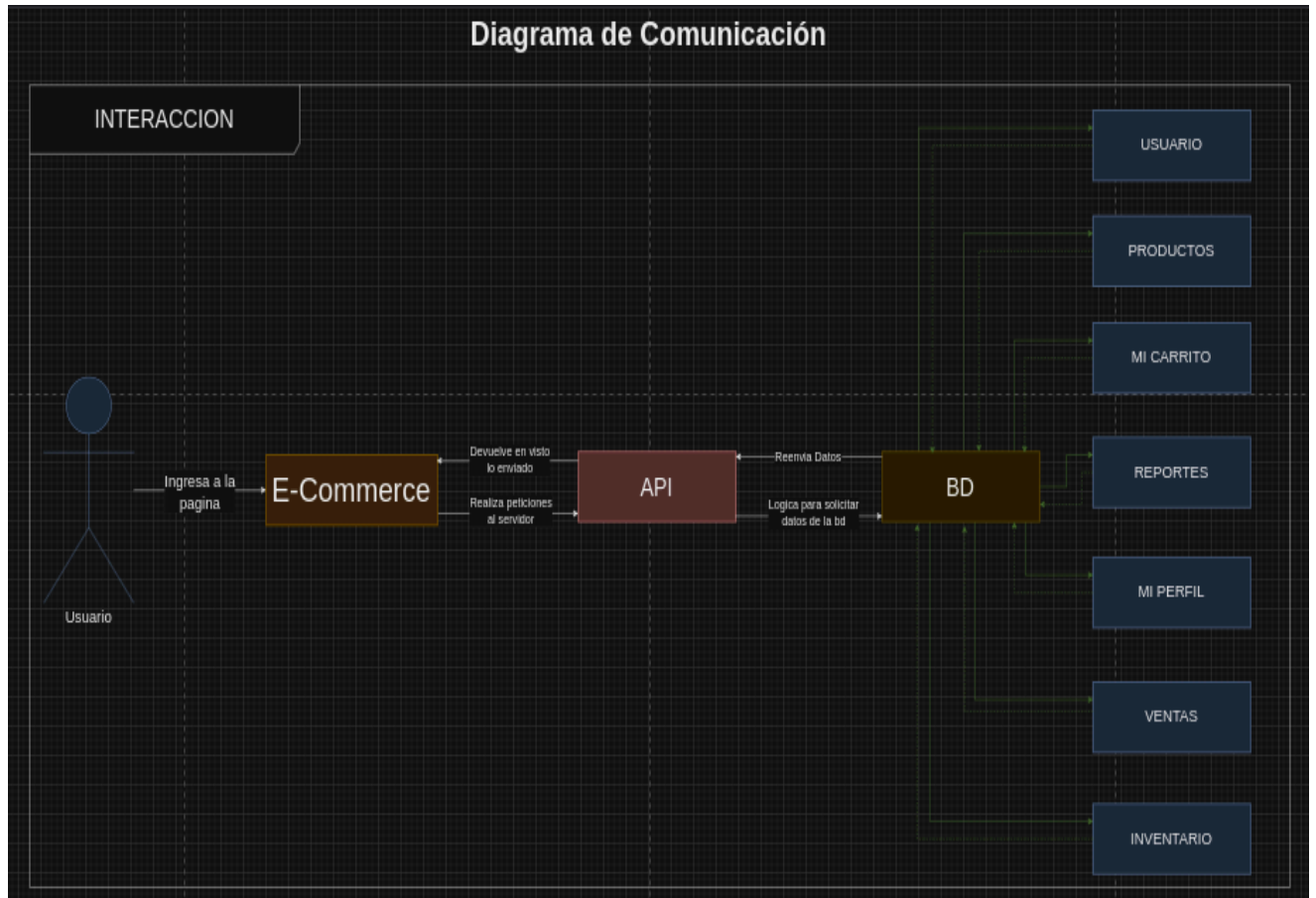


MY SHOP CART

