

ARQUITECTURA

PRESENTADO POR

LUISA D. MORALES HURTADO

ANA SOFIA PALECHOR GOMEZ

ANGIE YULIANA NARVAEZ ROSERO

DOCENTE

MARIA ISABEL BASTIDAS



**INSTITUCIÓN UNIVERSITARIA
COLEGIO MAYOR DEL CAUCA**

INSTITUCIÓN UNIVERSITARIA COLEGIO MAYOR DEL CAUCA

INGENIERÍA INFORMÁTICA

POPAYÁN-CAUCA

2025

Contenido

1. INTRODUCCIÓN	6
2. OBJETIVO	6
3. ALCANCE	6
4. DEFINICIONES	7
5. INTERESADOS DEL DISEÑO DE ARQUITECTURA	8
6. MODELO DE MOTIVADORES DE NEGOCIO	9
6.1 Motivadores	9
6.2 Metas	9
6.3 Objetivos	10
7. REQUERIMIENTOS FUNCIONALES	10
8. ATRIBUTOS DE CALIDAD	12
8.1 Trazabilidad	13
8.2 Seguridad	13
9. RESTRICCIONES	14
10. DECISIONES ARQUITECTÓNICAS DE DISEÑO	14
10.1 Tácticas	14
10.2 Patrones	15
10.3 Frameworks	16
11. VISTAS DE LA ARQUITECTURA	16
11.1 Vista lógica	17
11.1.1 Diagrama de clases	17
11.1.2 Diagrama de modelo relacional de base de datos	18
11.2 Vista de desarrollo	19
11.2.1 Diagrama de componentes	20
11.3 Vista de procesos	20
11.3.1 Diagrama de secuencia	20
11.3.2 Diagrama de actividades	35
11.4 Vista física	37
11.4.1 Diagrama de despliegue	38
11.5 Vista de escenarios	38
11.5.1 Diagrama de casos de uso	39
12. AMBIENTES IMPLEMENTADOS	39
13. DIMENSIONAMIENTO – CRECIMIENTO	40
14. COMPONENTES HABILITADORES DEL DESARROLLO	41

15.	RIESGOS DE LA ARQUITECTURA	42
16.	CONCLUSIÓN	43

INDICE DE TABLAS

Tabla 1. Interesados del diseño de arquitectura.....	8
Tabla 2. Motivadores.....	9
Tabla 3. Metas	9
Tabla 4. Objetivos	10
Tabla 5. Requerimientos Funcionales	12
Tabla 6. Atributos de Calidad	13
Tabla 7. Restricciones.....	14
Tabla 8. Tácticas.....	15
Tabla 9. Patrones.....	15
Tabla 10. Frameworks	16
Tabla 11. Ambientes implementados	40
Tabla 12. Habilitadores de Desarrollo	42
Tabla 13. Riesgos de la Arquitectura	43

ÍNDICE DE ILUSTRACIONES

Ilustración 1. Diagrama de clases	18
Ilustración 2. Diagrama de modelo relacional de base de datos.....	19
Ilustración 3. Diagrama de componentes	20
Ilustración 4. Diagrama de secuencia usuario (Iniciar sesión)	21
Ilustración 5. Diagrama de secuencia usuario (Editar cuenta).....	22
Ilustración 6. Diagrama de secuencia usuario (eliminar cuenta).....	23
Ilustración 7. Diagrama de secuencia usuario (cerrar sesión)	23
Ilustración 8. Diagrama de secuencia cliente (buscar producto).....	24
Ilustración 9. Diagrama de secuencia cliente (contactar al emprendedor).....	25
Ilustración 10. Diagrama de secuencia cliente (método de pago)	26
Ilustración 11. Diagrama de secuencia cliente (ver estado del pedido)	27
Ilustración 12. Diagrama de secuencia emprendedor (publicar producto)	28
Ilustración 13. Diagrama de secuencia emprendedor (ver lista de pedidos).....	29
Ilustración 14. Diagrama de secuencia emprendedor (generar factura)	30
Ilustración 15. Diagrama de secuencia emprendedor (actualizar estado pedido)	30
Ilustración 16. Diagrama de secuencia emprendedor (editar producto).....	31
Ilustración 17. Diagrama de secuencia emprendedor (eliminar producto)	32
Ilustración 18. Diagrama de secuencia administrador (validar cuentas)	33
Ilustración 19. Diagrama de secuencia administrador (validar productos)	34
Ilustración 20. Diagrama de actividades administrador	35
Ilustración 21. Diagrama de actividades emprendedor	36
Ilustración 22. Diagrama de actividades cliente.....	37
Ilustración 23. Diagrama de despliegue	38
Ilustración 24. Diagrama de casos de uso.....	39

1. INTRODUCCIÓN

El presente documento tiene como propósito describir los elementos fundamentales de la arquitectura de la página web desarrollada para el impulso de emprendimientos locales en la ciudad de Popayán. La plataforma busca servir como un espacio digital que conecte a los emprendedores con clientes potenciales, facilitando la publicación, promoción y gestión de productos en un entorno accesible y seguro. A través de esta herramienta, se pretende fortalecer la economía local al brindar visibilidad a los negocios emergentes y ofrecer a los usuarios una experiencia intuitiva para interactuar con los diferentes servicios.

2. OBJETIVO

Definir la estructura y las decisiones de diseño de la arquitectura de la página web orientada a emprendimientos locales de Popayán, tomando como base los requerimientos funcionales y no funcionales del sistema. Este documento busca servir de guía en la implementación, garantizando que los módulos de administración, emprendedores y clientes trabajen de manera integrada, permitiendo la gestión eficiente de usuarios, productos y pedidos, en coherencia con las necesidades de la comunidad emprendedora y los lineamientos tecnológicos establecidos.

3. ALCANCE

La página web para emprendimientos locales de la ciudad de Popayán (SheMarket) constituye un sistema de información orientado a la gestión integral de usuarios, productos y pedidos. Este sistema está diseñado para cubrir tres perfiles principales: administrador, encargado de aprobar y supervisar las cuentas y productos; emprendedor, responsable de registrar y gestionar sus productos; y cliente, quien puede visualizar, adquirir productos y comunicarse con los emprendedores.

El presente documento de arquitectura establece los lineamientos que describen la estructura, componentes y relaciones del sistema, garantizando una guía y fácil entendimiento requerido para la correcta comprensión, validación y evolución del proyecto. Su alcance incluye los módulos de autenticación de usuarios, gestión de productos, gestión de pedidos y notificaciones, los cuales en conjunto permiten el adecuado funcionamiento de la plataforma y aseguran la coherencia entre los requerimientos funcionales, no funcionales y las decisiones de diseño adoptadas.

4. DEFINICIONES

En esta sección se presentan los acrónimos, términos y siglas utilizados en el documento de arquitectura para facilitar su comprensión:

- **HTML (HyperText Markup Language):** Lenguaje de marcado utilizado para estructurar las páginas web.
- **CSS (Cascading Style Sheets):** Lenguaje de diseño que permite dar estilo y presentación a las páginas web.
- **JS (JavaScript):** Lenguaje de programación del lado del cliente que permite crear interactividad en la interfaz del sistema.
- **Python:** Lenguaje de programación utilizado en el backend para implementar la lógica de negocio y gestionar la conexión con la base de datos.
- **PostgreSQL:** Sistema gestor de bases de datos relacional (RDBMS) utilizado para almacenar información de usuarios, productos, pedidos y facturas.
- **TailwindCSS:** Framework CSS que proporciona clases utilitarias para construir interfaces modernas y responsivas.
- **Framework:** Conjunto de herramientas y librerías que facilitan el desarrollo de aplicaciones bajo buenas prácticas de arquitectura y patrones de diseño.
- **Patrón de diseño:** Solución reutilizable a problemas comunes en la arquitectura de software. En el proyecto se emplean:
 - **Factory Method:** para la creación de usuarios en el inicio de sesión.
 - **Composite:** para la gestión del carrito de compras.
 - **State:** para la protección de datos personales y gestión de sesiones.
- **RNF (Requerimiento No Funcional):** Condición o restricción que define atributos de calidad como seguridad, trazabilidad, desempeño y usabilidad.
- **RF (Requerimiento Funcional):** Funcionalidades que el sistema debe cumplir, como registro de usuarios, gestión de productos, pedidos y facturación.
- **HTTPS (HyperText Transfer Protocol Secure):** Protocolo seguro de comunicación que cifra la información transmitida entre cliente y servidor.
- **Hashing:** Proceso de transformar datos sensibles (ejemplo: contraseñas) en un valor cifrado irreversible, utilizado para mejorar la seguridad del sistema.

