

Centro Universitario de Occidente

Teoría de Sistemas 2

Ing. Pedro Domingo

Primer Semestre 2024



“BARTERPLACE”



#14

201930699

Erick Daniel Morales Xicar

Pensamiento Sistémico

Marco Teórico:

1. Concepto de actividad económica:

La actividad económica es cualquier proceso que implique la producción, distribución y consumo de bienes y servicios para satisfacer las necesidades de las personas. Abarca una amplia gama de actividades, desde la agricultura y la industria hasta el comercio y los servicios profesionales.

2. El comercio como actividad económica:

El comercio es la rama de la actividad económica que se centra en el intercambio de bienes y servicios entre diferentes agentes económicos. Puede realizarse a nivel local, nacional o internacional, y juega un papel fundamental en el desarrollo económico de las sociedades.

3. Evolución de las formas de intercambio:

a. Trueque:

El trueque fue la forma original de comercio, donde se intercambiaban bienes y servicios directamente sin necesidad de un medio de cambio común. Este sistema era eficiente en pequeñas comunidades, pero se volvió complejo a medida que las sociedades crecían y se diversificaron.

b. Moneda:

La invención de la moneda facilitó el intercambio al proporcionar un medio de cambio común y una unidad de cuenta. La moneda permitió a las personas comprar y vender bienes y servicios sin tener que encontrar un intercambio directo, y simplificó el cálculo de precios y valores.

c. Criptomoneda:

Las criptomonedas son una forma digital de moneda que utiliza la criptografía para asegurar las transacciones y controlar la creación de nuevas unidades. Son un desarrollo reciente y aún están en evolución, pero tienen el potencial de revolucionar el sistema monetario global.

4. Funciones del dinero:

El dinero cumple con tres funciones principales:

- Medio de cambio: Facilita el intercambio de bienes y servicios.
- Unidad de cuenta: Permite medir el valor de los bienes y servicios.
- Depósito de valor: Permite almacenar riqueza para el futuro.

5. Sistemas de comercio electrónico

El comercio electrónico es la compra y venta de bienes y servicios a través de internet. Se ha convertido en una forma importante de comercio en la economía global, y requiere de sistemas que brinden seguridad, eficiencia y confianza a las transacciones.

Objetivo del proyecto y contenido de la aplicación web

El objetivo del proyecto es desarrollar una plataforma web integral que facilite la compra, venta y participación voluntaria mediante una moneda virtual o simplemente un trueque en el contexto de transacciones locales. La aplicación permitirá a los usuarios realizar transacciones comerciales y ofrecer servicios voluntarios utilizando una moneda interna. Además, se integrará la posibilidad de realizar pagos con moneda local para ampliar la accesibilidad y utilidad del sistema.

- Desarrollo Frontend (Angular):
 - Diseñar una interfaz de usuario intuitiva y atractiva.
 - Implementar páginas y componentes para las funciones de compra, venta, voluntariado y trueque.
 - Garantizar la compatibilidad con cualquier dispositivo, mediante una aplicación web.
- Desarrollo Backend (Laravel):
 - Establecer una arquitectura sólida para el backend.
 - Implementar lógica de negocio para las transacciones de compra, venta y voluntariado.
 - Crear API seguras y eficientes para la comunicación entre el frontend y el backend

- Base de Datos (Mysql/PostgreSql):
 - Diseñar y crear una estructura de base de datos eficiente para almacenar información de usuarios, transacciones y productos/servicios.
 - Optimizar consultas para mejorar el rendimiento de la aplicación.
- Moneda Virtual:
 - Desarrollar un sistema de gestión de moneda virtual, incluyendo la asignación, retiro y transferencia de fondos.
 - Garantizar la seguridad y la integridad de las transacciones con la moneda.
 - Impulsar e incentivar el uso de esta moneda virtual.
- Compra y Venta:
 - Implementar funciones de publicación, búsqueda y compra de productos/servicios.
 - Integrar una pasarela de pago para transacciones en moneda local.
- Voluntariado:
 - Crear un sistema para que los usuarios ofrezcan y busquen oportunidades de voluntariado.
 - Establecer un proceso de selección y confirmación para participar como voluntario.
- Notificaciones y Mensajería:
 - Implementar un sistema de notificaciones en tiempo real para eventos importantes.
 - Desarrollar una funcionalidad de mensajería interna entre los usuarios.
- Administración y Seguridad
 - Crear un panel de administración para gestionar usuarios, productos, servicios y transacciones
 - Implementar medidas de seguridad, como encriptación de datos y protección contra ataques.
- Documentación y Soporte:
 - Crear manuales de usuario y documentación técnica.
 - Proporcionar soporte técnico y resolver problemas en tiempo hábil.

Herramientas a utilizar

Dada la información proporcionada en la toma de requerimientos, se mencionó que se contaba con un SERVIDOR XAMPP, por lo cual tomando en cuenta esto y algunos aspectos técnicos que se detallaran más adelante, concluimos en utilizar las siguientes tecnologías para la elaboración del software.

FRONTEND:

En el contexto de un software se refiere a la parte de la aplicación o sistema que interactúa directamente con los usuarios. Es la interfaz de usuario y la experiencia que tienen los usuarios al utilizar la aplicación. El frontend se encarga de presentar la información de manera visual, gestionar la interacción del usuario y enviar solicitudes al “Backend”. Por lo anterior la herramienta a utilizar será Angular:



[Documentación Angular](#)

Angular es un framework de desarrollo de aplicaciones web y móviles desarrollado y mantenido por Google. Se utiliza para construir aplicaciones del lado del cliente con funcionalidades avanzadas y complejas. Aquí hay algunas razones por las cuales angular es considerado bueno para el manejo del frontend:

- Arquitectura MVVM
- Datos Bidireccionales
- Inyección de Dependencias
- Componentes Reutilizables
- Tipado Estático con TypeScript
- Routing Integrado
- Amplia Comunidad y Soporte
- Herramientas de Desarrollo

Angular utiliza herramientas como:

- Lenguaje TypeScript: Lenguaje tipado estático a JavaScript, lo que facilita el desarrollo de aplicaciones más seguras y mantenibles.
- Node.js y npm: Node.js es usado para ejecutar el entorno de desarrollo y npm (Node Package Manager) para gestionar las dependencias y bibliotecas necesarias.
- Angular CLI (Command Line Interface)
- RxJS (Reactive Extensions for JavaScript)

Instalación:

Si se quiere instalar node, npm y Angular se necesita copiar y pegar los siguientes comandos en la terminal (Sistema Operativo basado en Linux)

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

Si se utiliza windows o otro sistema de linux, se puede abocar a la página oficial de Angular:

[Página Oficial Angular](#)

BACKEND:

Se refiere a la parte de un sistema informático que gestiona y procesa la lógica del negocio, la manipulación de datos y la interacción con la base de datos. En el desarrollo de web, el backend se encarga de manejar las operaciones del servidor y la comunicación con el frontend (Interfaz del usuario), por lo cual se utilizara Laravel:

[Documentación Laravel](#)



Laravel es un framework de desarrollo web en PHP, que se utiliza comúnmente para construir el backend de aplicaciones. Algunas razones por las que Laravel es un popular y considerado bueno para el desarrollo web incluyen:

- Elegancia y Simplicidad
- ORM Eloquent : tiene un sistema de mapeo objeto-relacional llamado Eloquent, ya que utiliza una sintaxis para la base de datos orientada a objetos
- Migraciones a la BD
- Sistema de Plantillas Blade
- Rutas y Controladores: tiene un sistema de enrutamiento simple y poderoso junto con controladores que facilitan la organización del código y la gestión de solicitudes HTTP.

Instalación:

Para instalar php y laravel deberá ejecutar los siguientes comandos en la terminal:

- `sudo apt install curl php-cli php-mbstring`
- `curl -sS https://getcomposer.org/installer -o composer-setup.php`
- `sudo php composer-setup.php --install-dir=/usr/local/bin --filename=composer`

Si se está utilizando un sistema de windows o otro sistema de linux, se puede abocar a la página oficial de Laravel:

[Página Oficial Laravel](#)

DBMS:

Un DBMS es un software que proporciona una interfaz para interactuar con una base de datos. Estos sistemas facilitan la creación, manipulación y administración de bases de datos, permitiendo a los usuarios almacenar, recuperar y gestionar eficientemente la información, y uno de los principales en la categoría de bases de datos relacionales es MySQL.



MySQL es un sistema de gestión de bases de datos relacional de código abierto, que sigue los principios de SQL (Structured Query Language), y por ende este gestor de bases de datos, posee las siguientes características:

- Es una base de datos relacional
- Basada en SQL
- Características ACID(Atomicidad, Consistencia, Aislamiento y Durabilidad)
- Soporte de Índices y Optimización de Consultas

Instalación:

Se necesitara instalar PostgreSQL, deberá ejecutar los siguientes comandos en la terminal

- `sudo apt install mysql-serve`
- `sudo systemctl status mysql`
- `sudo systemctl start mysql`

[Documentación MySQL](#)

Ejecución

Despues, para poder usar el proyecto, deberá ejecutar el Backend, y usando el siguiente comando:

- `php artisan serve`

Y para ejecutar el Frontend:

- `ng serve / ng s`

Y acceder a la BD:

- `mysql -u root -p`

Facilidad de uso y navegación:

Para asegurarnos de que los usuarios puedan encontrar fácilmente la información que necesitan, es importante que la página web sea fácil de usar y navegar. Algunas posibles estrategias que utilizamos son:

- **Diseño intuitivo:** La página web debería tener un diseño intuitivo y fácil de entender, con una navegación clara y sencilla.
- **Organización clara de la información:** La información en cada sección debería estar organizada de manera clara y coherente, con subtítulos y secciones claramente identificadas.
- **Uso de herramientas de búsqueda:** Podríamos incluir una herramienta de búsqueda en la página web, para que los usuarios puedan buscar información específica.
- **Diseño responsivo:** Es importante que la página web sea accesible y fácil de usar en los diferentes dispositivos, como computadoras de escritorio, laptops, tablets y teléfonos móviles.
- **Pruebas de usabilidad:** Una forma de asegurarte de que la página web sea fácil de usar es realizar pruebas de usabilidad con diferentes usuarios.

EDT Sobre “BarterPlace”

El EDT son los niveles de la estructura de desglose del trabajo que nos ayudan a separar las tareas según las dependencias. Como los proyectos pueden diferir en gran medida, los niveles de la EDT también serán distintos.

Definimos el primer nivel, que sería la meta del proyecto:

Desarrollar un software de comercio electrónico completo y escalable que satisfaga las necesidades de los clientes e impulse el crecimiento del negocio.

Como segundo nivel debemos definir los objetivos:

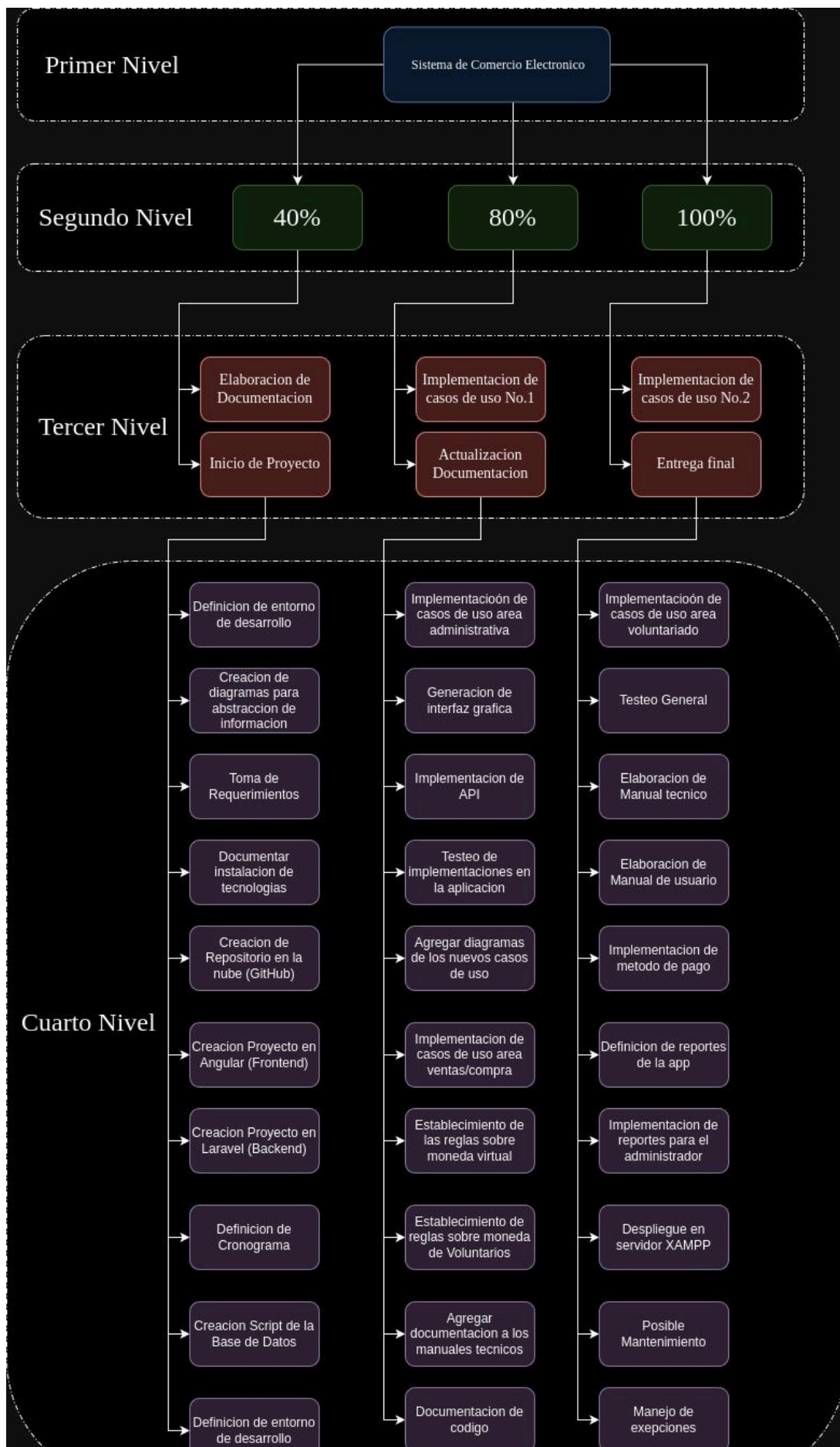
- Entrega del 40% del proyecto
- Entrega del 80% del proyecto
- Entrega del 100% del proyecto