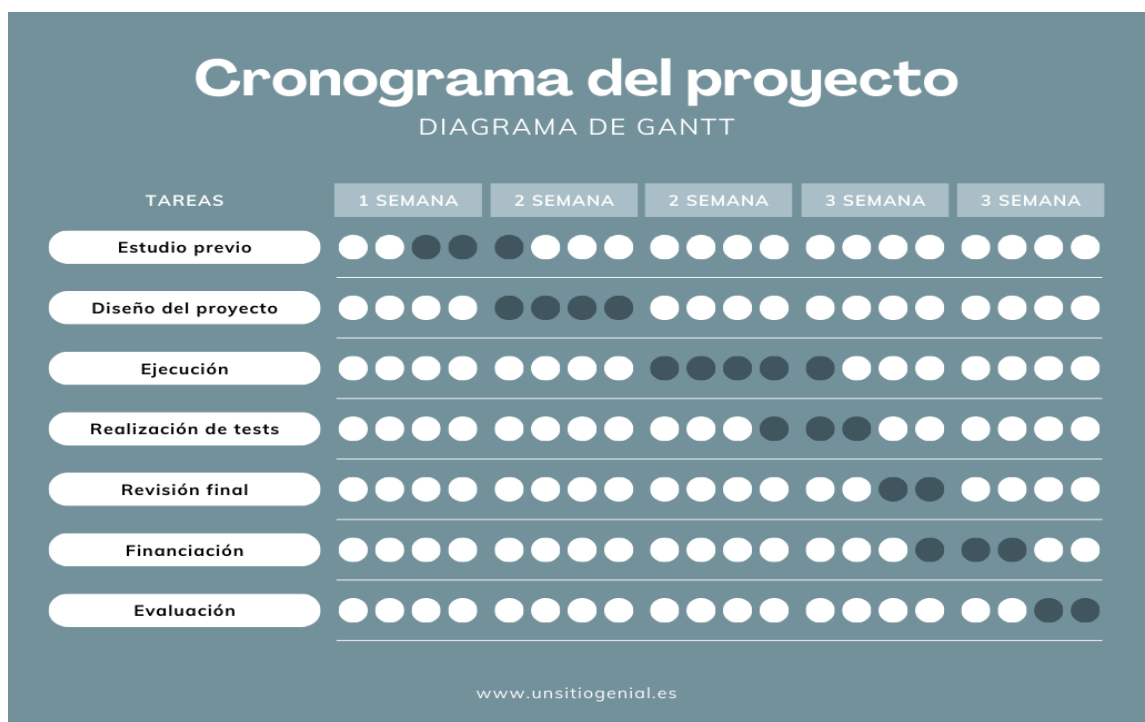


DYNAMIC FORM