5. INTERESADOS DEL DISEÑO DE ARQUITECTURA

ID	Rol	Nombre Completo	Cargo	Contacto	Funciones
INT-01	Director Proyecto	Maria Isabel Bastidas Mutiz	Docente	Correo electrónico: mbastidas@unimayor.edu.co	<ul style="list-style-type: none"> - Orientar el proceso de desarrollo. - Revisar y aprobar el documento de arquitectura. - Evaluar el proyecto según criterios académicos.
INT-02	Desarrolladora	Luisa Dariana Morales Hurtado	Estudiante ingeniería informática	Correo electrónico: ldmorales@unimayor.edu.co	<ul style="list-style-type: none"> - Participar en el modelado de base de datos. - Colaborar en la definición de la arquitectura. - Elaborar parte de la documentación de soporte.
INT-03	Desarrolladora	Ana Sofia Palechor Gomez	Estudiante ingeniería informática	Correo electrónico: anapalechor@unimayor.edu.co	<ul style="list-style-type: none"> - Participar en el diseño y codificación del sistema. - Documentar requerimientos funcionales y no funcionales. - Apoyar en la elaboración del documento de arquitectura.
INT-04	Desarrolladora	Angie Yuliana Narvaez Rosero	Estudiante ingeniería informática	Correo electrónico: anarvaezr@unimayor.edu.co	<ul style="list-style-type: none"> - Implementar funcionalidades del sistema. - Encargada de pruebas y validación de módulos. - Contribuir en diagramas y documentación técnica.

Tabla 1. Interesados del diseño de arquitectura

6. MODELO DE MOTIVADORES DE NEGOCIO

El modelo de motivadores de negocio aplicado al sistema web de información para emprendimientos locales de Popayán (SheMarket) permite identificar las razones estratégicas que impulsan el desarrollo de la plataforma, así como las metas y objetivos que guían su implementación.

6.1 Motivadores

ID	Descripción	Importancia
MOT-001	Baja visibilidad de los emprendimientos locales en la ciudad de Popayán, lo que limita su alcance de clientes.	Alta
MOT-002	Necesidad de un canal digital que facilite la interacción entre emprendedores y clientes, dado el crecimiento del comercio electrónico.	Alta
MOT-003	Falta de herramientas centralizadas para la gestión de productos y pedidos de los emprendedores locales.	Media

Tabla 2. Motivadores

6.2 Metas

ID	Motivadores Asociados	Descripción
MET-001	MOT-001	Incrementar la visibilidad digital de los emprendimientos locales mediante una plataforma accesible.
MET-002	MOT-002	Establecer un canal confiable de comunicación y compra entre clientes y emprendedores.
MET-003	MOT-003	Ofrecer un sistema centralizado para registrar, administrar y dar seguimiento a productos y pedidos

Tabla 3. Metas

6.3 Objetivos

ID	Metas Asociadas	Descripción
OBJ-001	MET-001	<ul style="list-style-type: none"> - Lograr que al menos un 70% de los emprendedores registrados aumenten su visibilidad digital en el primer año. - Permitir que los productos publicados estén disponibles en un 95% del tiempo para los clientes.
OBJ-002	MET-002	<ul style="list-style-type: none"> - Garantizar que el 90% de los pedidos realizados sean gestionados correctamente en el sistema durante los seis primeros meses. - Asegurar que la plataforma brinde un canal de comunicación cliente-emprendedor con una disponibilidad alta.
OBJ-003	MET-003	<ul style="list-style-type: none"> - Reducir el tiempo de gestión de productos y pedidos para los emprendedores. - Permitir que los emprendedores realicen seguimiento de sus pedidos en tiempo real.

Tabla 4. Objetivos

7. REQUERIMIENTOS FUNCIONALES

ID	Nombre	Descripción	Entrada	Proceso	Salida
RF-01	Registro de usuarios	Permitir el registro de usuarios en la plataforma bajo los roles de administrador, emprendedor o cliente.	Datos del usuario (nombre, correo, contraseña, rol).	Validación de datos y almacenamiento en la base de datos.	¡Registro exitoso como (tipo de usuario)! Serás redirigido...
RF-02	Inicio de sesión	Permitir que los usuarios ingresen al sistema con credenciales válidas.	Correo, contraseña y elección de rol.	Verificación de credenciales en la base de datos.	Acceso al sistema según rol.

RF-03	Gestión de perfil de usuario	Los usuarios podrán actualizar información personal y credenciales.	Nuevos datos del perfil.	Validación y actualización en la base de datos.	Perfil actualizado.
RF-04	Gestión de productos	El emprendedor podrá registrar, modificar y eliminar productos.	Datos del producto (nombre, precio, descripción, imagen).	Almacenamiento, actualización o eliminación en la base de datos.	Producto creado, actualizado o eliminado.
RF-05	Visualización de productos	Los clientes podrán consultar el catálogo de productos publicados por los emprendedores.	Solicitud de consulta.	Recuperación de productos registrados en la base de datos.	Lista de productos disponibles.
RF-06	Búsqueda y filtrado de productos	Permitir a los clientes buscar productos por nombre.	Palabra clave o filtros.	Búsqueda en la base de datos con criterios de filtrado.	Resultados de búsqueda.
RF-07	Realización de pedidos	Los clientes podrán generar pedidos sobre los productos seleccionados.	Datos del pedido (productos, cantidad, cliente, método de pago y comprobante de pago).	Validación de stock y creación del pedido en la base de datos.	Pedido registrado.
RF-08	Gestión de pedidos	Los emprendedores podrán visualizar y actualizar el estado de los pedidos recibidos.	Consulta de pedidos.	Recuperación de pedidos del sistema y actualización de estado (pendiente, enviado, entregado).	Estado actualizado del pedido.
RF-09	Gestión de usuarios (administrador)	El administrador podrá aprobar o rechazar cuentas de emprendedores y clientes.	Solicitud de gestión de usuarios.	Validación de datos y actualización de estado en la base de datos.	Usuario aprobado/rechazado.
RF-10	Notificaciones	El sistema generará	Evento generado	Envío de notificación	Notificación entregada.

		notificaciones automáticas a usuarios sobre cambios en pedidos, cuentas y productos.	(nuevo pedido, cambio de estado, aprobación).	por interfaz de usuario.	
RF-11	Contacto cliente-emprendedor	Los clientes podrán comunicarse con los emprendedores a través de la plataforma.	Mensaje del cliente.	Envío del mensaje al emprendedor asociado.	Mensaje entregado al emprendedor.
RF-12	Cierre de sesión	Permitir a los usuarios cerrar su sesión de forma segura.	Solicitud de cierre de sesión.	Eliminación de la sesión activa en el sistema.	Usuario desconectado.

Tabla 5. Requerimientos Funcionales

8. ATRIBUTOS DE CALIDAD

ID	Atributo de Calidad	ID-RNF	Descripción	Importancia
ATC-001	Usabilidad	RNF-001	La plataforma debe ser intuitiva y fácil de usar, permitiendo que emprendedores y clientes sin experiencia técnica puedan registrarse y navegar sin dificultad.	Alto
ATC-002	Seguridad	RNF-002, RNF-003	El sistema garantizará la protección de datos personales y credenciales de acceso, aplicando protocolos seguros de autenticación y encriptación.	Alto
ATC-003	Disponibilidad	RNF-004	La disponibilidad del sistema será del 95% mensual, asegurando que emprendedores y clientes puedan acceder a la plataforma en la mayoría del tiempo.	Alto
ATC-004	Desempeño	RNF-005	El sistema debe soportar un promedio de 20 solicitudes simultáneas sin pérdida significativa de rendimiento.	Medio
ATC-005	Mantenibilidad	RNF-006	El sistema deberá permitir la actualización de funcionalidades y	Medio

			corrección de errores sin afectar la operatividad de los usuarios finales.	
ATC-006	Accesibilidad	RNF-007	La plataforma debe estar disponible desde distintos dispositivos (computadores y móviles), adaptándose a diferentes tamaños de pantalla.	Alto

Tabla 6. Atributos de Calidad

8.1 Trazabilidad

El sistema implementará mecanismos de trazabilidad para garantizar que todas las acciones realizadas por los usuarios (clientes, emprendedores, administradores) queden registradas de forma segura.

Esto permitirá identificar qué usuario ejecutó cada operación, en qué momento y desde qué entorno, facilitando la auditoría del sistema, la gestión de incidentes y la mejora continua.

- Se registrarán eventos como: inicio y cierre de sesión, creación de cuentas, publicación de productos, generación de pedidos, emisión de facturas y actualizaciones en el estado de un pedido.
- Los registros estarán almacenados en una tabla de auditoría en la base de datos y podrán ser consultados únicamente por perfiles con rol de administrador.
- Los logs de errores del sistema serán capturados para diagnóstico técnico, con el fin de corregir fallas oportunamente.

8.2 Seguridad

El sistema contempla medidas de seguridad para la protección de la información de los usuarios (clientes, emprendedores y administradores), en concordancia con la Ley 1581 de 2012 y Decreto 1377 de 2013 sobre protección de datos personales.

Comunicación segura:

- El sistema usará protocolos HTTPS para proteger la transmisión de datos entre cliente y servidor.

Autenticación y autorización:

- Solo los usuarios registrados podrán acceder a sus funcionalidades según su rol (cliente, emprendedor, administrador).
- El sistema contará con control de sesiones, de manera que al cerrar sesión se invalide el acceso.

9. RESTRICCIONES

ID	Tipo	Descripción
RST-01	Legal	La plataforma deberá cumplir con la Ley 1581 de 2012 de Protección de Datos Personales, garantizando la seguridad en el almacenamiento y tratamiento de la información de usuarios.
RST-02	Técnica	La aplicación debe estar desarrollada utilizando el framework Flask en Python y bases de datos Posgres, de acuerdo con los lineamientos definidos para el proyecto.
RST-03	Infraestructura	El sistema dependerá de un servidor web con conexión estable a internet, lo cual puede limitar la disponibilidad en caso de fallos de red.
RST-04	Software	El sistema será accesible desde navegadores web modernos (Google Chrome, Mozilla Firefox, Edge), pero no contempla compatibilidad con versiones obsoletas.
RST-05	Datos	Las imágenes de productos cargadas por los emprendedores tendrán un tamaño máximo de 5 MB para optimizar el almacenamiento y la velocidad de carga.