Presupuesto:

Q100,000 - Q500,000

Elaboramos el EDT mediante un diagrama, el cual nos permitirá tener una visión gráfica sobre cómo se comportan los diferentes niveles en el EDT 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



Descripciones

Descripción Tareas en 40%

- Definición de entorno de desarrollo: **(Activa)**
 - El entorno de desarrollo para este proyecto combina las herramientas de Angular, Laravel y PostgreSQL.
- Creación de diagramas para abstracción de información: **(Activa)**
 - En el proceso de abstracción de información, se emplearán diagramas como el Diagrama de Clases para modelar las relaciones entre entidades, el Diagrama de Flujo para representar la lógica del sistema y el Diagrama de Base de Datos para visualizar la estructura de la base de datos.
- Toma de Requerimientos: **(En proceso)**
 - En la fase de toma de requerimientos, se realizará un proceso de interacción con los stakeholders para identificar y documentar de manera detallada las necesidades y expectativas del sistema.
- Documentar instalación de tecnologías: **(En proceso)**
 - La documentación de la instalación de tecnologías es esencial para proporcionar una guía clara y completa sobre cómo configurar y desplegar el entorno de desarrollo.
- Creación de Repositorio en la nube (GitHub): **(Terminada)**
 - La creación de un repositorio en la nube, como GitHub, implica establecer un espacio en línea para almacenar y gestionar el código fuente de un proyecto.
- Creación Proyecto en Angular (Frontend): **(Terminada)**
 - En la creación de un proyecto en Angular, se establece un entorno de desarrollo mediante la instalación de herramientas como Node.js y Angular CLI.
- Creación Proyecto en Laravel (Backend): **(En proceso)**
 - Se instalará el framework utilizando Composer, se configurará el entorno y se comenzará a desarrollar el backend.

- Definición de Cronograma: (En proceso)
 - La definición de un cronograma implica la planificación temporal y secuencial de actividades dentro de un proyecto. Este documento detalla las tareas, su duración, interdependencias y asignación de recursos, proporcionando un marco temporal estructurado.
- Creación Script de la Base de Datos: (En proceso)
 - La creación de un script de la base de datos involucra el desarrollo de un conjunto de instrucciones estructuradas para definir la estructura, tablas, relaciones y, en algunos casos, los datos iniciales de una base de datos.

Descripción Tareas en 80%

- Implementación de casos de uso área administrativa: (Activa)
 - La implementación de casos de uso en el área administrativa implica traducir las funcionalidades identificadas en los casos de uso en un sistema o aplicación específicos.
- Generación de interfaz gráfica: (Activa)
 - La generación de interfaz gráfica implica la creación de elementos visuales y la disposición de componentes que constituyen la parte visual de una aplicación o sistema.
- Implementación de API: (Activa)
 - Implica desarrollar el conjunto de endpoints, lógica de negocios y funcionalidades necesarias para que otras aplicaciones o servicios interactúen con el sistema.
- Testeo de implementaciones en la aplicación: (Activa)
 - Se refiere a la evaluación sistemática de las nuevas funcionalidades o cambios introducidos en el software.
- Agregar diagramas de los nuevos casos de uso: (Activa)
 - Implica la creación de representaciones visuales que describan las interacciones y relaciones entre el sistema y los actores para las funcionalidades recién implementadas.

- Implementación de casos de uso área ventas/compra:(Activa)
 - Implica la transformación de las funcionalidades identificadas en los casos de uso específicos para esta área en acciones concretas dentro de un sistema o aplicación.
- Establecimiento de las reglas sobre moneda virtual:(Activa)
 - Implica la definición e implementación de políticas y normativas que rigen el uso, intercambio y gestión de una moneda virtual en un sistema o plataforma.
- Establecimiento de reglas sobre moneda de Voluntarios:(Activa)
 - Implica la creación y aplicación de políticas que regulen el uso y gestión de una moneda específica en el contexto de programas de voluntariado.
- Agregar documentación a los manuales técnicos:(Activa)
 - Implica incorporar información detallada y clara sobre la implementación, configuración y mantenimiento de aspectos técnicos en un sistema o proyecto.
- Documentación de código:(Activa)
 - Implica agregar comentarios y explicaciones claras dentro del código fuente de un programa o aplicación para describir su lógica, funcionamiento y propósito.

Descripción Tareas en 100%

- Implementación de casos de uso área voluntariado:(Activa)
 - Implica traducir las interacciones y funcionalidades identificadas en los casos de uso en acciones concretas dentro del sistema.
- Testeo General:(Activa)
 - Implica llevar a cabo evaluaciones exhaustivas de un sistema o aplicación para garantizar su correcto funcionamiento en diversos escenarios y condiciones.
- Elaboración de Manual tecnico:(Activa)
 - Este manual proporciona información técnica esencial para desarrolladores, administradores y cualquier persona involucrada en el ciclo de vida del producto.

- Elaboración de Manual de usuario: (Activa)
 - Este manual está diseñado para usuarios finales y aborda aspectos como la navegación en la interfaz, la ejecución de funciones, la interpretación de mensajes de error y cualquier otra información relevante para la interacción efectiva con el software.
- Implementación del método de pago: (Activa)
 - implica la integración y desarrollo de funcionalidades específicas dentro de un sistema o plataforma para permitir transacciones financieras.
- Definición de reportes de la app: (Activa)
 - Esto incluye determinar los tipos de reportes, los parámetros de filtrado, el diseño visual y la frecuencia de actualización.
- Implementación de reportes para el administrador: (Activa)
 - Esto incluye la generación de reportes personalizados, la implementación de gráficos y tablas relevantes, así como la capacidad de filtrar y exportar datos.
- Despliegue en servidor XAMPP: (Activa)
 - Esto generalmente implica copiar los archivos del proyecto al directorio adecuado en el servidor XAMPP, configurar la base de datos si es necesario, y asegurarse de que todos los servicios necesarios estén activos.
- Posible Mantenimiento: (Activa)
 - Esto puede incluir la aplicación de parches de seguridad, la solución de problemas, la optimización del rendimiento y la introducción de nuevas funcionalidades o mejoras.
- Manejo de excepciones: (Activa)
 - Implica la implementación de mecanismos para gestionar situaciones inesperadas o errores durante la ejecución de un programa.

Diccionario

El proyecto contará con 4 programadores:

- Desarrollador Backend
- Desarrollador Frontend
- DBA (Administrador Base de Datos)
- Desarrollador Full Stack

Primera Entrega:

Nombre	A cargo	Presupuesto	Tiempo
Definición de entorno de desarrollo	Todos los programadores	Q. 0.00	2-3 Días hábiles
Creación de diagramas para abstracción de información	Todos los programadores	Q2500	5 Días hábiles
Toma de Requerimientos	Desarrollador Full Stack	Q.0.00	2 Días (pueden ser en distintas fechas)
Documentar instalación de tecnologías	Todos los programadores	Q2500	1 Dia habil
Creación de Repositorio en la nube	Desarrollador Full Stack	Q.0.00	1 Dia habil
Creación Proyecto en Angular (Frontend)	Desarrollador Frontend	Q.0.00	1 Dia habil
Creación Proyecto en Laravel (Backend)	Desarrollador Backend	Q.0.00	1 Dia habil
Definición de Cronograma	Desarrollador Full Stack	Q1,000	1-2 Días hábiles
Creacion Script de la Base de Datos	DBA	Q5,000	3 Días Hábiles

Segunda Entrega:

Nombre	A cargo	Presupuesto	Tiempo
Implementación de casos de uso área administrativa	Desarrollador Frontend Desarrollador Backend	Q.10,000	7-9 Dias Habiles
Generación de interfaz gráfica	Desarrollador Frontend	Q5,000	7-9 Dias habiles
Implementación de API	Desarrollador Backend	Q7,000	5 Días hábiles
Testeo de implementaciones en la aplicación	Desarrollador Full Stack	Q.5,000	3 Días hábiles
Agregar diagramas de los nuevos casos de uso	Desarrollador Full Stack	Q.0.00	3 Días hábiles
Implementación de casos de uso área ventas/compra	Desarrollador Frontend Desarrollador Backend DBA	Q.30,000	9-12 Días hábiles
Establecimiento de las reglas sobre moneda virtual:	Desarrollador Full Stack	Q.0.00	1 Día
Establecimiento de reglas sobre moneda de Voluntarios	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	Duración del Proyecto

Tercera Entrega:

Nombre	A cargo	Presupuesto	Tiempo
Implementación de casos de uso área voluntariado	Desarrollador Frontend Desarrollador Backend DBA	Q.8,000	3-5 Días hábiles
Testeo General	Desarrollador Full Stack	Q.12,000	5-7 Días hábiles
Elaboración de Manual tecnico	Todos los programadores	Q.0.00	3-5 Días hábiles
Elaboración de Manual de usuario	Desarrollador Full Stack	Q1,000	2-4 Días hábiles
Implementación del método de pago	Desarrollador Frontend Desarrollador Backend DBA	Q.5,000	2-4 Días hábiles
Definición de reportes de la app	Desarrollador Full Stack	Q2,500	1-2 Días hábiles
Implementación de reportes para el administrador	Desarrollador Frontend Desarrollador Backend DBA	Q.15,000	3-5 Días hábiles
Despliegue en servidor XAMPP	Desarrollador Full Stack	Q.5,000	2-3 Días hábiles
Posible Mantenimiento	Desarrollador Full Stack	Q.25,000	7-9 Dias habiles
Manejo de excepciones	Todos los programadores	Q.5,000	3-5 Días hábiles

Presupuesto total: Q146,500 , el cual puede variar dependiendo de cambios luego de aprobación del proyecto.

Cronograma BarterPlace

Fechas Importantes:

Elaboración del Acta de Constitución del Proyecto	22/02/2024
Primer Entregable (40%)	20/03/2024
Segundo Entregable (80%)	11/04/2024
Tercer Entregable (100%)	18/04/2024

Definición de entorno de desarrollo:

Se refiere al proceso de seleccionar y configurar las herramientas, recursos y tecnologías que se utilizarán para crear, probar y ejecutar un software.

Semana 3 (Febrero) 23/02/2024 - 01/03/2024

Diseño del Proyecto:

Se refiere al proceso de creación de la documentación de las tecnologías a utilizar, definición y a su vez diagramas que ayudan a ver la interacción del usuario con el sistema.

Semana 3 (Febrero) 23/02/2024 - 01/03/2024

Semana 1 (Marzo) 04/03/2024 - 08/03/2024

Ejecución:

Se refiere al proceso de la creación de los módulos (Admin, Compras, Ventas y Voluntariado)

Semana 1,2,3,4 (Marzo) 04/03/2024 - 29/03/2024

Semana 1,2 (Abril) 01/04/2024 - 12/04/2024

Realización de tests:

Es un proceso fundamental para garantizar la calidad del software. Consiste en ejecutar el software en diferentes escenarios para identificar y corregir errores, así como para verificar que cumple con los requisitos establecidos.

Semana 1,2 (Abril) 01/04/2024 - 12/04/2024

Reportes:

Consiste en la elaboración del módulo de reportes que serán accedados por el administrador del sistema.