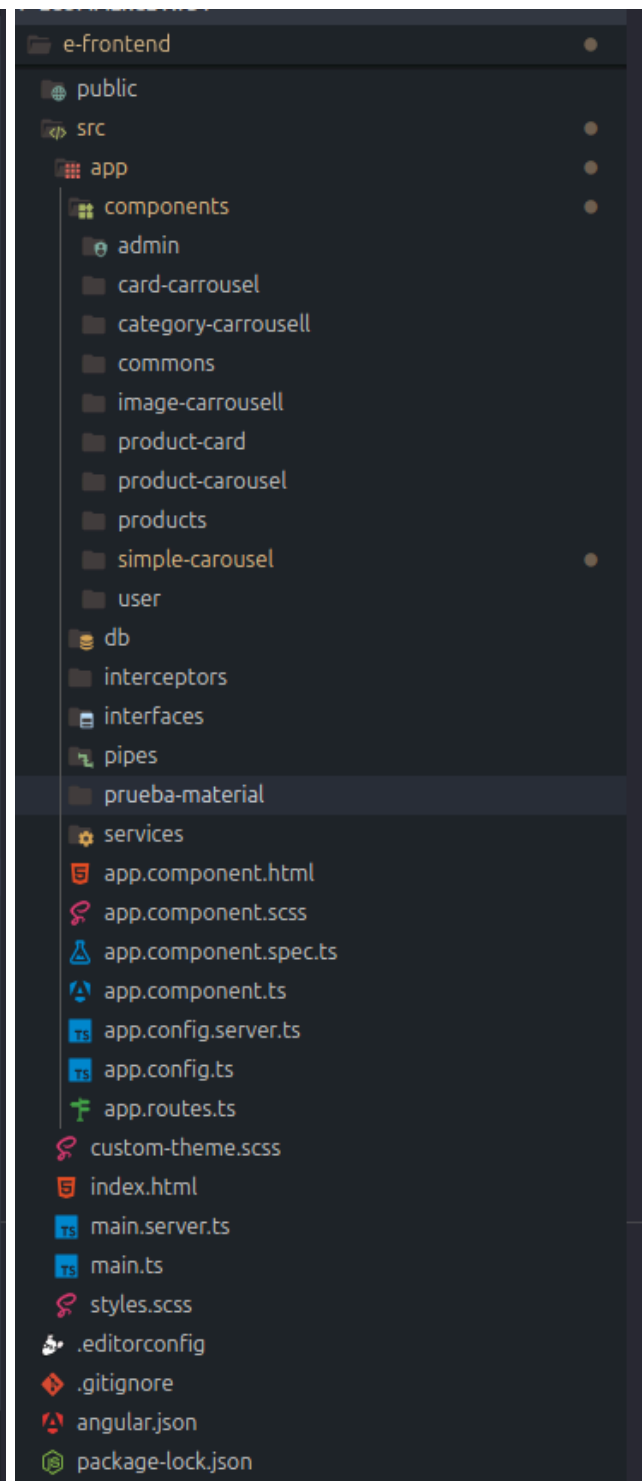
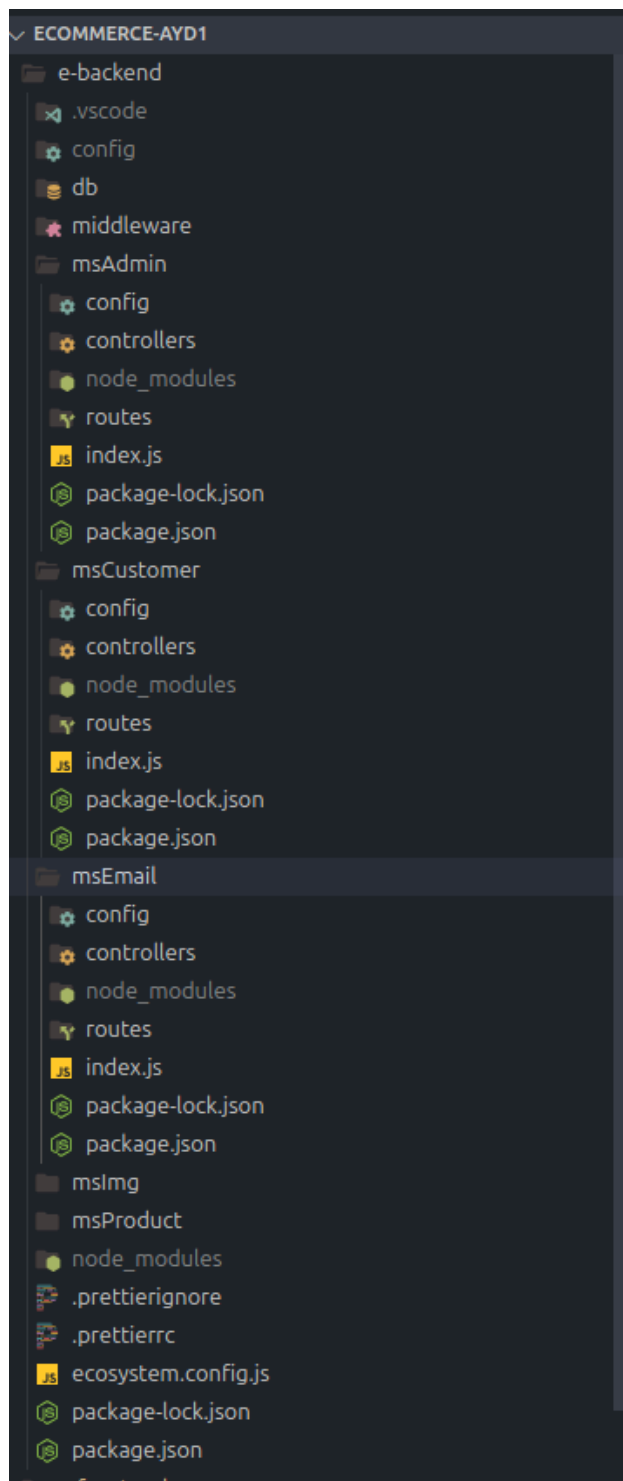
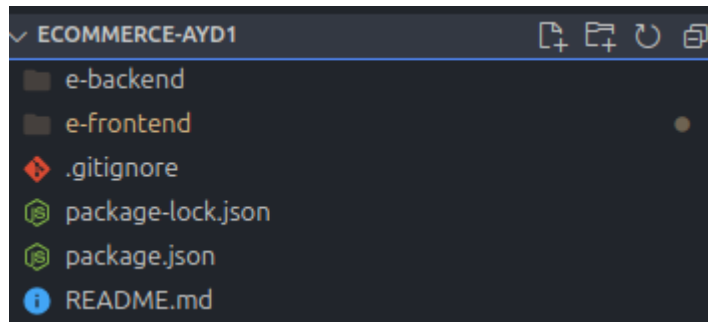
Diagrama de Comunicación