Tabla 7. Restricciones

10. DECISIONES ARQUITECTÓNICAS DE DISEÑO

10.1 Tácticas

ID	ID Relacionado	Táctica Asociada
TAC-001	ATC-001	Usabilidad. La táctica empleada es el diseño de interfaces intuitivas y pruebas de usuario para validar la facilidad de navegación.
TAC-002	ATC-002	Seguridad. La táctica empleada es la implementación de contraseñas encriptadas, HTTPS y control de acceso por roles.
TAC-003	ATC-003	Disponibilidad. La táctica empleada incluye la realización de copias de seguridad semanales y monitoreo del servidor.

TAC-004	ATC-004	Desempeño. La táctica empleada es la optimización de consultas a la base de datos y cacheo de información para mejorar tiempos de respuesta.
TAC-005	RF-08	Pedidos. La táctica empleada es la validación del stock de productos antes de confirmar un pedido para garantizar la consistencia de datos.
TAC-006	RF-07	Búsqueda de productos. La táctica empleada es la indexación de datos para acelerar los procesos de filtrado y búsqueda.

Tabla 8. Tácticas

10.2 Patrones

Los patrones de diseño aplicados en el sistema permiten dar solución a problemas recurrentes de la arquitectura, garantizando escalabilidad, reutilización de código y cumplimiento de los atributos de calidad definidos (seguridad, trazabilidad, usabilidad).

ID	Tipo	Descripción
PAT-01	Creación (Factory Method)	Se emplea el patrón Factory en el módulo de inicio de sesión. Esto permite centralizar la creación de objetos de usuario (cliente, emprendedor, administrador) según las credenciales proporcionadas. Con esto se asegura flexibilidad en la autenticación y validación de roles.
PAT-02	Estructural (Composite)	Se utiliza en la gestión del carrito de compras. Gracias a este patrón, tanto productos individuales como grupos de productos pueden manejarse de forma uniforme, y el sistema calcula automáticamente el total de la compra en función de los elementos agregados.
PAT-03	Comportamiento (State)	Se implementa en el módulo de protección de datos y gestión de sesiones. El sistema cambia de estado según la interacción del usuario: En cuenta, producto, pedido, estado de aceptación de políticas de datos y estado de cierre de sesión.

Tabla 9. Patrones

10.3 Frameworks

Los frameworks y tecnologías seleccionadas para el desarrollo del sistema se definieron en función de la facilidad de integración, la curva de aprendizaje y el soporte comunitario, garantizando así la mantenibilidad y escalabilidad del proyecto.

	Nombre	Tipo	Respaldo
FRM-01	Python	Lenguaje Backend	Permite implementar la lógica de negocio y la conexión con la base de datos. Lenguaje robusto y multiplataforma.
FRM-02	HTML5 / CSS3	Lenguajes de maquetado y estilo	Utilizados para la estructura y diseño visual de la aplicación web. CSS3 permite personalización y adaptación responsiva.
FRM-03	TailwindCSS	Framework CSS (UI)	Permite un diseño responsivo y moderno con clases utilitarias que aceleran la maquetación.
FRM-04	JavaScript	Lenguaje Frontend	Gestiona la interacción dinámica en la página (validaciones, carrito de compras, manejo de vistas y eventos).
FRM-05	PostgreSQL	DBMS Relacional	Base de datos robusta, segura y escalable. Permite manejar usuarios, productos y pedidos con integridad referencial.
FRM-06	bcrypt / hashlib	Librerías de seguridad	Implementadas para encriptación de contraseñas y protección de datos sensibles.

Tabla 10. Frameworks

11. VISTAS DE LA ARQUITECTURA

Las vistas de la arquitectura reflejan la forma en que se estructuró la solución tanto en lo funcional como en lo técnico. Permiten evidenciar cómo se resolvieron los requerimientos funcionales y no funcionales a través de la organización de clases, componentes y procesos.

11.1 Vista lógica

Está orientada a describir la estructura funcional del sistema y cómo los usuarios interactúan con los distintos módulos. Se centra en mostrar los elementos que brindan funcionalidad al sistema y la forma en que estos se relacionan entre sí.

Esta vista se centra en la representación de las entidades principales: Usuarios, Productos, Pedidos, Carrito y Factura, así como sus relaciones. Esta organización permite visualizar cómo se almacenan y procesan los datos que soportan la interacción de clientes, emprendedores y administradores dentro de la plataforma.

11.1.1 Diagrama de clases

Refleja la estructura del sistema en términos de objetos, atributos y métodos. Aquí se observa, por ejemplo, cómo un Usuario puede tener diferentes roles (cliente, emprendedor, administrador), cómo un Cliente gestiona un Carrito, y cómo este carrito genera un Pedido que puede terminar en una Factura. Este modelo asegura coherencia entre los módulos de autenticación, compra y administración.

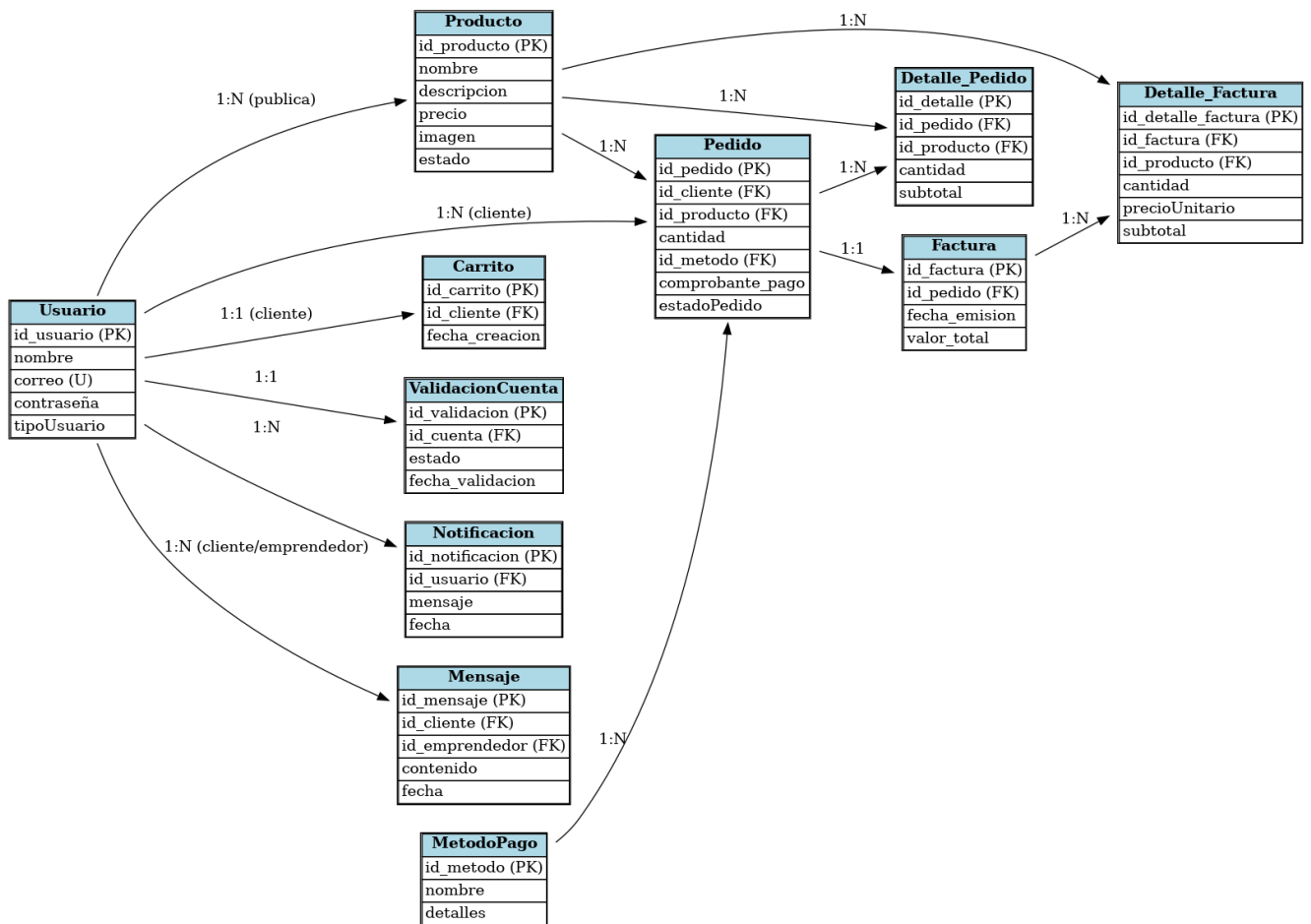


Ilustración 2. Diagrama de modelo relacional de base de datos

11.2 Vista de desarrollo

La vista de desarrollo muestra los componentes tecnológicos utilizados en el sistema. En nuestro caso, se incluyen:

- Frontend: HTML5, CSS3, TailwindCSS y JavaScript para la interfaz y validaciones.
- Backend: Python para la lógica de negocio y manejo de sesiones.
- Base de datos: PostgreSQL como sistema gestor de datos.
- Integración: Conexión entre la interfaz web y la base de datos mediante consultas SQL, que permiten registrar y consultar información de manera segura.