Semana 2 (Abril) _____ 08/04/2024 - 12/04/2024

Actualización de documentación

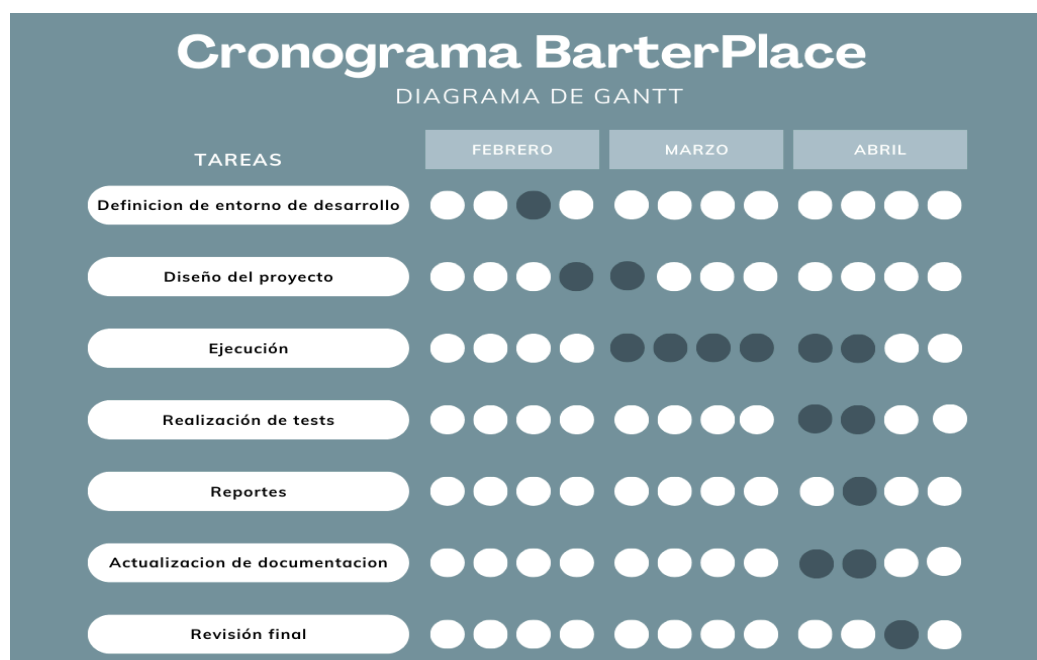
Consiste en modificar y mejorar la documentación existente para reflejar cambios en la información, procesos o sistemas

Semana 1,2 (Abril) _____ 01/04/2024 - 12/04/2024

Revisión final:

Consiste en el proceso donde se evaluará el proyecto en general, y se considera para poder arreglar posibles fallas o terminar algún módulo incompleto.

Semana 3 (Abril) _____ 15/04/2024 - 18/04/2024



Casos de uso alto nivel

Caso de uso	Registrarse	CU 001
Actor(es)	Usuario (Comprador,Vendedor, Voluntario)	
Poscondición	El usuario se registra en el sistema	
Resumen	El usuario(ya sea Comprador,Vendedor, Voluntario) ingresa sus credenciales.	
Tipo	Primario	

Caso de uso	Inicio de Sesión	CU 002
Actor(es)	Usuario (Comprador,Vendedor, Voluntario), Administrador	
Poscondición	El usuario ingresa al sistema	
Resumen	El usuario(ya sea Comprador,Vendedor, Voluntario y Administrador) accede al sistema a través de la página por medio de su usuario y contraseña.	
Tipo	Primario	

Caso de uso	Ver publicaciones sin ser usuario	CU 003
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	
Tipo	Primario	

Caso de uso	Ver publicaciones	CU 004
Actor(es)	Usuario (Visitante, Comprador)	
Poscondición	El usuario verá publicaciones existentes	
Resumen	El usuario verá “todas” las publicaciones activas	
Tipo	Primario	

Caso de uso	Ver ventas	CU 005
Actor(es)	Usuario (Vendedor, Comprador, Voluntario)	
Poscondición	El usuario verá publicaciones tipo “Venta” existentes	
Resumen	El usuario verá “todas” las publicaciones tipo venta activas	
Tipo	Secundario	

Caso de uso	Ver voluntariados	CU 006
Actor(es)	Usuario (Vendedor, Comprador, Voluntario)	
Poscondición	El usuario verá publicaciones tipo “Voluntario” existentes	
Resumen	El usuario verá “todas” las publicaciones tipo voluntariado activas	
Tipo	Secundario	

Caso de uso	Subir Imagen/es	CU 007
Actor(es)	Usuario Vendedor	
Poscondición	El usuario vendedor puede subir imagenes	
Resumen	El usuario vendedor puede subir imágenes en el apartado de “subir publicación”, sobre el artículo a publicar	
Tipo	Primario	

Caso de uso	Guardar Publicación	CU 008
Actor(es)	Usuario vendedor	
Poscondición	El usuario vendedor puede guardar la publicación creada	
Resumen	El usuario vendedor en el sistema puede terminar de crear su publicación luego de subir los requerimientos.	
Tipo	Primario	

Caso de uso	Presionar Imagen	CU 009
Actor(es)	Usuario (Comprador, Voluntario)	
Poscondición	El usuario (comprador, voluntario) puede presionar las imágenes o publicaciones	
Resumen	El estudiante presiona la publicación, las cuales servirá para poder más información	
Tipo	Primario	

Caso de uso	Descripción Publicación	CU 010
Actor(es)	Usuario Vendedor	
Poscondición	El usuario vendedor deberá añadir una descripción de su publicación creada	
Resumen	El usuario vendedor añade la descripción de la publicación que ha creado para que los demás usuarios puedan verlo.	
Tipo	Primario	

Caso de uso	Agregar Precio	CU 011
Actor(es)	Usuario Vendedor	
Poscondición	El usuario vendedor deberá añadir el precio	
Resumen	El usuario vendedor deberá añadir el precio del articulo a vender	
Tipo	Primario	

Caso de uso	Definir Voluntariado	CU 012
Actor(es)	Usuario Vendedor	
Poscondición	El usuario vendedor deberá especificar qué es voluntariado	
Resumen	El usuario vendedor deberá añadir la recompensa que se tendrá por asistir al voluntariado.	
Tipo	Primario	

Caso de uso	Autorización de Recompensa	CU 013
Actor(es)	Administrador	
Poscondición	El administrador deberá autorizar la publicación	
Resumen	El administrador autoriza la publicación para que esta pueda ser publicada	
Tipo	Primario	

Caso de uso	Dar Recompensa	CU 014
Actor(es)	Usuario Vendedor	
Poscondición	El usuario vendedor dará la recompensa	
Resumen	El usuario vendedor deberá “cerrar el voluntariado”, y se le mostrará una lista para ingresar a los usuarios que le dará la recompensa	
Tipo	Primario	

Caso de uso	Contestar cuestionario	CU 015
Actor(es)	Usuario Comprador	
Poscondición	El usuario contestara el cuestionario	
Resumen	Él responde, para poder calificar al usuario vendedor	
Tipo	Primario	

Caso de uso	Ingresar Dinero	CU 016
Actor(es)	Usuario	
Poscondición	El usuario podrá ingresar dinero a su cuenta	
Resumen	El usuario subirá imagen de su depósito, para que el administrador pueda autorizar	
Tipo	Primario	

Caso de uso	Retiro Dinero	CU 017
Actor(es)	Usuario	
Poscondición	El usuario podrá retirar dinero a su cuenta	
Resumen	El usuario retira dinero mediante transferencia a su cuenta de banco la cual deberá ser autorizada por el administrador	
Tipo	Primario	

Caso de uso	Cambio de Moneda	CU 018
Actor(es)	Usuario	
Poscondición	El usuario podrá cambiar de moneda	
Resumen	El usuario cambia su dinero local por la moneda del sistema	
Tipo	Primario	

Caso de uso	Cambio Dinero	CU 019
Actor(es)	Usuario	
Poscondición	El usuario podrá cambiar dinero	
Resumen	El usuario podrá cambiar dinero obtenido en moneda virtual y pasarlo a la moneda local	
Tipo	Primario	

Caso de uso	Autorizar Publicación	CU 020
Actor(es)	Administrador	
Poscondición	El admin podrá autorizar publicaciones	
Resumen	El administrador deberá autorizar las publicaciones de los usuarios esto para evitar publicaciones erróneas	
Tipo	Primario	

Caso de uso	Autorizar Depósito	CU 021
Actor(es)	Administrador	
Poscondición	El admin podrá autorizar depósitos	
Resumen	El administrador deberá autorizar los depósitos lo cual deberá verificar los datos	
Tipo	Primario	

Caso de uso	Realizar Reporte	CU 022
Actor(es)	Usuario (Vendedor, Comprador, Voluntario)	
Poscondición	El usuario hará reportes	
Resumen	El usuario si piensa que es coherente podrá reportar la publicación mediante un formulario	
Tipo	Primario	

Caso de uso	Evaluación Reporte	CU 023
Actor(es)	Administrador	
Poscondición	El administrador revisará los reportes	
Resumen	El administrador evaluará los reportes para decidir si bloquear el usuario	
Tipo	Primario	

Caso de uso	Finalizar Venta	CU 024
Actor(es)	Usuario Vendedor	
Poscondición	El usuario vendedor finaliza la venta	
Resumen	El usuario deberá finalizar la venta, y especificar mediante un formulario detalles de la venta que el usuario comprador deberá validar	
Tipo	Primario	

Caso de uso	Chat personal	CU 025
Actor(es)	Usuario (Vendedor, Voluntario, Comprador)	
Poscondición	El usuario podrá contar con un chat	
Resumen	El usuario utilizara el chat para poder comunicarse y solicitar más detalles	
Tipo	Primario	

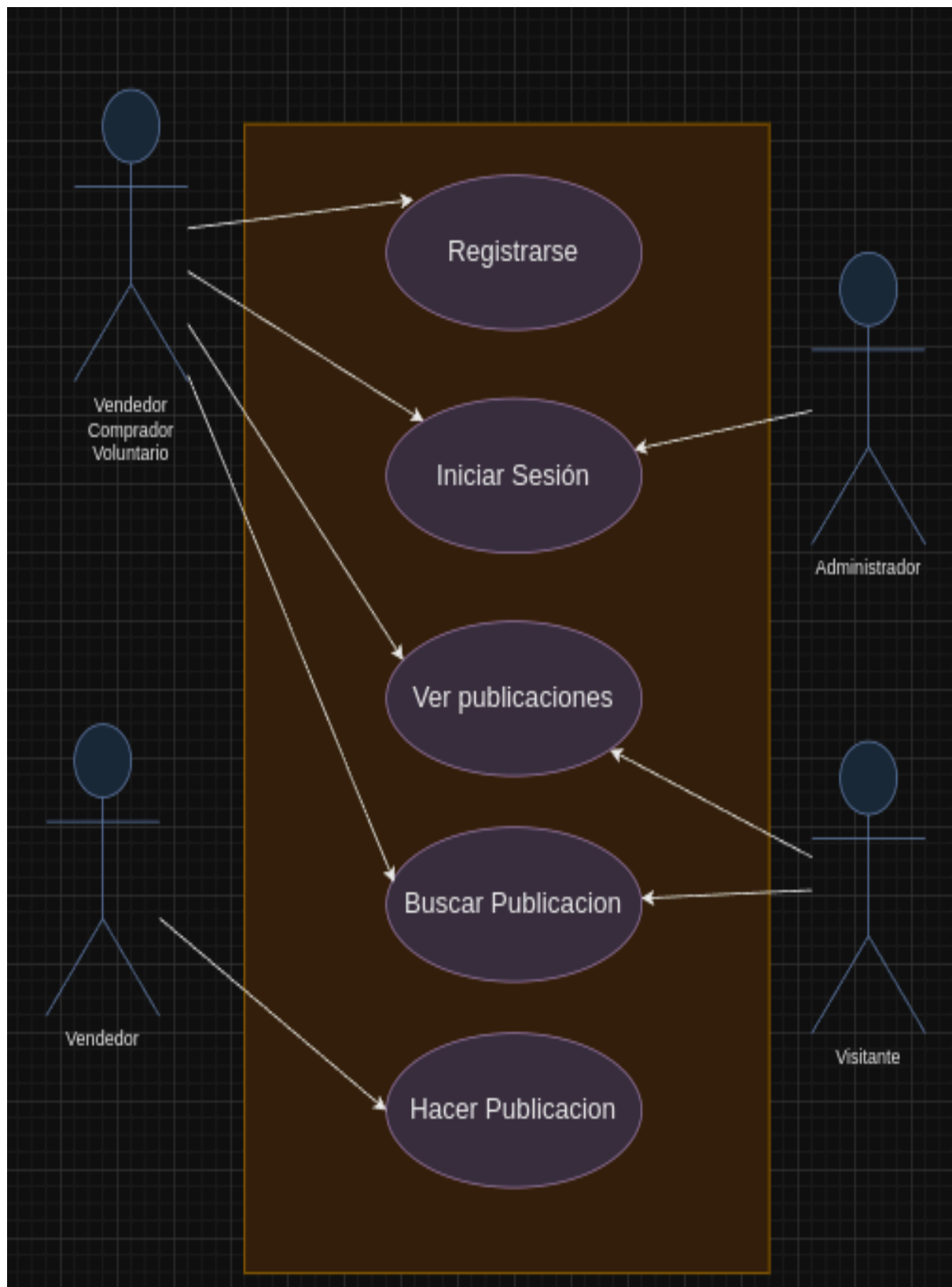
Caso de uso	Editar Perfil	CU 026
Actor(es)	Usuario (Vendedor, Voluntario, Comprador)	
Poscondición	El usuario podrá editar su perfil	
Resumen	El usuario podrá añadir o quitar información sobre contacto, ubicación, entre otros.	
Tipo	Primario	