Cronograma



Estructura de Carpetas



Esquema de Arquitectura

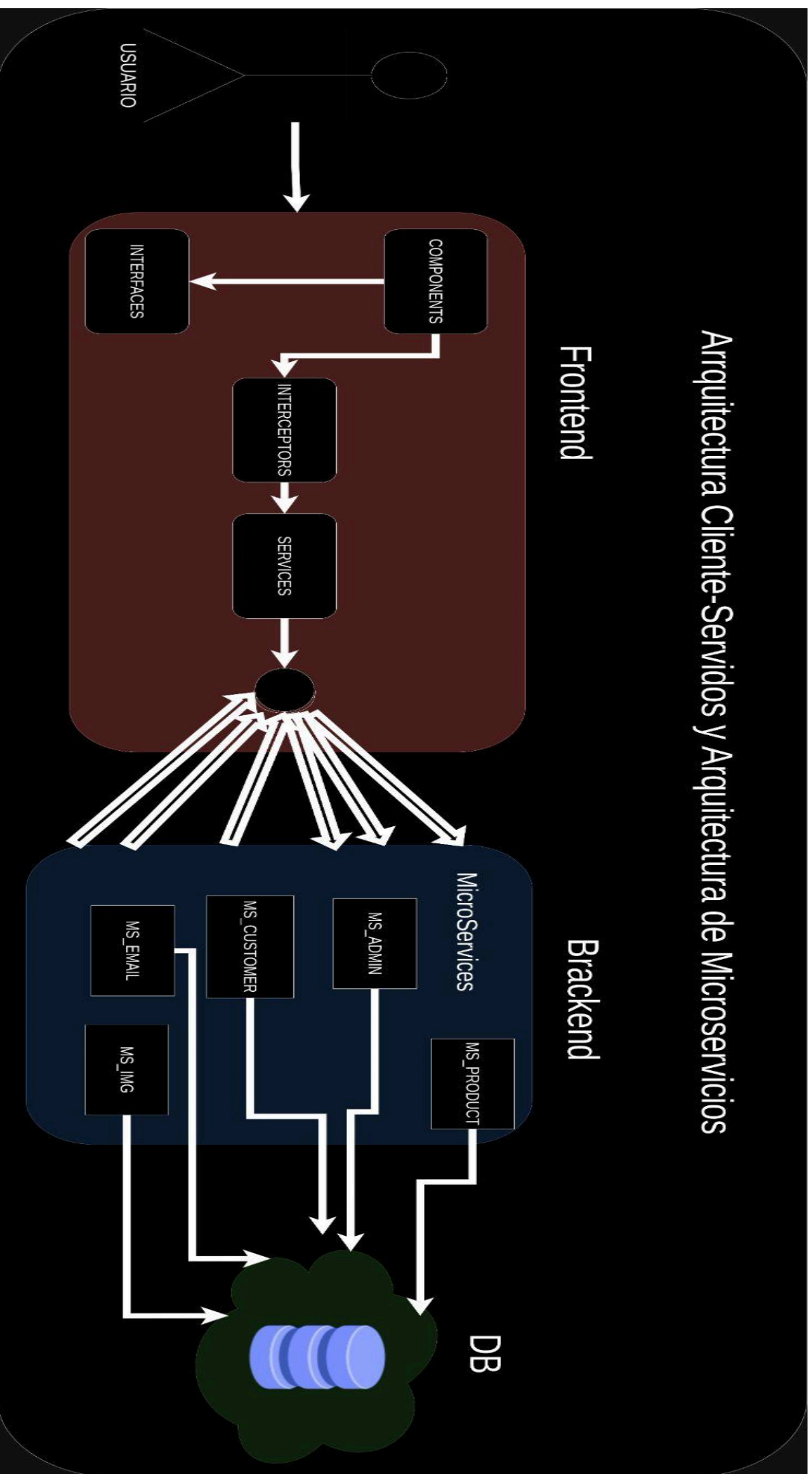
Arquitectura Cliente/Servidor:

- **Razón:** El proyecto sigue una arquitectura cliente/servidor porque se separan dos capas principales: el **frontend** (cliente) y el **backend** (servidor).
 - El frontend (dentro de la carpeta e-frontend) utiliza **Angular** para interactuar con los usuarios finales a través de la interfaz de usuario.
 - El backend (dentro de la carpeta e-backend) está diseñado para manejar la lógica del servidor y la comunicación con la base de datos.
 - Los dos componentes están separados, lo que permite que el cliente envíe solicitudes al servidor, y el servidor responda con los datos o servicios necesarios. Esta interacción define el modelo cliente/servidor.

Arquitectura de Microservicios:

- **Razón:** En el backend, se ha organizado el código en múltiples servicios como msAdmin, msCustomer, msEmail, msImg, y msProduct. Esto es un indicativo de que utilizamos la **arquitectura de microservicios**, donde cada servicio (msAdmin, msCustomer, etc.) tiene una funcionalidad específica e independiente. Estos microservicios pueden comunicarse entre sí, pero se despliegan y escalan de manera independiente.
 - Cada microservicio tiene su propio conjunto de rutas, controladores, y módulos, lo que refuerza la idea de que cada servicio es autónomo.
 - La estructura modular y la capacidad de contener cada microservicio dentro de su propio espacio de configuración y dependencias (notado en los archivos package.json y node_modules dentro de cada servicio) también es un indicador claro de la arquitectura de microservicios.