11.2.1 Diagrama de componentes

El diagrama de componentes evidencia cómo estos elementos se comunican entre sí para dar soporte a los casos de uso del sistema.

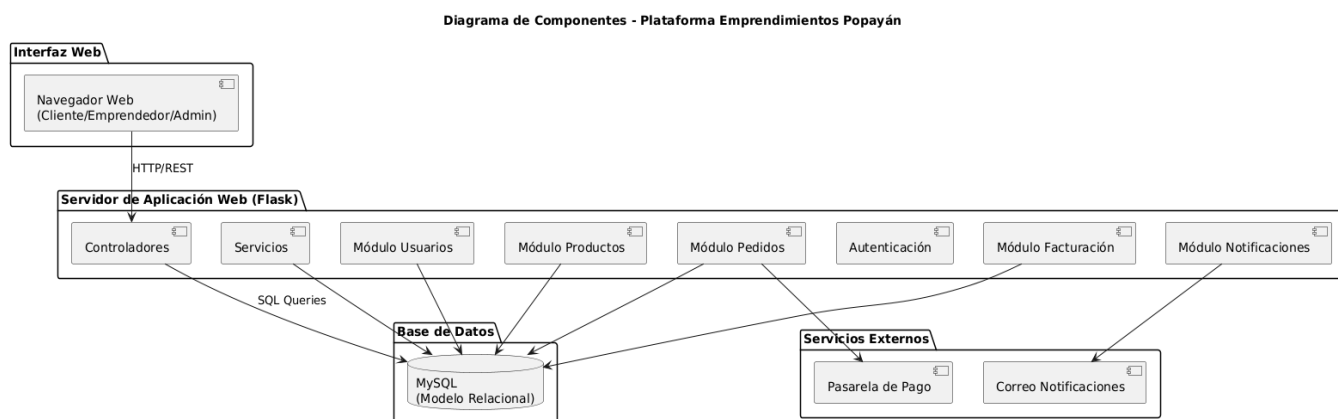


Ilustración 3. Diagrama de componentes

11.3 Vista de procesos

La vista de procesos ilustra el comportamiento dinámico del sistema. En el proyecto, se representaron procesos clave como:

- Inicio de sesión: autenticación y validación de roles.
- Gestión del carrito: agregar productos, calcular el total y generar pedidos.
- Facturación: creación de facturas asociadas a los pedidos realizados.

11.3.1 Diagrama de secuencia

- Usuario

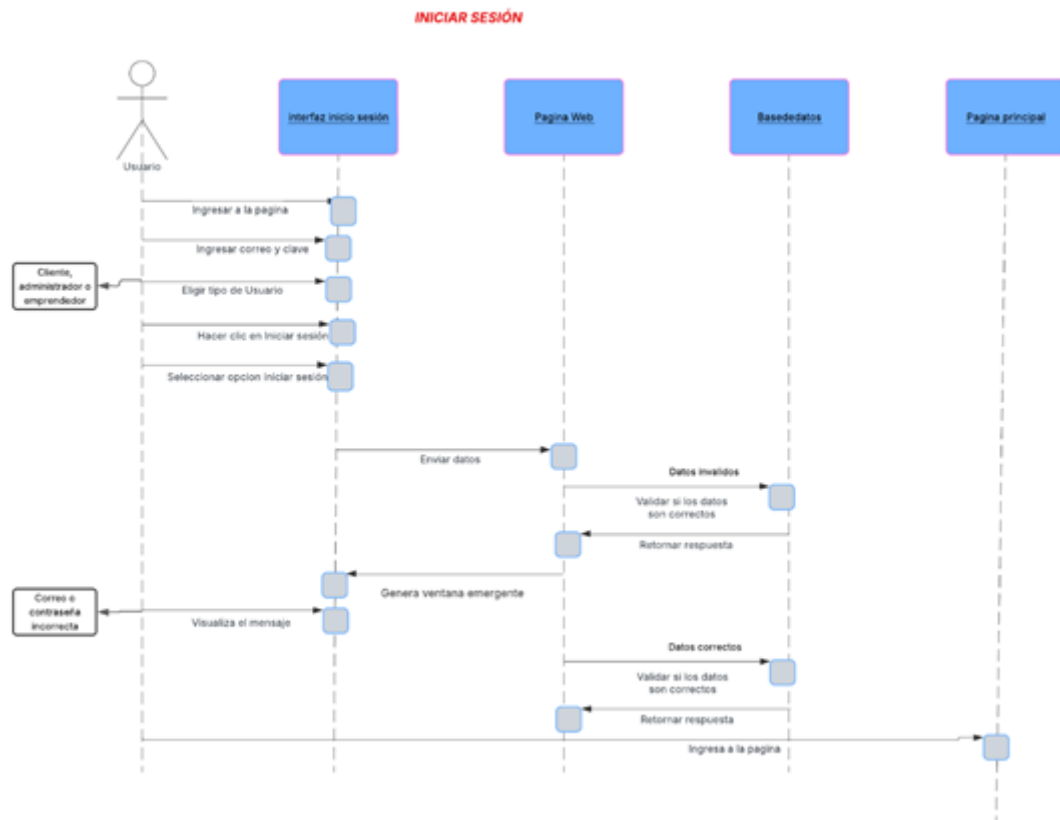


Ilustración 4. Diagrama de secuencia usuario (Iniciar sesión)

Editar cuenta

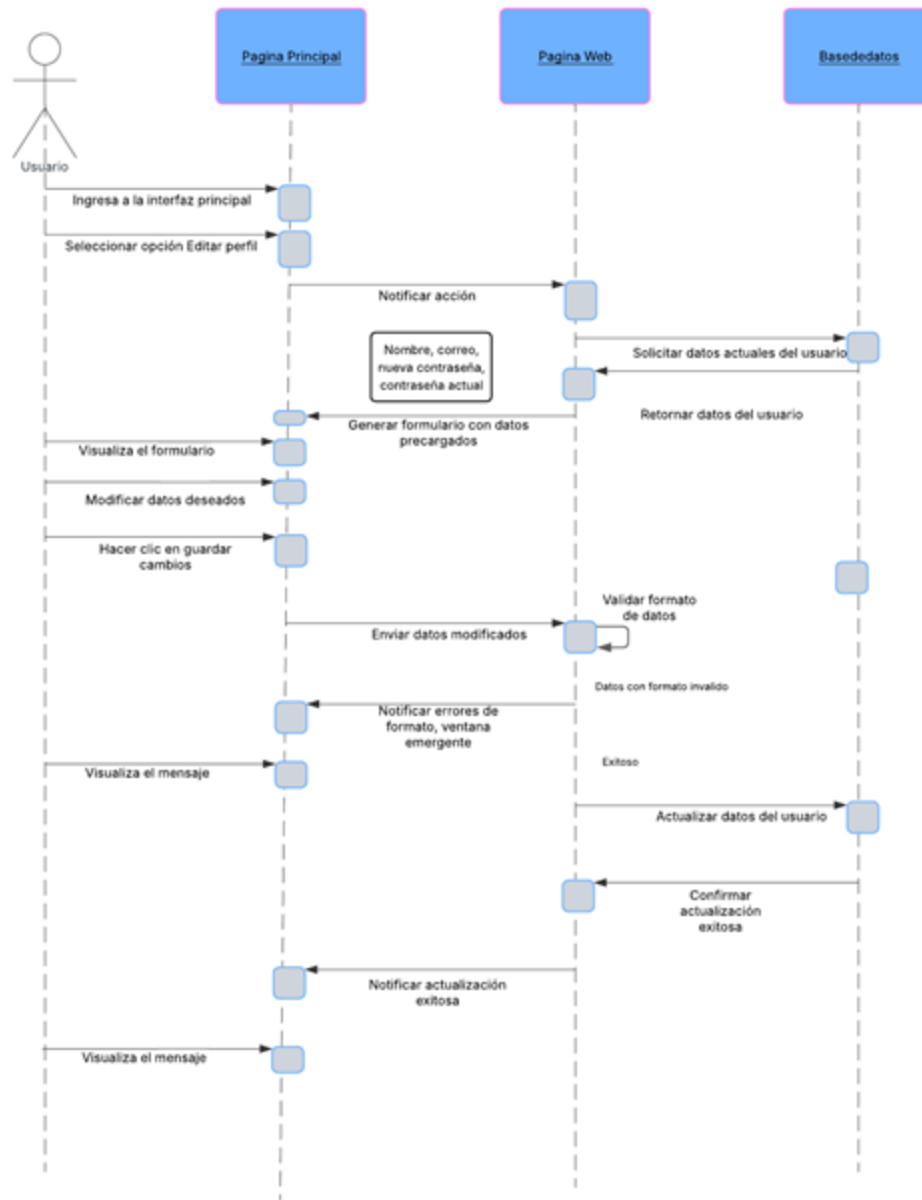


Ilustración 5. Diagrama de secuencia usuario (Editar cuenta)