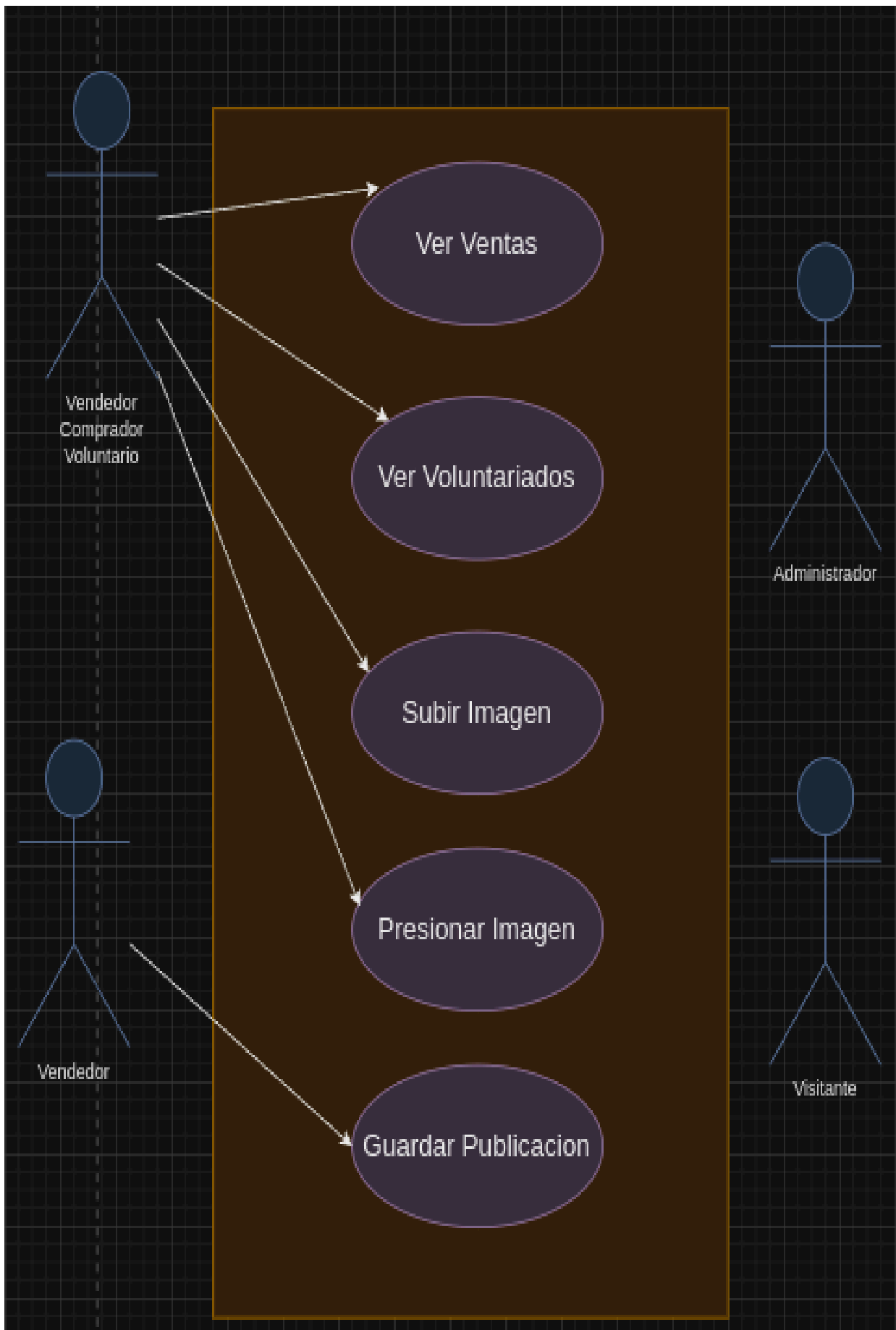
Caso de uso	Editar Publicación	CU 027
Actor(es)	Usuario (Vendedor)	
Poscondición	El usuario podrá editar su publicación	
Resumen	El usuario podrá editar su publicación, campos como el precio, descripcion	
Tipo	Primario	

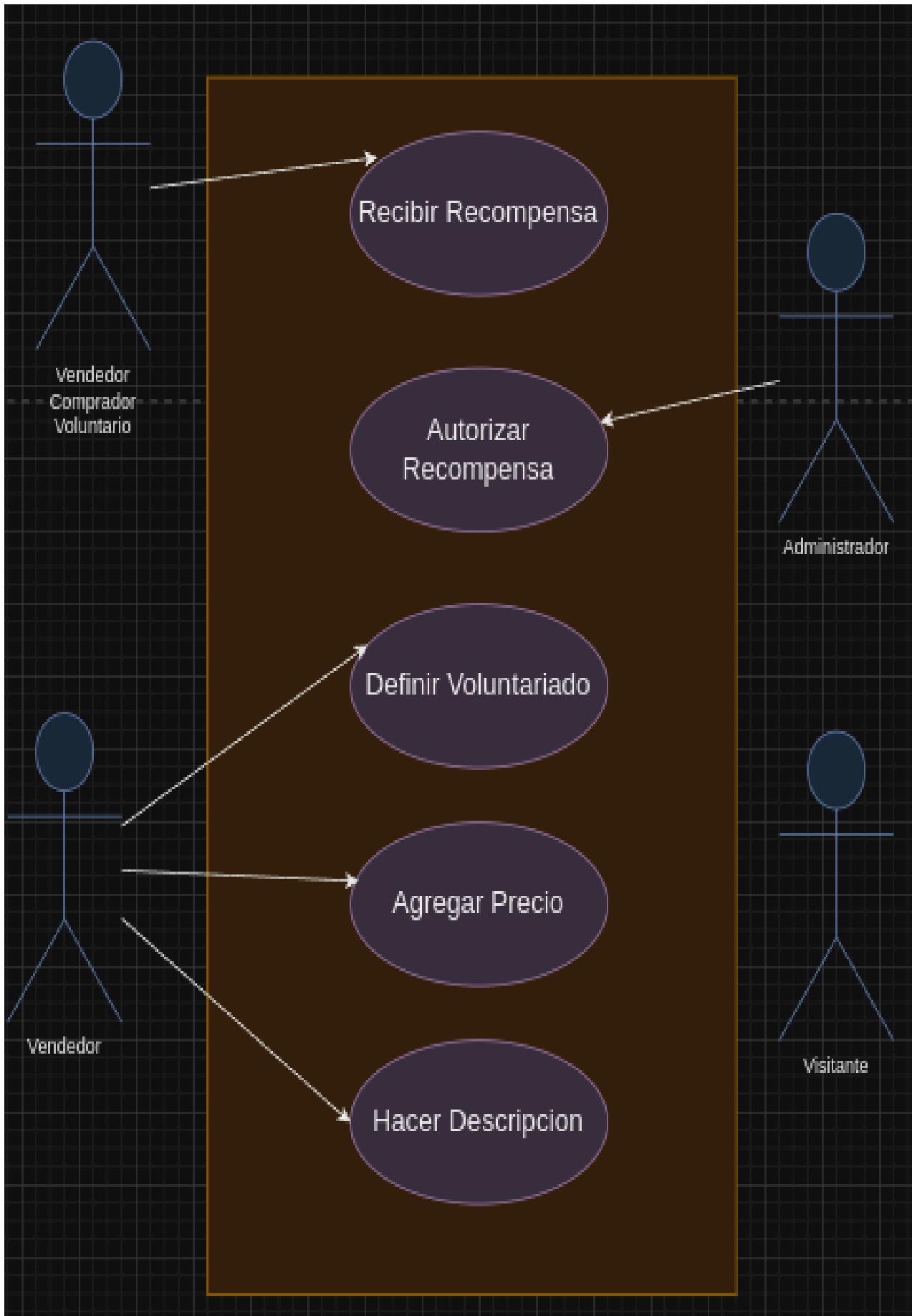
Caso de uso	Comentar Publicación	CU 028
Actor(es)	Usuario (Voluntario, Comprador)	
Poscondición	El usuario podrá comentar la publicación	
Resumen	El usuario podrá comentar la publicación, esto para que otros usuarios puedan revisar los comentarios	
Tipo	Primario	

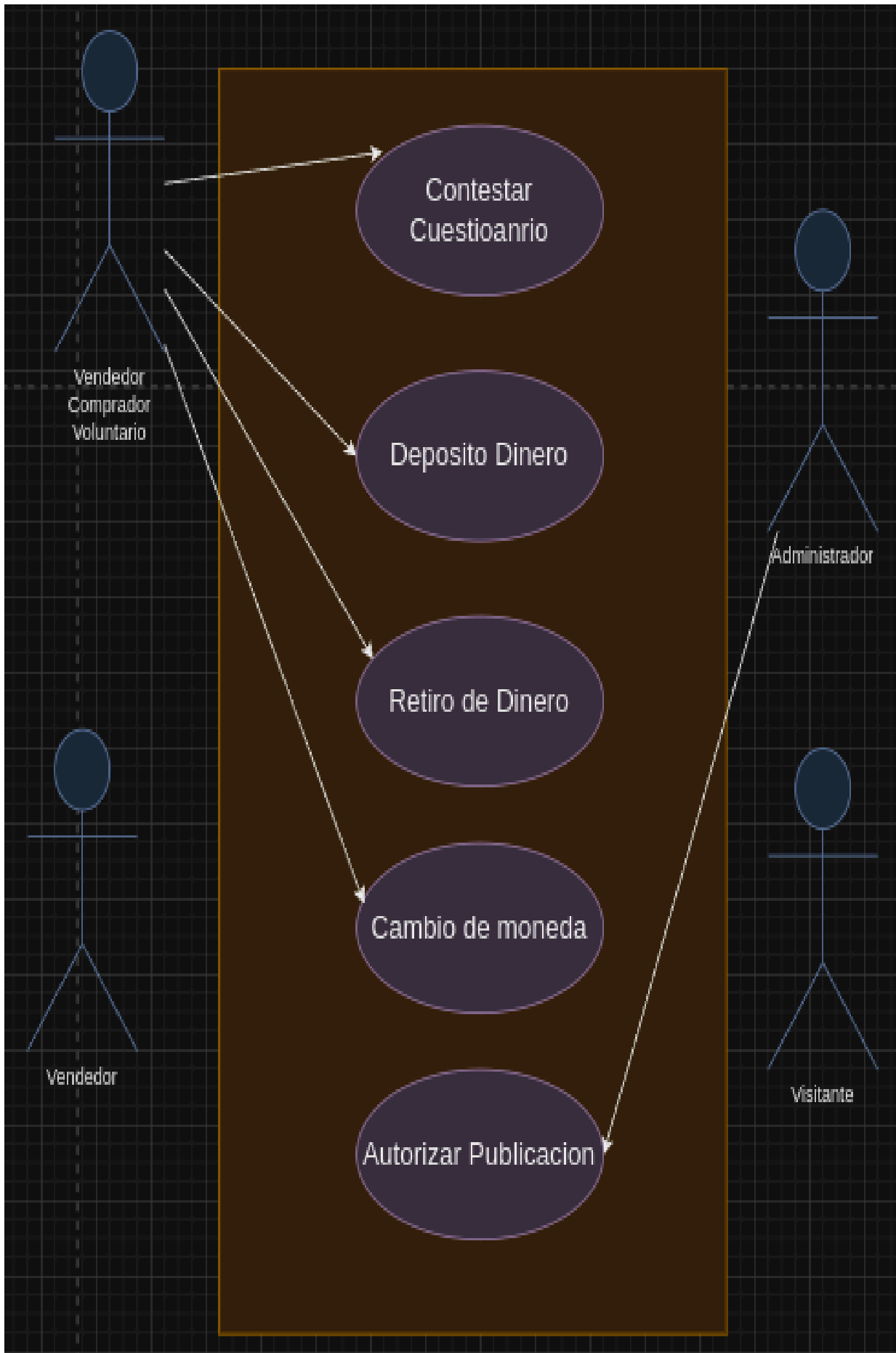
Caso de uso	Ver Mis Publicaciones	CU 029
Actor(es)	Usuario (Vendedor)	
Poscondición	El usuario podrá visualizar su publicación	
Resumen	El usuario podrá visualizar sus publicaciones, donde seguidamente podrá editarlas o podrá administrarlas	
Tipo	Primario	