Arquitectura Cliente-Servidos y Arquitectura de Microservicios



Definición de Actividades

- 1. Definición de entorno de desarrollo**
- 2. Creación de diagramas para abstracción de información**
- 3. Toma de Requerimientos**
- 4. Documentar instalación de tecnologías**
- 5. Creación de Repositorio en la nube (GitHub)**
- 6. Creación del Proyecto en Angular (Frontend)**
- 7. Creación del Proyecto en Express (Backend)**
- 8. Creación del Script de la Base de Datos**
- 9. Implementación de casos de uso (área administrativa)**
- 10. Generación de interfaz gráfica**
- 11. Implementación de API**
- 12. Testeo de implementaciones en la aplicación**
- 13. Agregar diagramas de los nuevos casos de uso**
- 14. Implementación de casos de uso (área ventas/compra)**
- 15. Establecimiento de reglas sobre carrito de compras**
- 16. Establecimiento de reglas sobre sucursales y sus productos**
- 17. Agregar documentación a los manuales técnicos**
- 18. Documentación de código**
- 19. Implementación de casos de uso carrito de compras**
- 20. Testeo General**
- 21. Elaboración de Manual Técnico**
- 22. Elaboración de Manual de Usuario**
- 23. Implementación del método de pago**
- 24. Definición de reportes de la app**

25. **Implementación de reportes para el administrador**
 26. **Publicidad de la Aplicación**
 27. **Gestión con bancos para asociar cuentas y/o tarjetas**
 28. **Despliegue en servidor**
 29. **Posible Mantenimiento**
 30. **Manejo de excepciones**
-

Secuencia de Actividades

Fase de Inicio:

- **Definición de entorno de desarrollo:**

Configuración y selección de las herramientas necesarias para el desarrollo (Angular para frontend, Express para backend, bases de datos).

- **Creación de diagramas para abstracción de información:**

Diseño de diagramas UML, flujo de trabajo, y la estructura de datos para el sistema.

- **Toma de Requerimientos:**

Recolección de las necesidades del sistema a partir de las reuniones con los stakeholders.

- **Documentar instalación de tecnologías:**

Redacción de la guía para la instalación de las herramientas y tecnologías que se utilizarán en el proyecto.

- **Creación de Repositorio en la nube (GitHub):**

Configuración del repositorio para el control de versiones y el trabajo colaborativo.

- **Creación del Proyecto en Angular (Frontend):**

Inicialización y configuración del proyecto de frontend en Angular.

- **Creación del Proyecto en Express (Backend):**

Inicialización y configuración del backend en Express.js.

- **Creación del Script de la Base de Datos:**

Creación de las tablas y relaciones necesarias para las bases de datos, optimizando las consultas según las necesidades del sistema.

Fase de Diseño:

- **Implementación de casos de uso (área administrativa):**

Desarrollo de las funcionalidades administrativas para la gestión de usuarios, productos y configuraciones.

- **Generación de interfaz gráfica:**

Diseño y desarrollo de la UI en Angular, asegurando una experiencia de usuario fluida y responsiva.

- **Implementación de API:**

Desarrollo de los endpoints necesarios para las interacciones entre el frontend y el backend.

- **Testeo de implementaciones en la aplicación:**

Pruebas de las funcionalidades desarrolladas en la aplicación para asegurar que cumplan con los requerimientos.

- **Agregar diagramas de los nuevos casos de uso:**

Actualización de los diagramas del sistema con las funcionalidades recién implementadas.

Fase de Ejecución:

- **Implementación de casos de uso (área ventas/compra):**

Desarrollo de las funcionalidades necesarias para gestionar las ventas y compras de productos.

- **Establecimiento de reglas sobre carrito de compras:**

Implementación de la lógica de negocio relacionada con el manejo del carrito de compras, actualizando productos y cantidades.

- **Establecimiento de reglas sobre sucursales y sus productos:**

Definición de la gestión de las sucursales y la relación con los productos disponibles en cada una.

- **Agregar documentación a los manuales técnicos:**

Actualización de los manuales técnicos con los cambios y mejoras realizados.

- **Documentación de código:**

Adición de comentarios y explicaciones en el código fuente para facilitar su comprensión y mantenimiento.

Fase de Pruebas:

- **Testeo General:**

Realización de pruebas exhaustivas sobre el sistema para garantizar su correcto funcionamiento en diferentes escenarios.