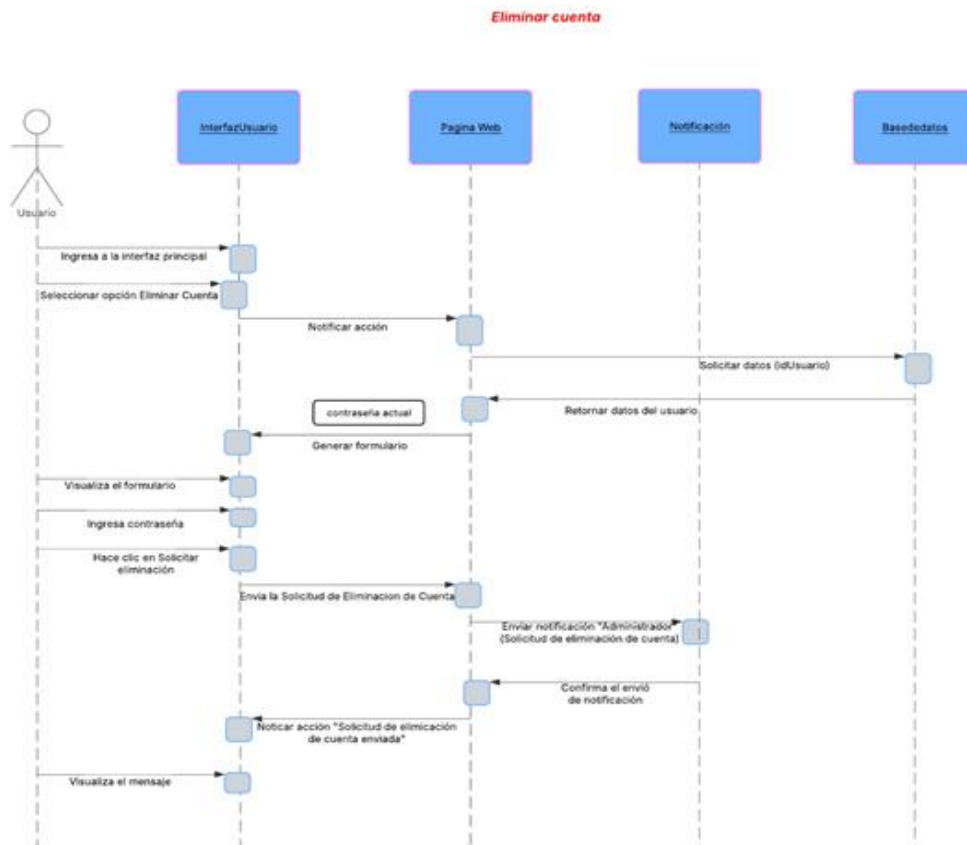


Ilustración 6. Diagrama de secuencia usuario (eliminar cuenta)

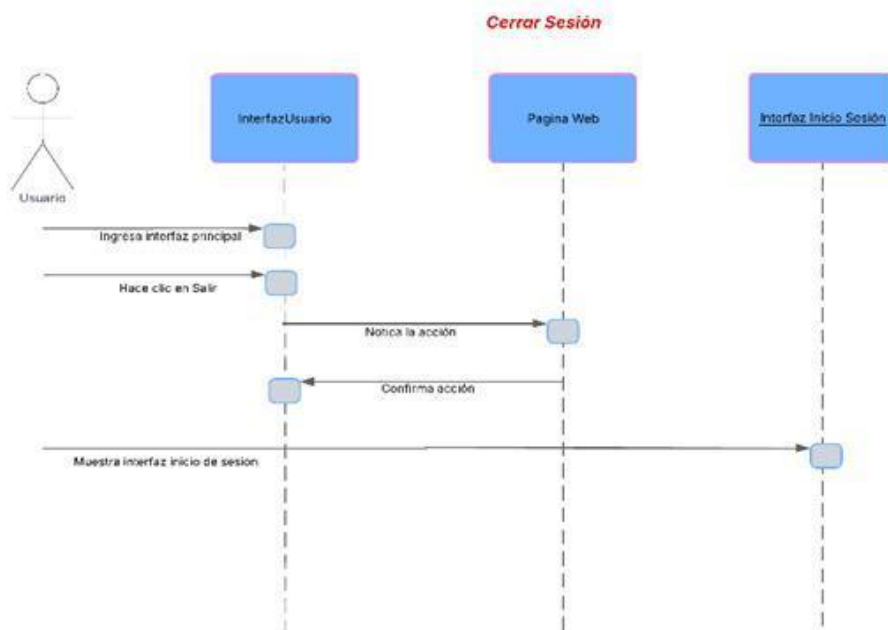


Ilustración 7. Diagrama de secuencia usuario (cerrar sesión)

- Cliente

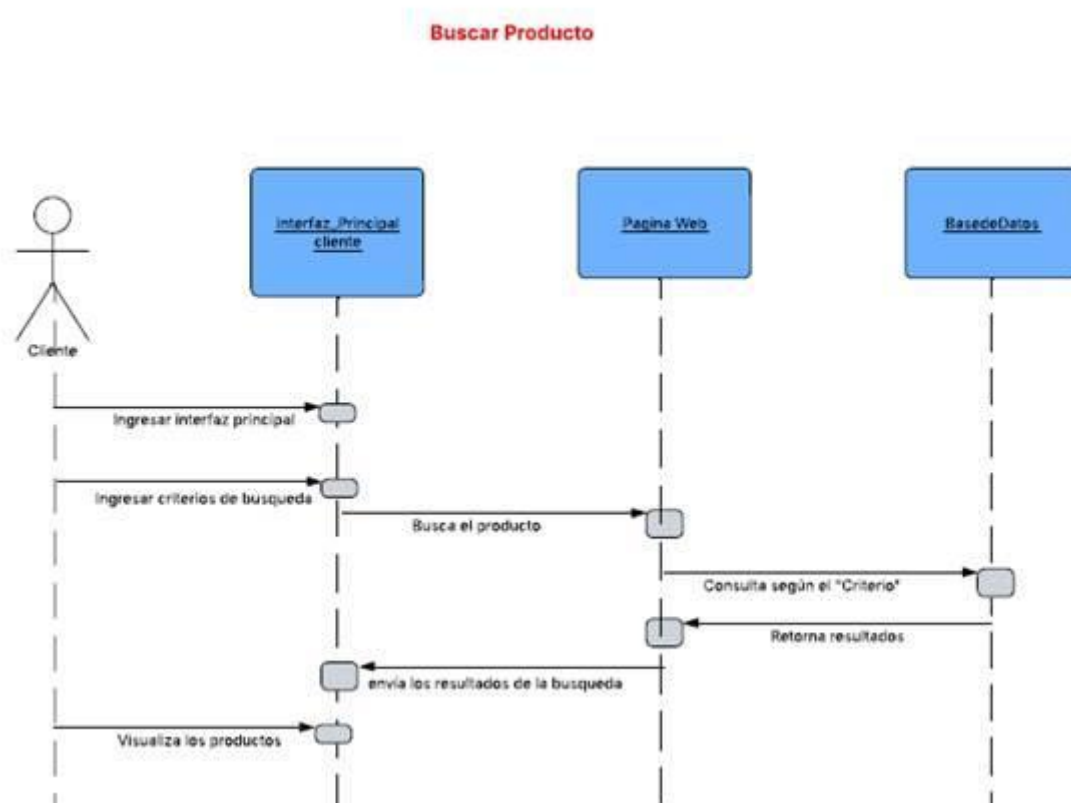


Ilustración 8. Diagrama de secuencia cliente (buscar producto)

CONTACTAR AL EMPRENDEDOR

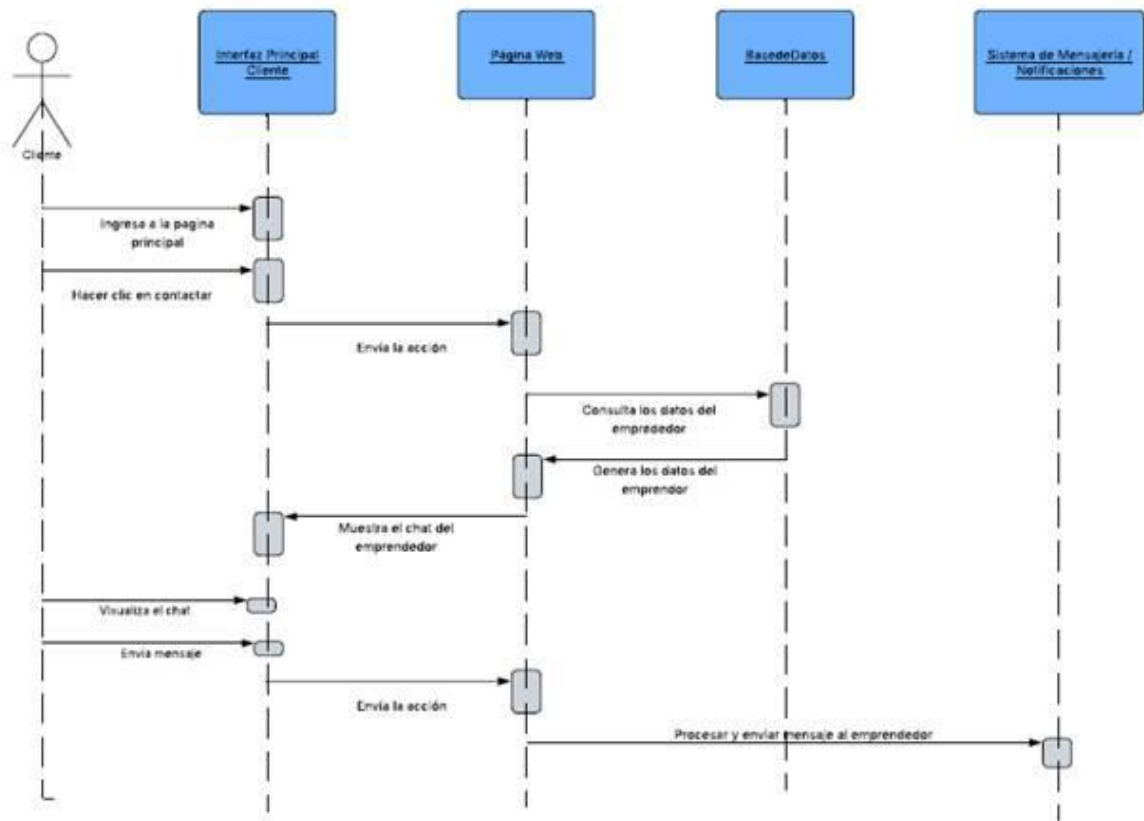


Ilustración 9. Diagrama de secuencia cliente (contactar al emprendedor)