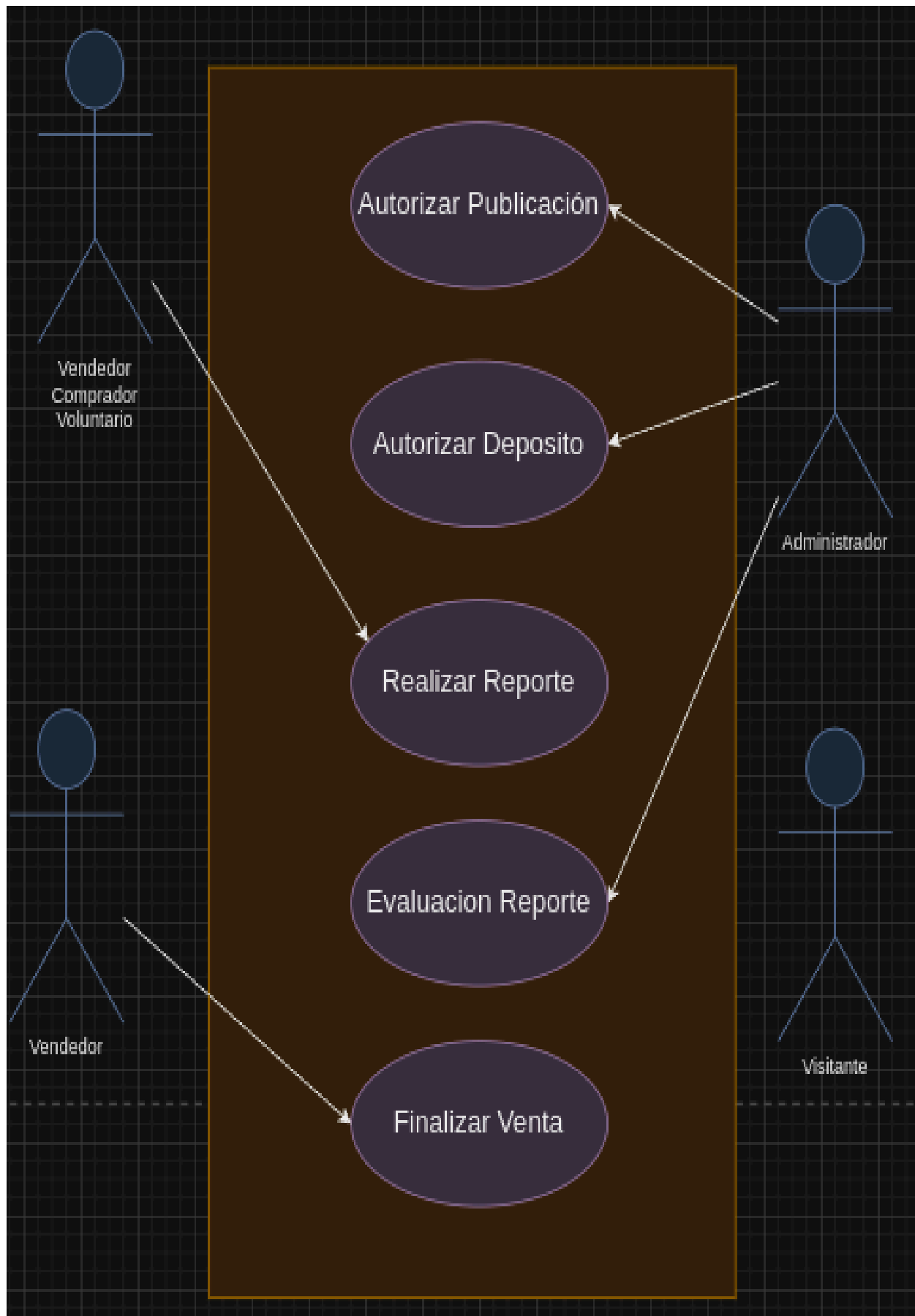
Casos de Uso(Diagramas)

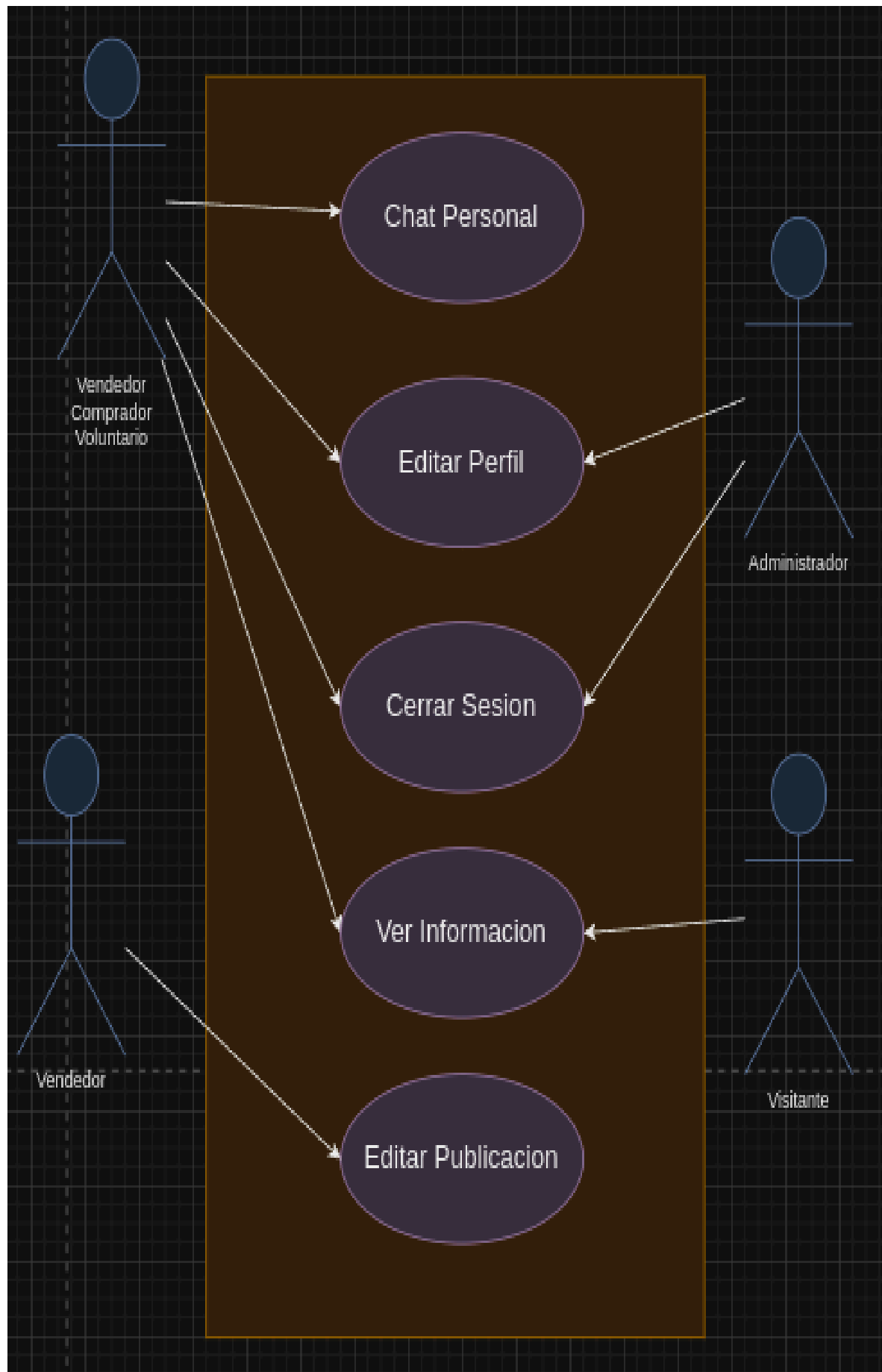




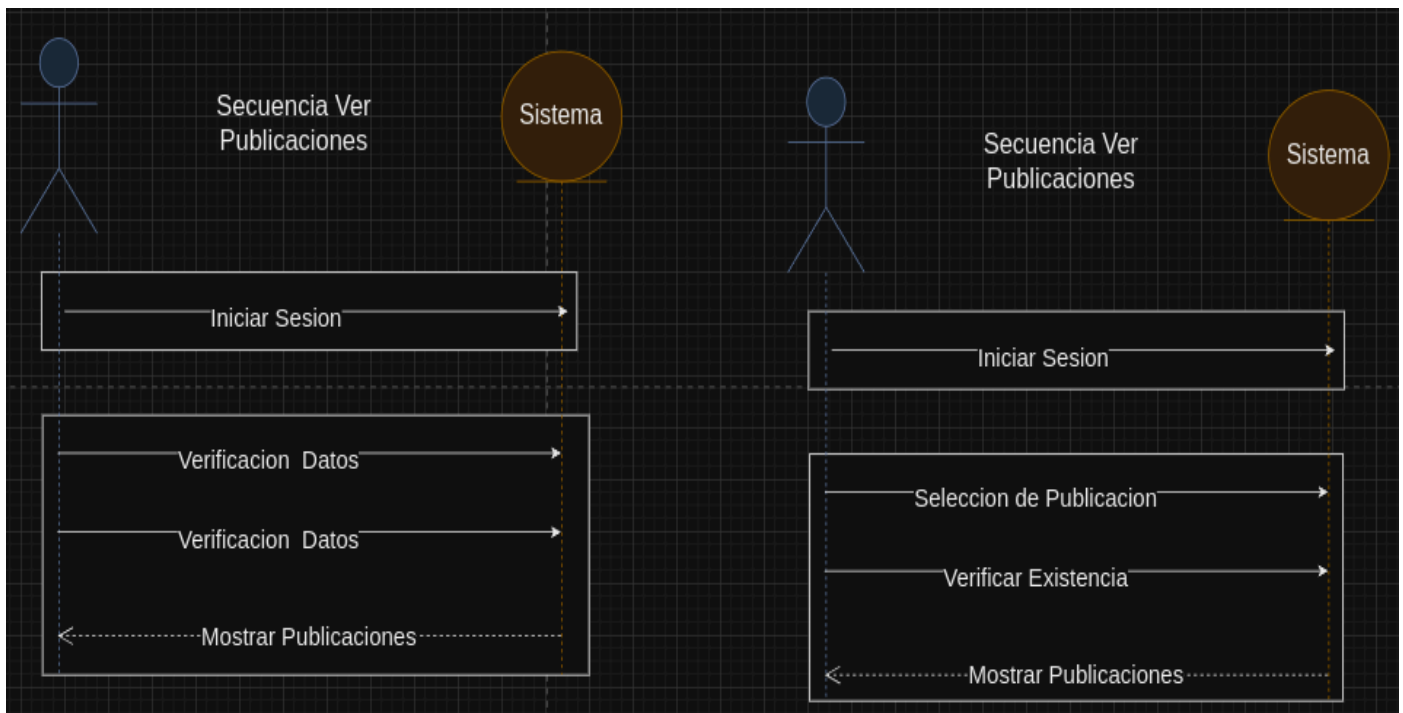
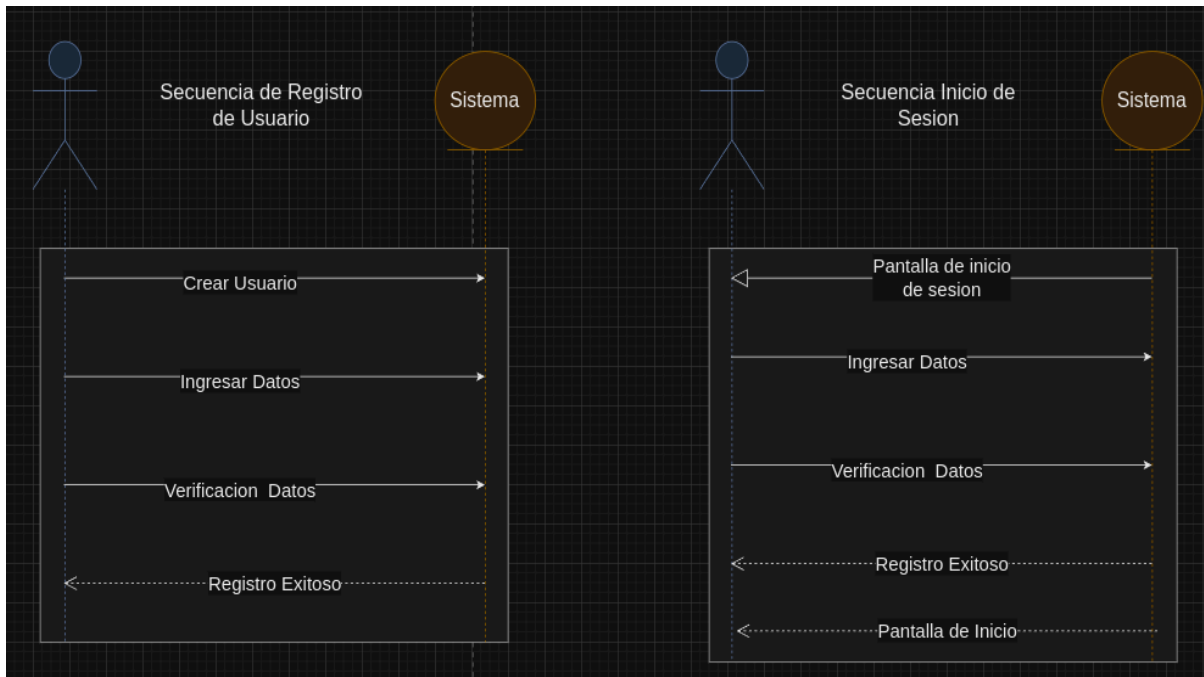