- **Implementación de casos de uso (carrito de compras):**

Desarrollo de la lógica y la funcionalidad completa del carrito de compras, desde la selección de productos hasta la actualización de cantidades y precios.

Fase de Reportes:

- **Implementación del método de pago:**

Integración de pasarelas de pago y la implementación de los métodos de pago, garantizando transacciones seguras.

- **Definición de reportes de la app:**

Establecimiento de los tipos de reportes disponibles para los administradores, con filtros y parámetros personalizados.

- **Implementación de reportes para el administrador:**

Desarrollo de la funcionalidad para que los administradores puedan acceder a reportes detallados sobre productos, ventas, y otros aspectos relevantes del negocio.

- **Agregar documentación a los manuales técnicos:**

Inclusión de detalles técnicos adicionales y actualizaciones en los manuales.

Fase de Publicidad y Mantenimiento:

- **Publicidad de la Aplicación:**

Estrategias de marketing y publicidad de la aplicación en medios digitales y redes sociales para atraer clientes.

- **Gestión con bancos para asociar cuentas y/o tarjetas:**

Proceso de integración con bancos para asociar cuentas o tarjetas en la plataforma de pagos.

Fase de Finalización:

- **Revisión final:**

Verificación de que todas las funcionalidades estén completamente implementadas y pruebas de la estabilidad general del sistema.

- **Elaboración de Manual Técnico:**

Creación de un manual para desarrolladores y administradores con detalles técnicos del sistema.

- **Elaboración de Manual de Usuario:**

Creación de un manual para usuarios finales, cubriendo las principales funcionalidades del sistema.

- **Despliegue en servidor:**

Implementación final en un entorno de producción, utilizando un servidor configurado para alojar la aplicación.

- **Posible Mantenimiento:**

Definición de un plan de mantenimiento preventivo y correctivo para asegurar la continuidad y el correcto funcionamiento de la plataforma.

- **Manejo de excepciones:**

Implementación de manejo de errores y excepciones para mejorar la resiliencia y estabilidad del sistema.

Fechas Importantes

Elaboración del Acta de Constitución del Proyecto:

- **12/08/2024**

Primer Entregable (40% del Proyecto):

- **23/08/2024**

Incluye la definición del entorno de desarrollo, creación de diagramas, toma de requerimientos, repositorio en GitHub y la creación de los proyectos en Angular y Express.

Segundo Entregable (80% del Proyecto):

- **04/09/2024**

Incluye la implementación de los casos de uso (administrativa, ventas, compras, carrito de compras), testeo general y establecimiento de reglas del negocio.

Tercer Entregable (100% del Proyecto):

- **11/09/2024**

Incluye la implementación final de reportes, manuales, despliegue en servidor, y la gestión con bancos.

Tareas y Fechas por Semana

1. Semana 1 (12/08/2024 - 18/08/2024):

- Definición de entorno de desarrollo
- Creación de diagramas para abstracción de información
- Toma de Requerimientos

2. Semana 2 (19/08/2024 - 25/08/2024):

- Documentar instalación de tecnologías
- Creación de Repositorio en GitHub
- Creación del Proyecto en Angular (Frontend)
- Creación del Proyecto en Express (Backend)

3. Semana 3 (26/08/2024 - 01/09/2024):

- Creación del Script de la Base de Datos
- Implementación de casos de uso (área administrativa)
- Generación de interfaz gráfica

4. Semana 4 (02/09/2024 - 08/09/2024):

- Implementación de API
- Testeo de implementaciones en la aplicación
- Establecimiento de reglas sobre carrito de compras y sucursales

5. Semana 5 (09/09/2024 - 11/09/2024):

- Elaboración de Manual Técnico y Manual de Usuario
- Implementación del método de pago
- Despliegue en servidor
- Revisión final y pruebas

Conclusión

Error Encoder presenta una solución robusta para responder a las necesidades del creciente mercado de comercio electrónico, manteniendo un firme compromiso con las políticas de privacidad, estrategias de marketing efectivas y la satisfacción del cliente. Nos enfocamos en ofrecer una plataforma competitiva que impulse el desarrollo económico local y potencie el crecimiento de las comunidades mediante nuestra solución tecnológica innovadora.