Metodo de pago

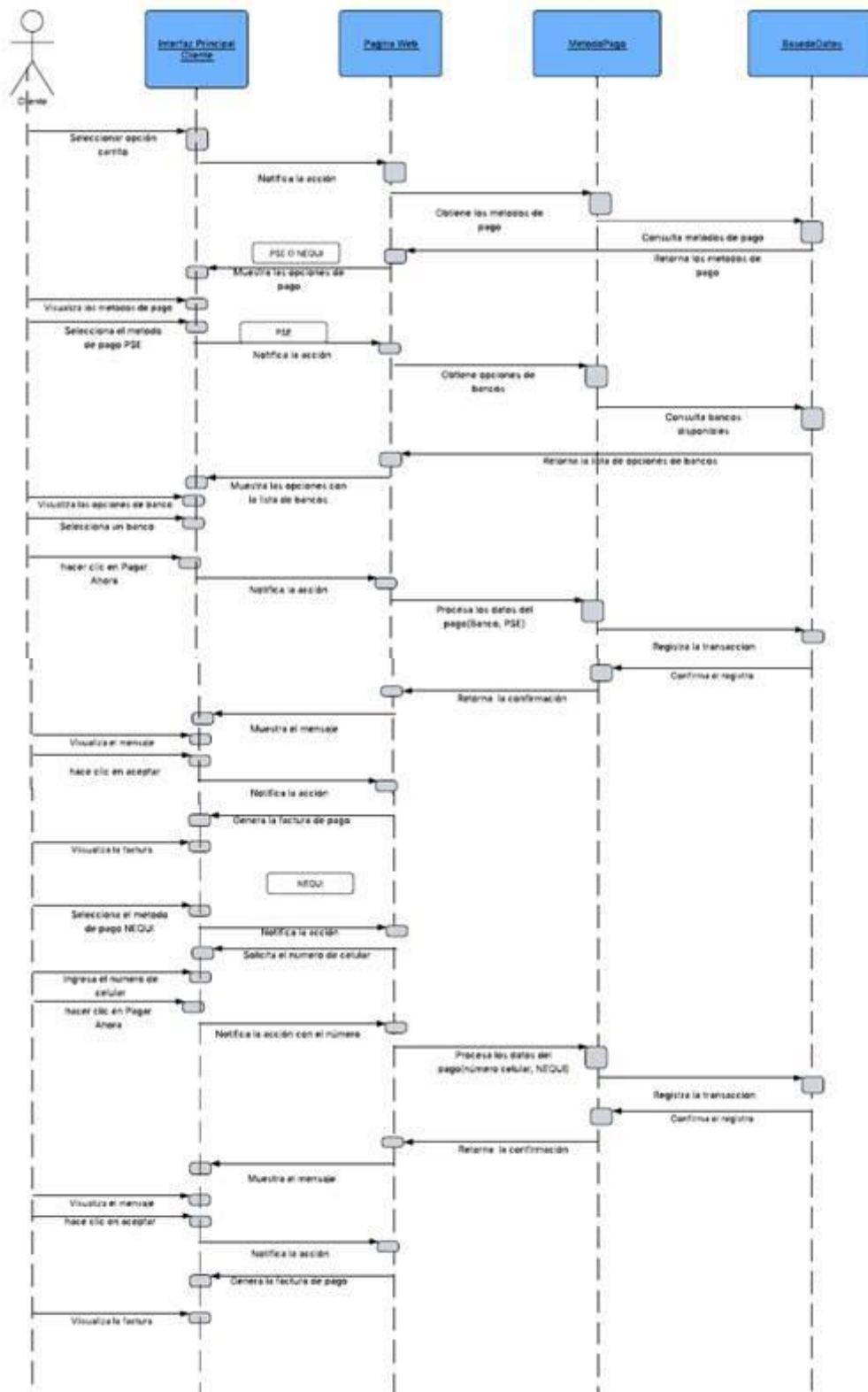


Ilustración 10. Diagrama de secuencia cliente (método de pago)

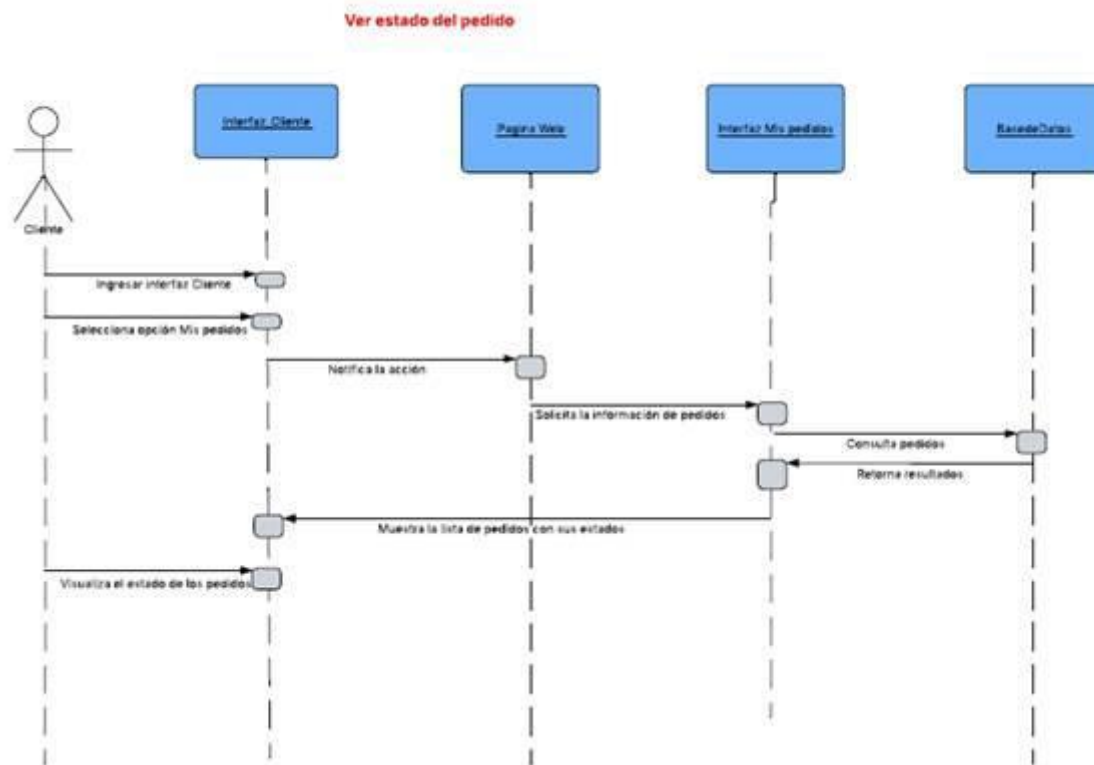


Ilustración 11. Diagrama de secuencia cliente (ver estado del pedido)