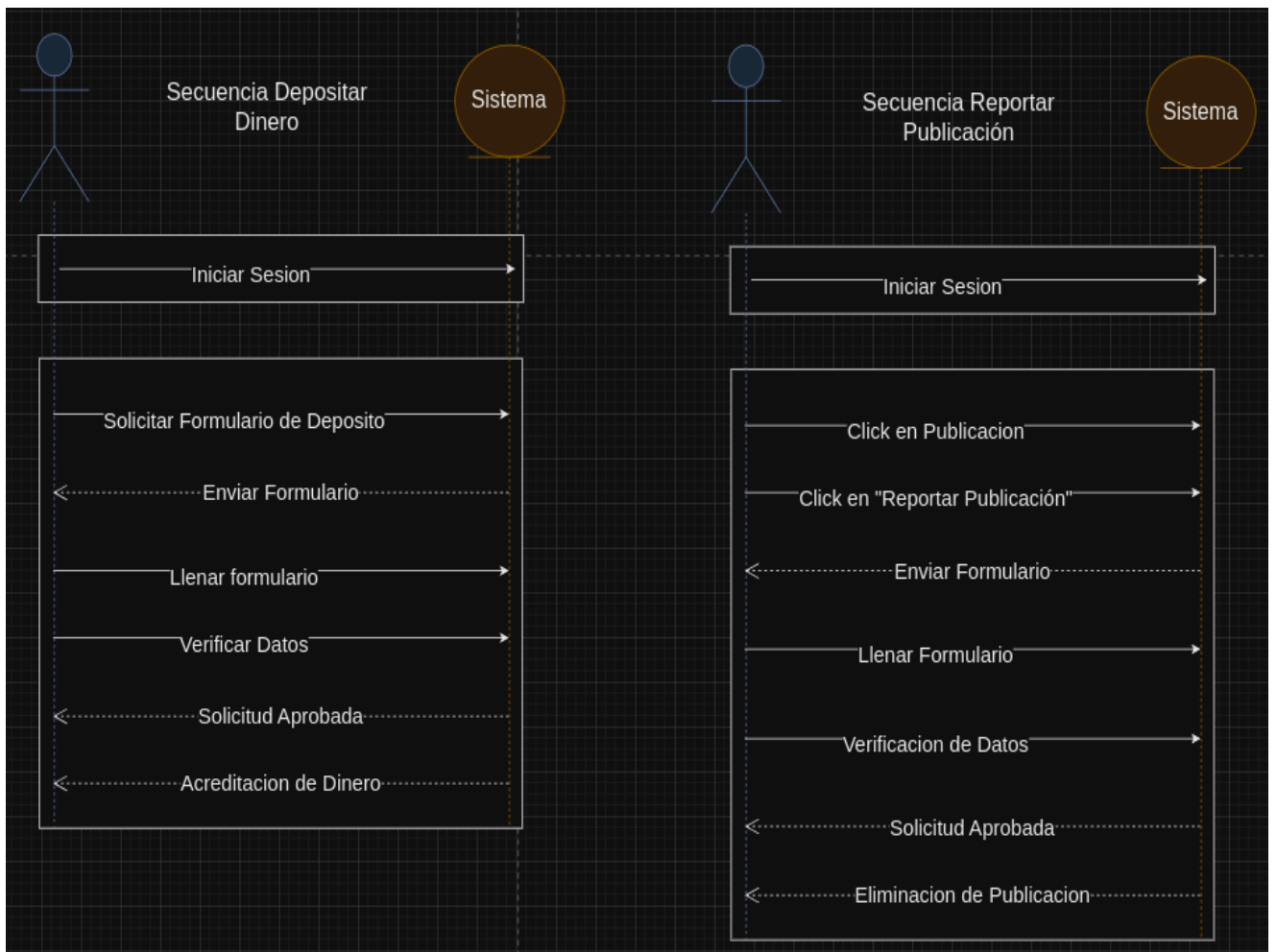
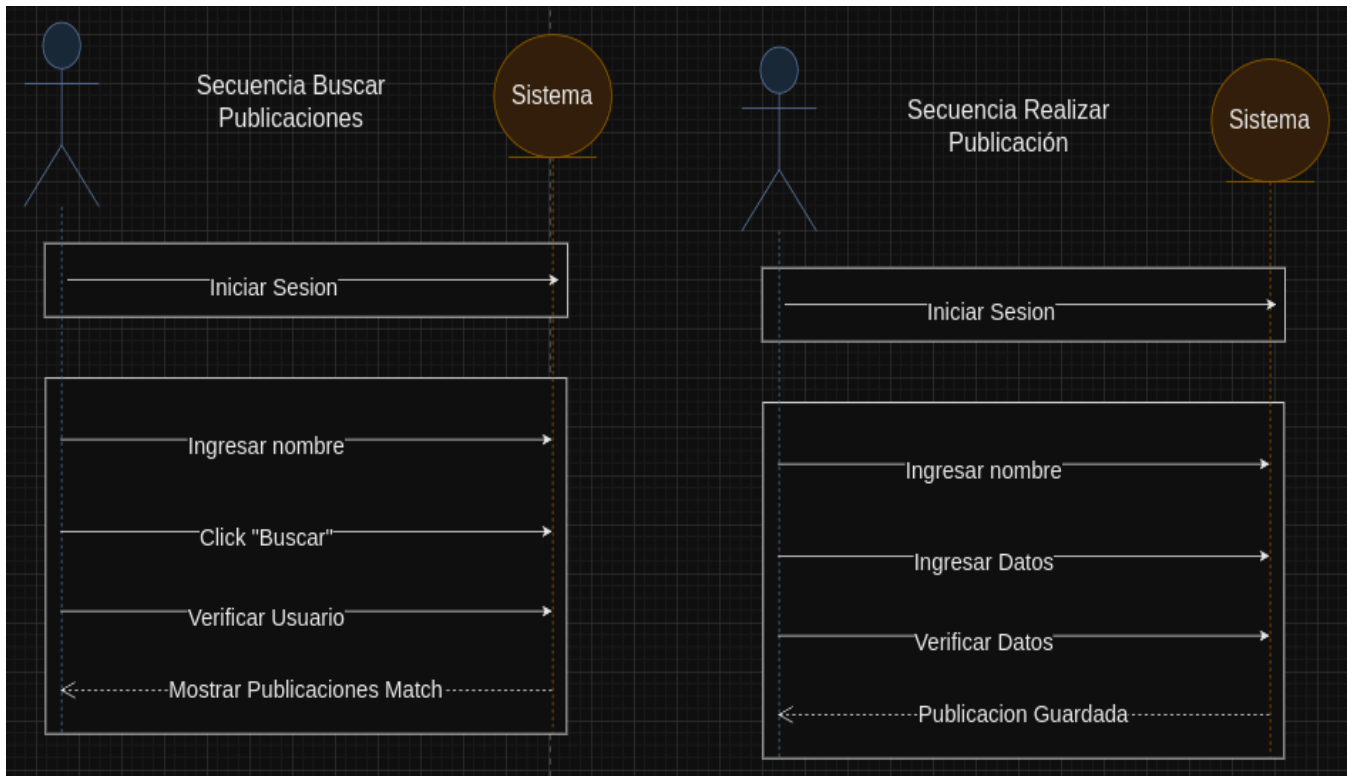