Error Encoder tiene el potencial de posicionarse como una plataforma líder en su espacio, generando un impacto tanto en el comercio electrónico como en la tecnología. Nuestra visión se enfoca en:

1. Potenciar el Impacto Social:

Al ofrecer una plataforma que facilita el intercambio de bienes y servicios, promovemos la inclusión económica, fortalecemos el comercio local y ayudamos a comunidades a mejorar su acceso al mercado.

2. Oportunidades de Crecimiento:

Error Encoder tiene grandes oportunidades de expansión mediante asociaciones estratégicas, como colaboraciones con bancos locales para simplificar las transacciones dentro de la plataforma. A medida que integramos nuevas funcionalidades y exploramos nuevos mercados, garantizamos el crecimiento sostenible a largo plazo.

Con esta visión, **Error Encoder** está preparada para convertirse en un referente en el sector, contribuyendo al desarrollo social y económico mientras ofrecemos una experiencia de usuario eficiente y segura.

Preguntas

Después de todo el proyecto, podemos responder las siguientes preguntas:

¿Cuál es el problema?

El problema radica en la ausencia de una plataforma eficiente que permita a los usuarios de una comunidad crear un perfil, adquirir productos exclusivos de la empresa **Error Encoder**, y gestionar compras de manera segura y conveniente.

¿Cuál es la solución?

La solución propuesta es desarrollar una plataforma de comercio electrónico personalizada para **Error Encoder**, que facilite todas estas actividades mediante una interfaz dinámica, segura y orientada al cliente.

¿Quién?

El equipo de desarrollo estará formado por programadores, desarrolladores Full Stack, desarrolladores Frontend y Backend, un DBA (Administrador de Bases de Datos) y otros profesionales que aseguren la implementación exitosa del proyecto.

¿Dónde?

El desarrollo del proyecto se llevará a cabo en un entorno de trabajo dedicado a la programación y el diseño de software, con la colaboración remota de equipos según sea necesario.

¿Cómo?

La plataforma será desarrollada utilizando **Angular** para el frontend, **Express** para el backend, y **MariaDB** como el sistema gestor de bases de datos. El desarrollo estará centrado en la seguridad, escalabilidad y usabilidad del usuario.

¿Cuándo?

El proyecto seguirá un cronograma basado en entregas clave:

- **Inicio del Proyecto:** 12/08/2024
- **Finalización del Proyecto:** 11/09/2024

¿Cuánto costará?

El costo estimado del proyecto se ha fijado en **Q.146,500**, teniendo en cuenta los presupuestos detallados para cada fase de desarrollo, así como los posibles gastos adicionales.

¿Con qué se llevará a cabo la solución?

La solución será implementada mediante el desarrollo de software personalizado utilizando **Angular**, **Express**, **MySQL**, y políticas de privacidad y seguridad robustas que garanticen la protección de los datos y transacciones.

¿Cuál es el alcance y bajo qué condiciones?

El alcance del proyecto incluye el desarrollo completo de la plataforma para **Error Encoder**, desde la toma de requisitos hasta el despliegue final. Las condiciones son cumplir los plazos establecidos, garantizar la calidad del código y asegurar la satisfacción del cliente final.

¿Dónde se implementará la solución?

La solución será implementada como una plataforma web accesible desde cualquier navegador estándar, brindando a los clientes acceso fácil desde cualquier dispositivo conectado a Internet.

¿Quiénes están involucrados?

Los principales involucrados son el equipo de desarrollo de **Error Encoder**, los usuarios finales que utilizarán la plataforma (clientes y proveedores), y socios estratégicos que puedan estar involucrados, como proveedores de servicios y plataformas de pago.