- **Emprendedor**

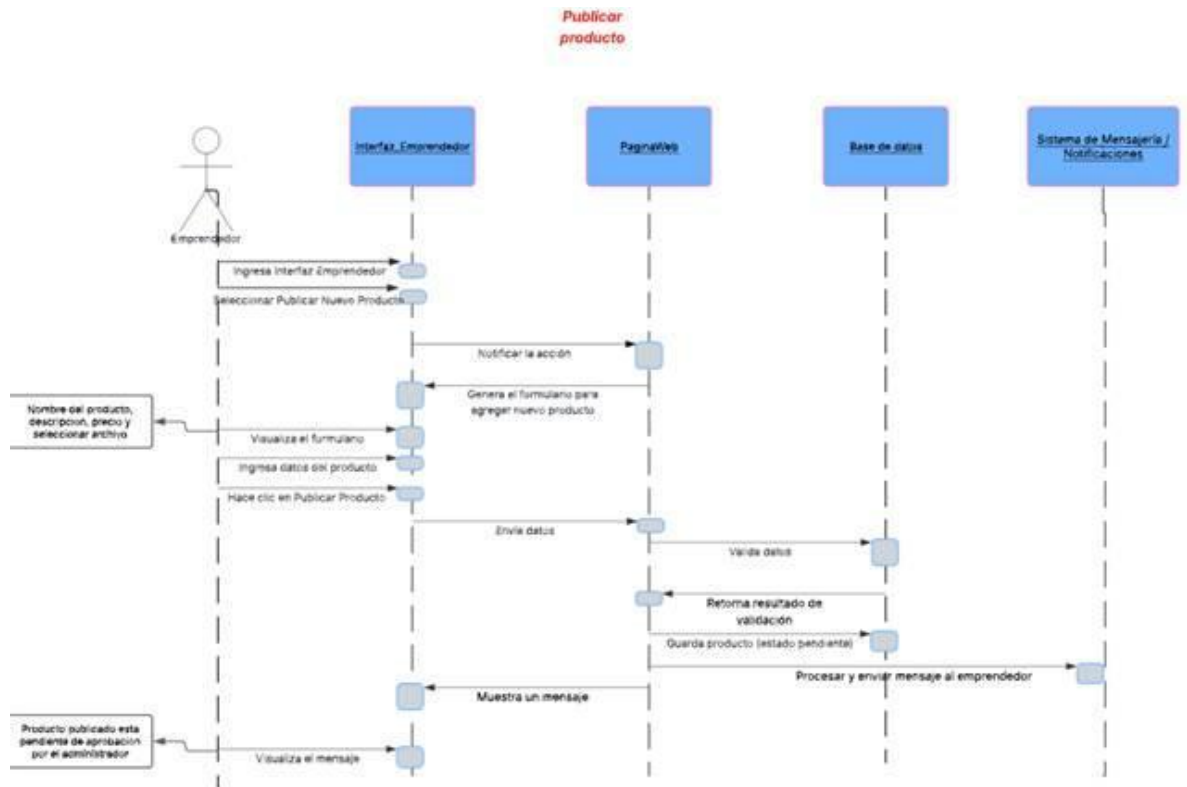


Ilustración 12. Diagrama de secuencia emprendedor (publicar producto)

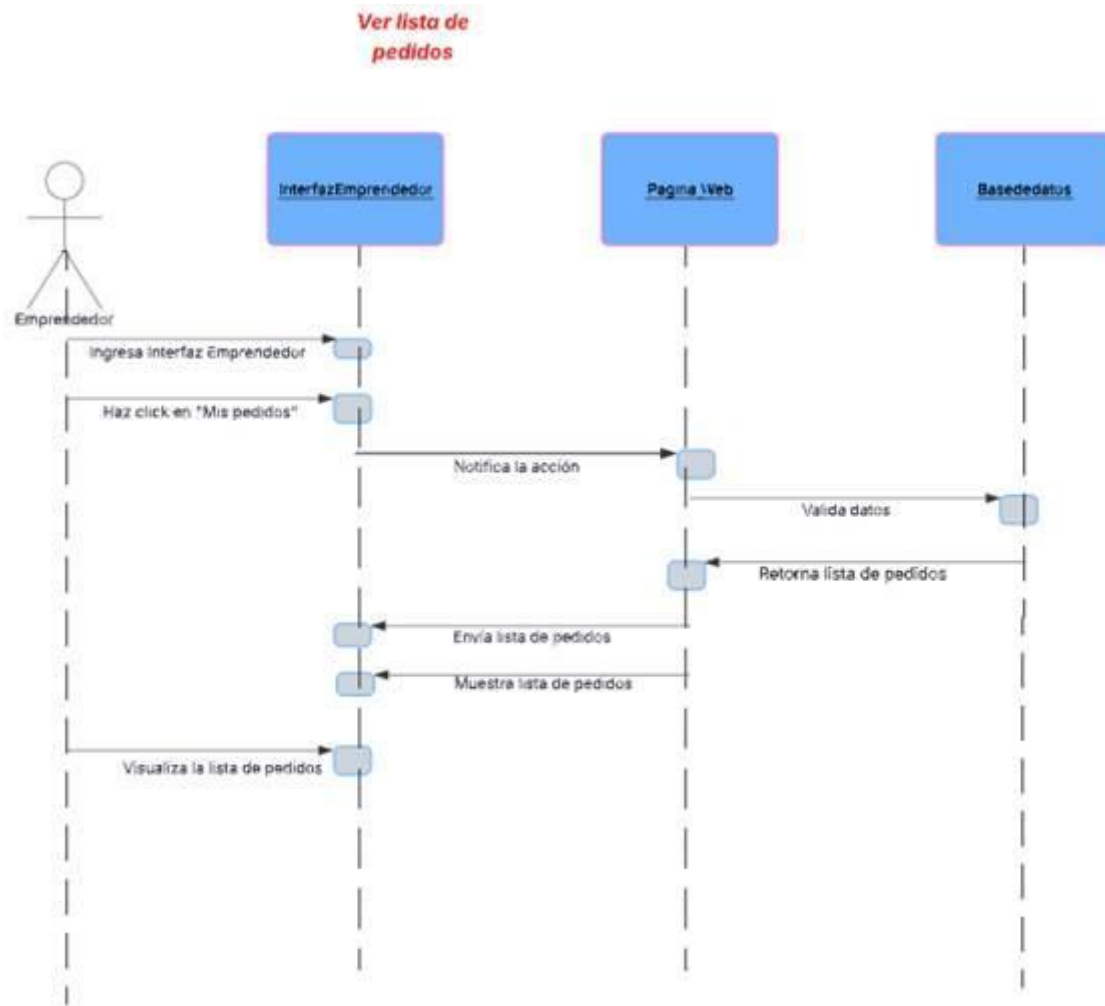


Ilustración 13. Diagrama de secuencia emprendedor (ver lista de pedidos)

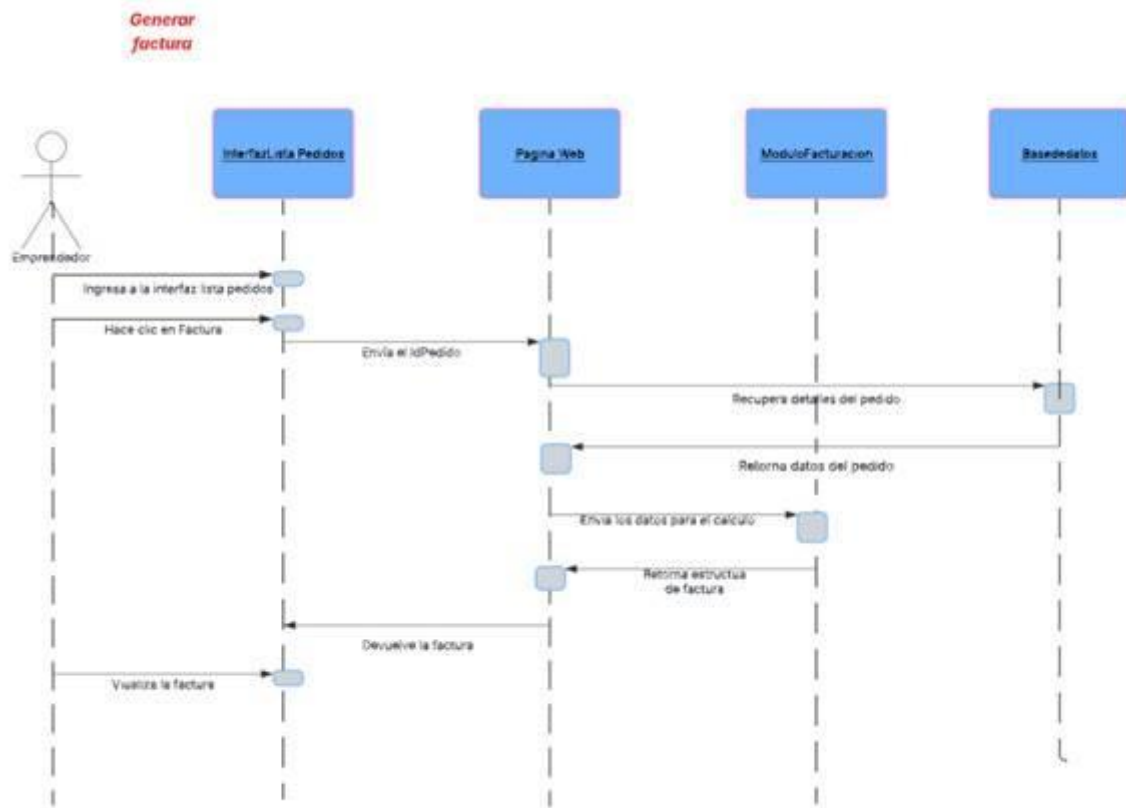


Ilustración 14. Diagrama de secuencia emprendedor (generar factura)