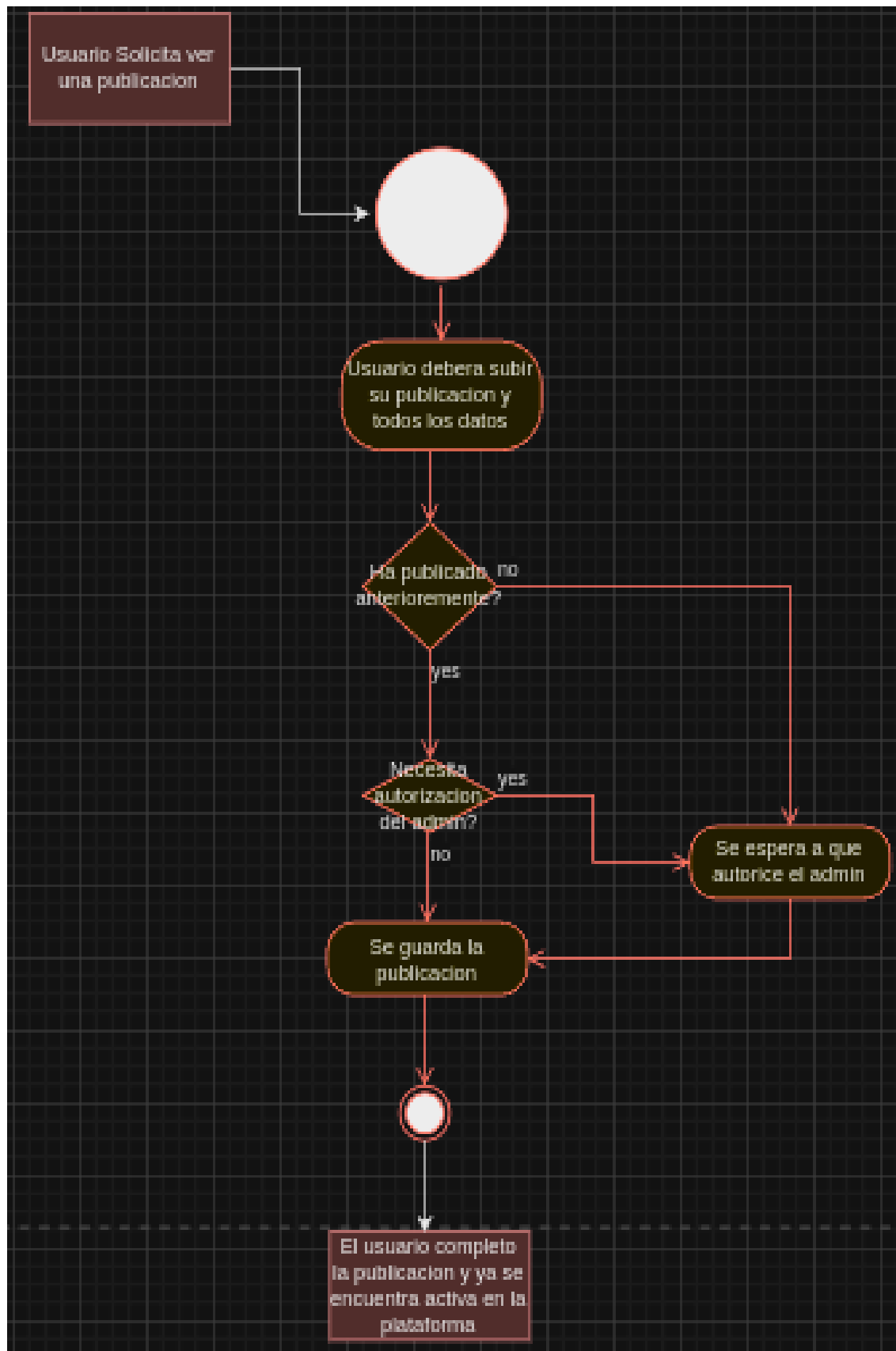


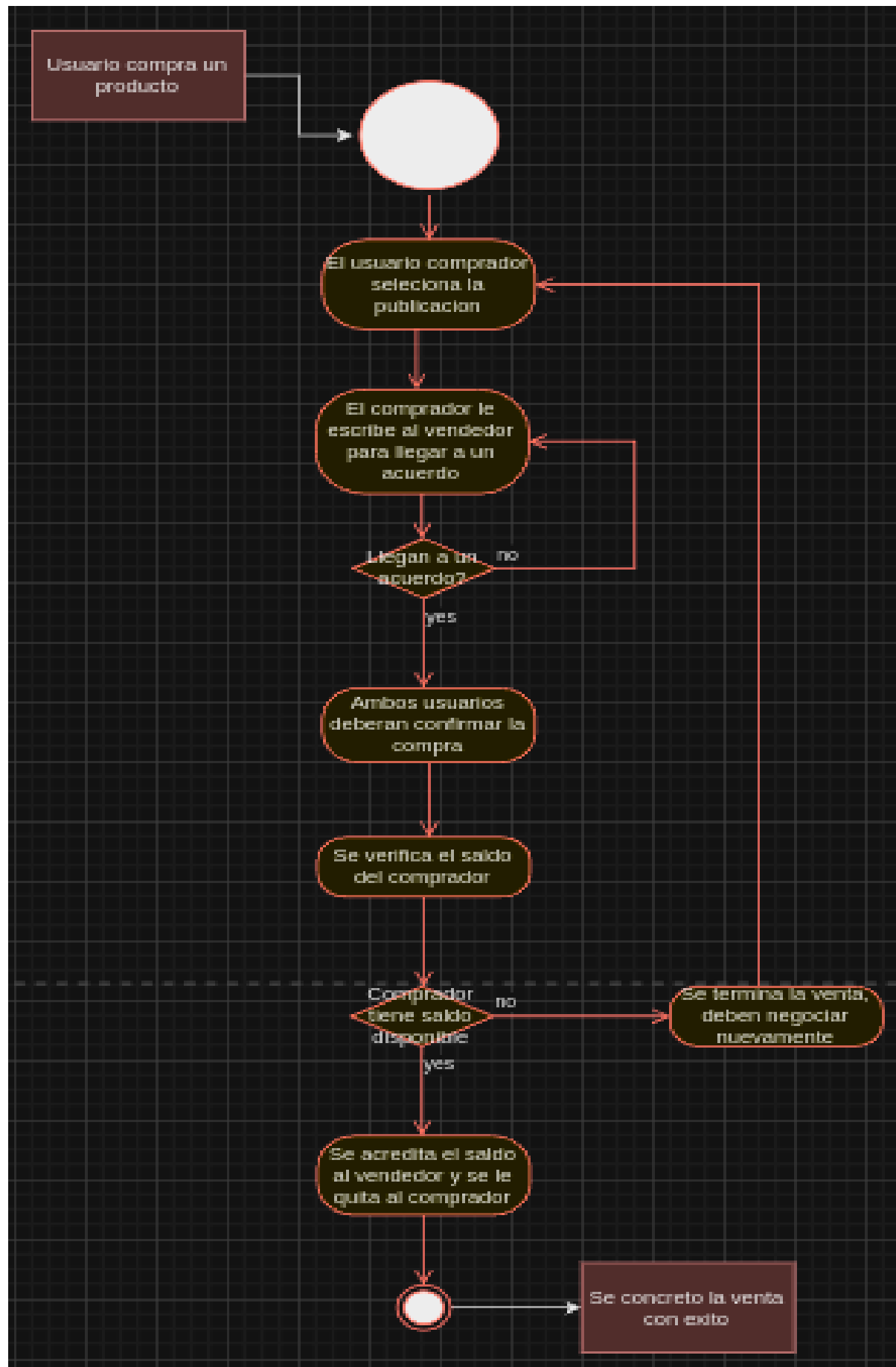
Diagramas de Secuencia





Diagramas de Actividades





Organigrama

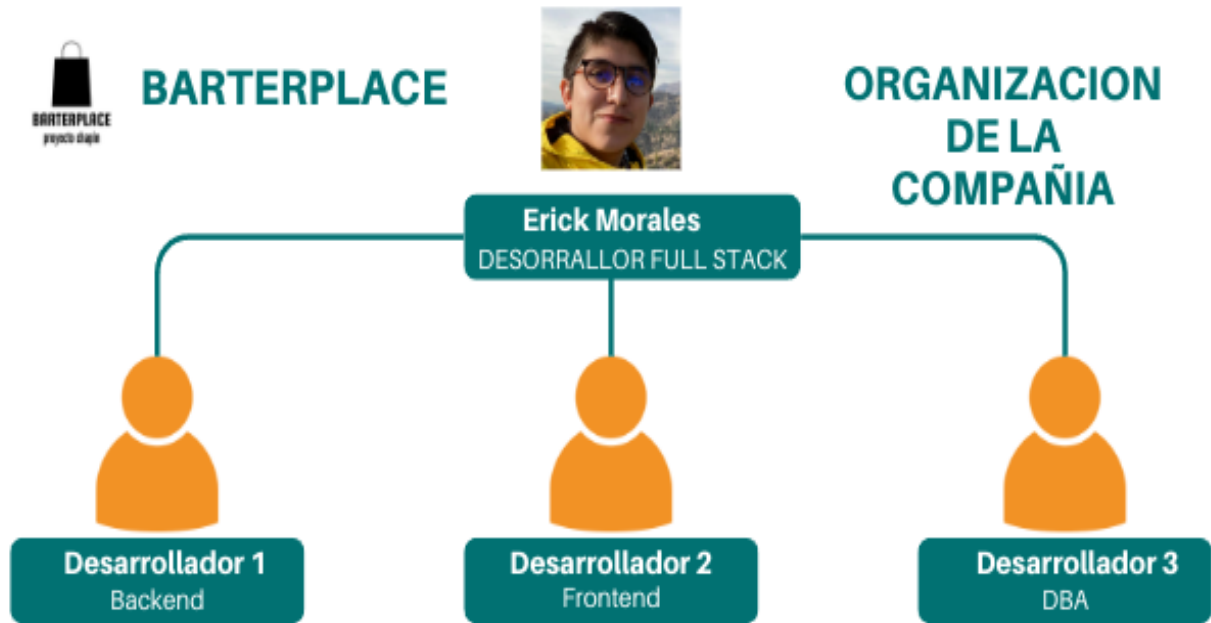


Diagrama Entidad Relación (Notación Peter Chen)

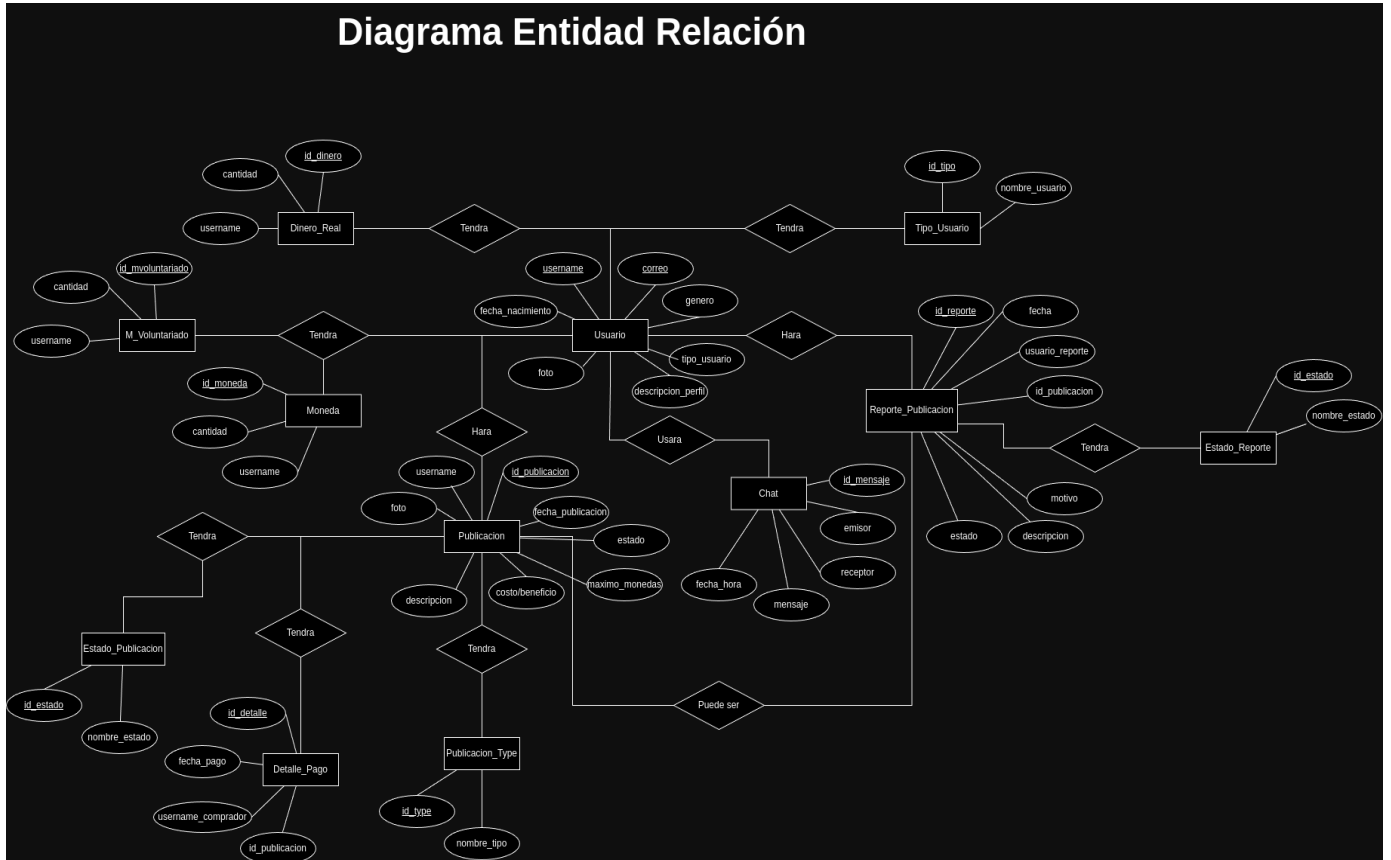


Diagrama de Despliegue

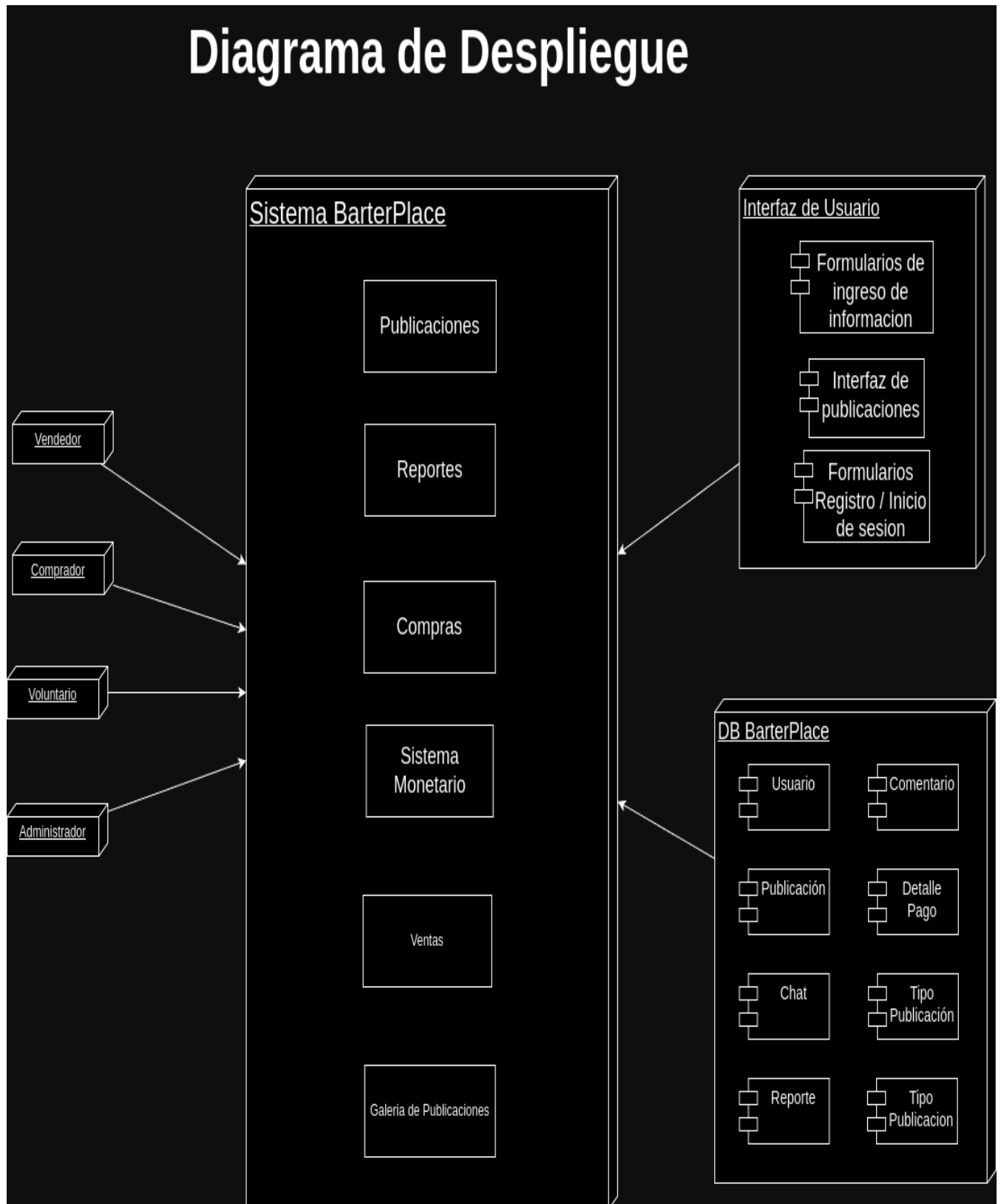


Diagrama de Paquetes

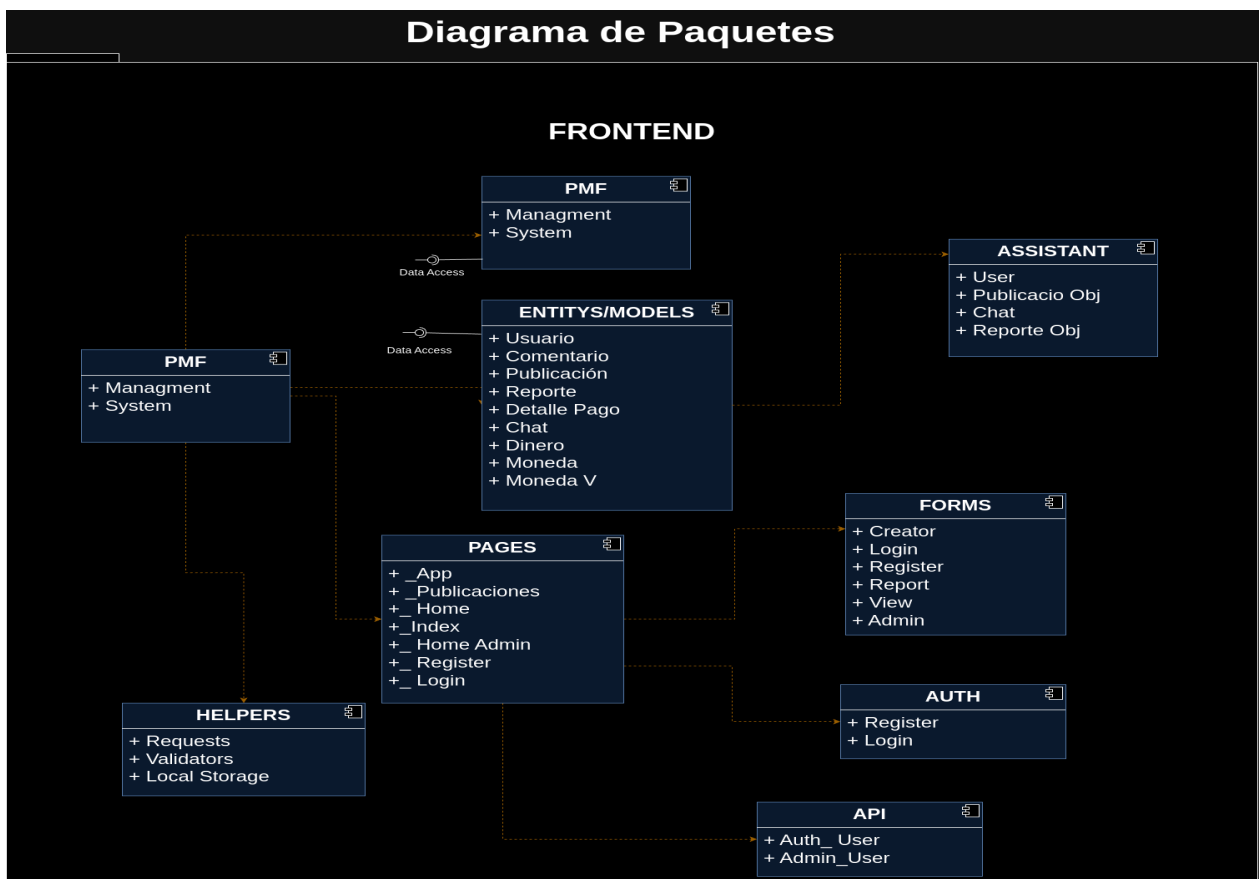
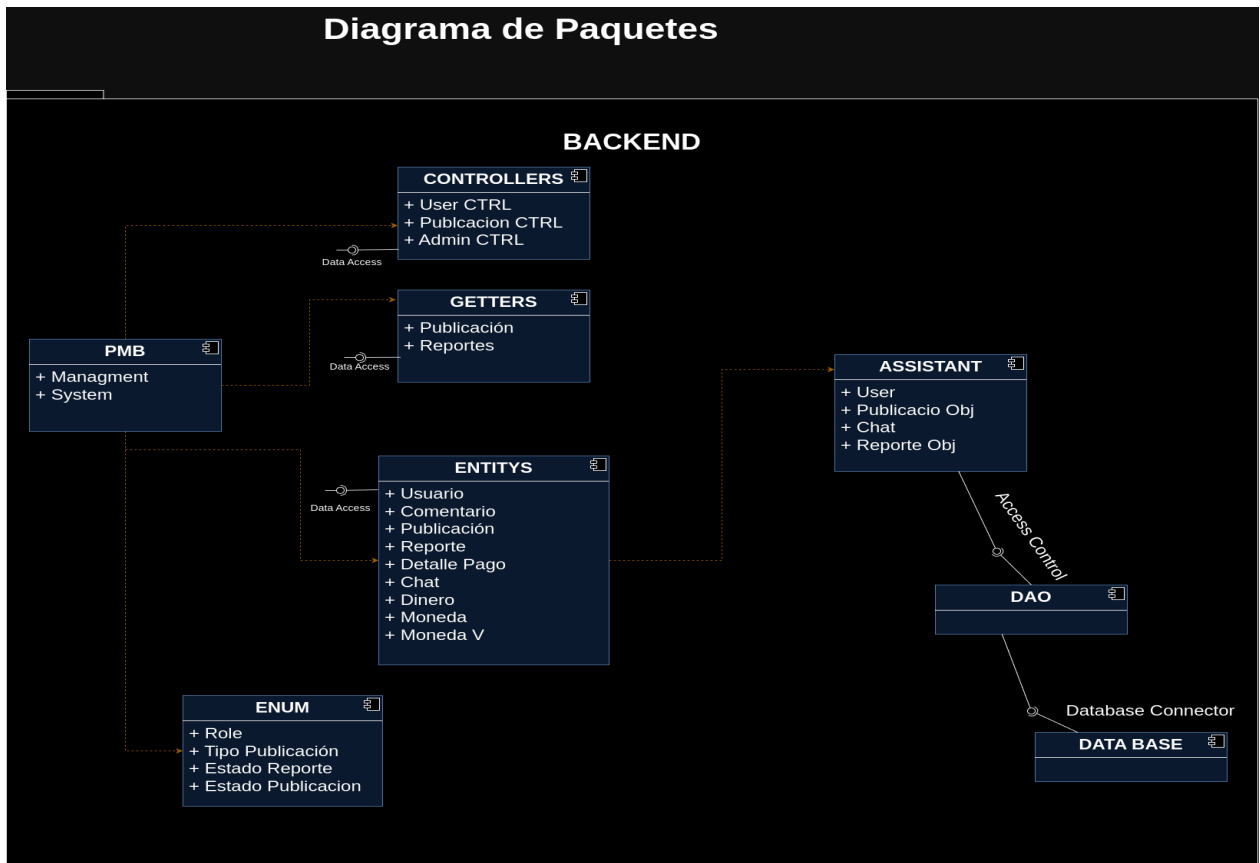
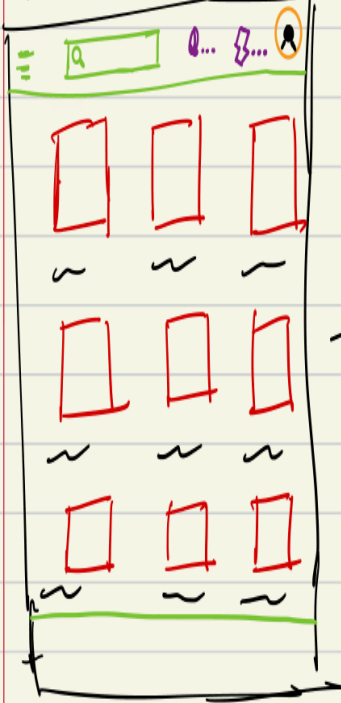


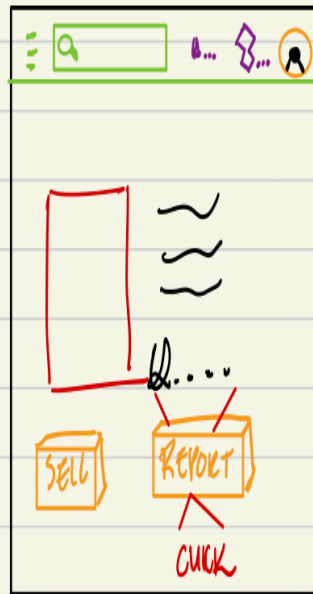
Diagrama de Estructura

Posibles Vistas

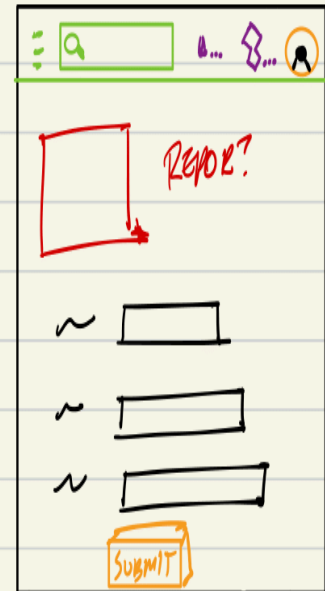
PANTALLA PRINCIPAL



REPORTE
PUBLICACION



FORMULARIO
REPORTE



FORMULARIO REGISTRO

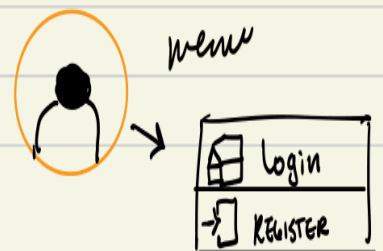
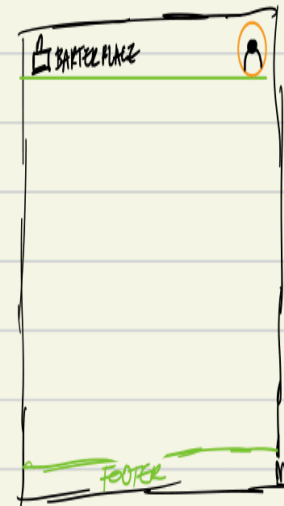
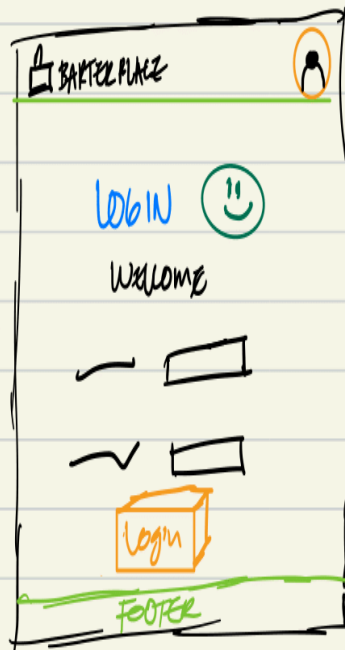
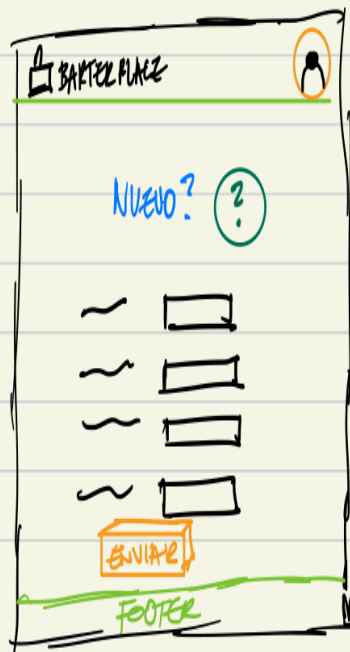


Diagrama de Comunicación