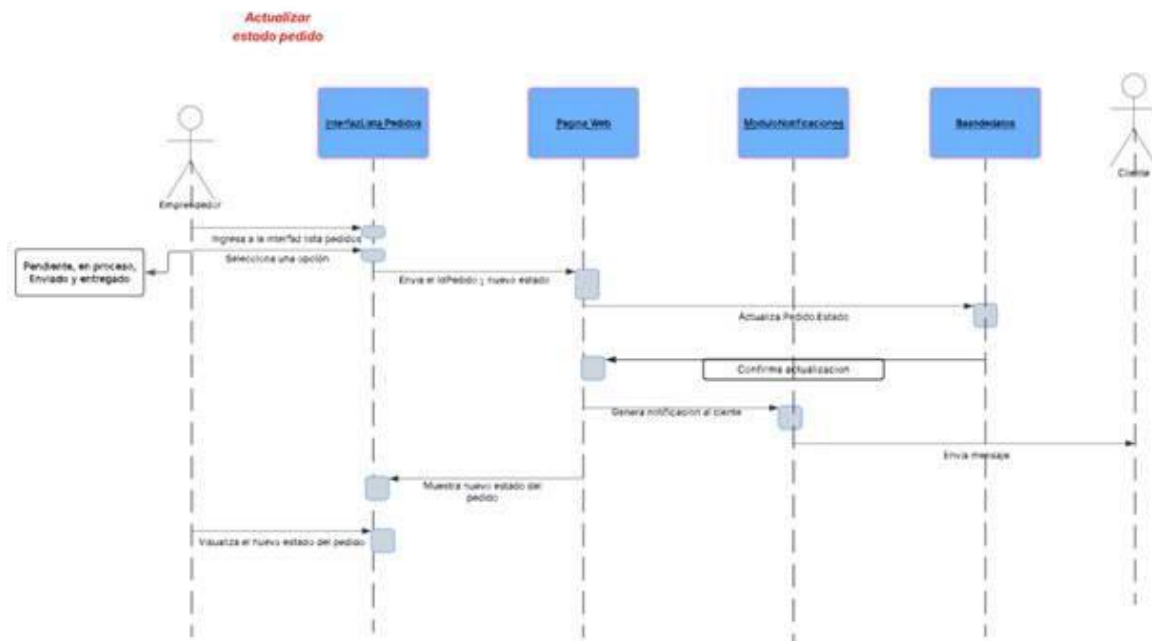


Ilustración 15. Diagrama de secuencia emprendedor (actualizar estado pedido)

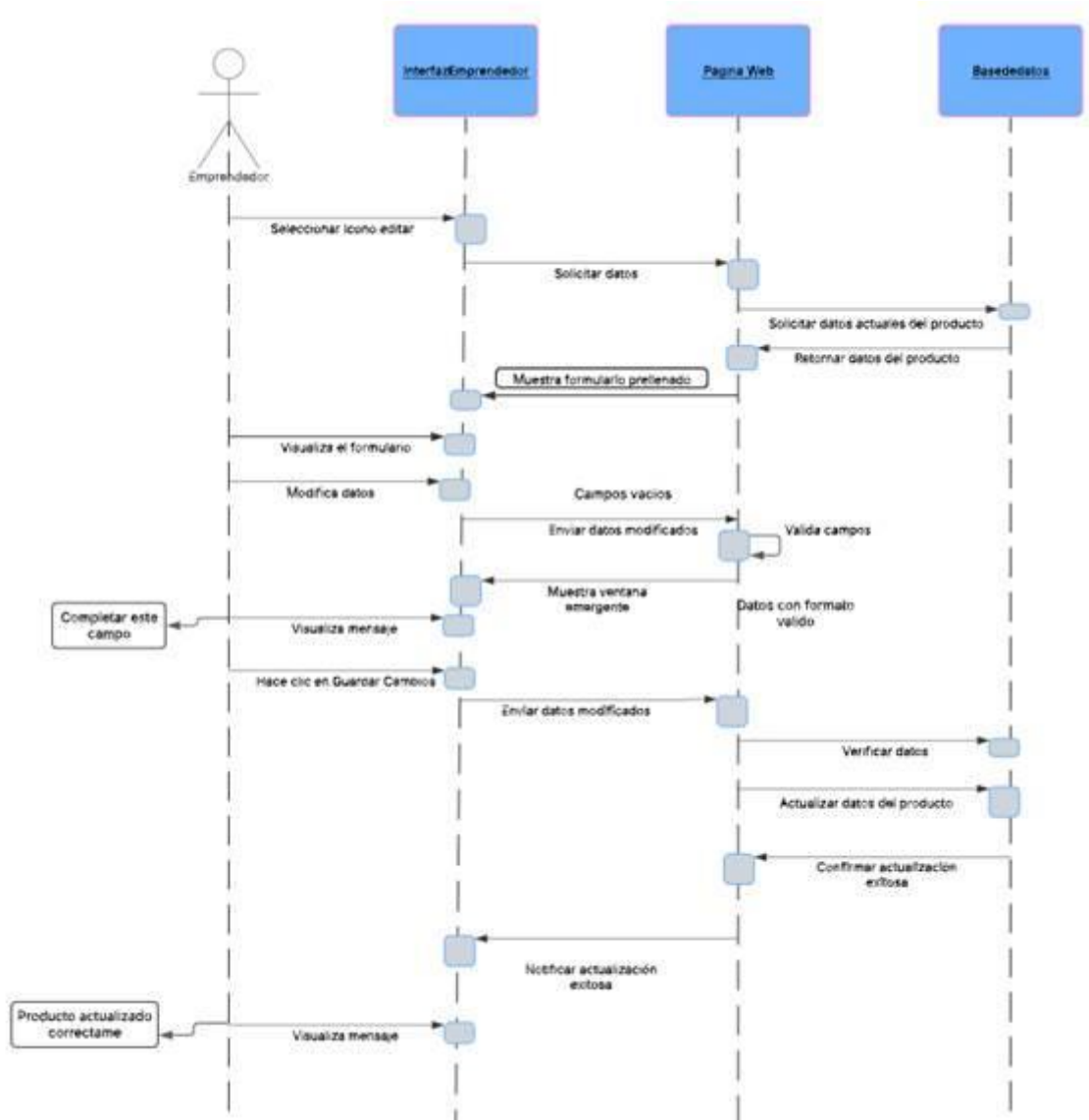


Ilustración 16. Diagrama de secuencia emprendedor (editar producto)

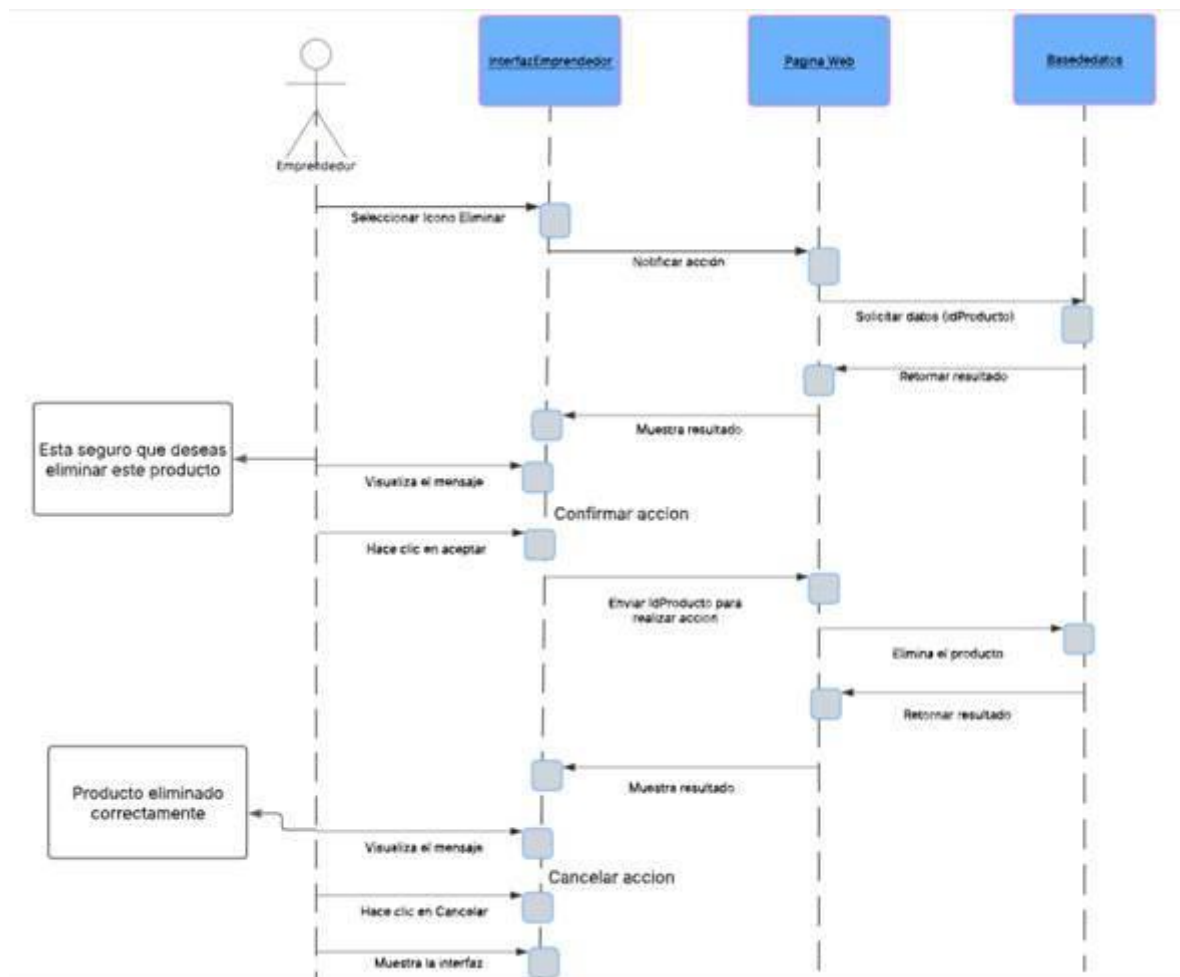


Ilustración 17. Diagrama de secuencia emprendedor (eliminar producto)

- Administrador

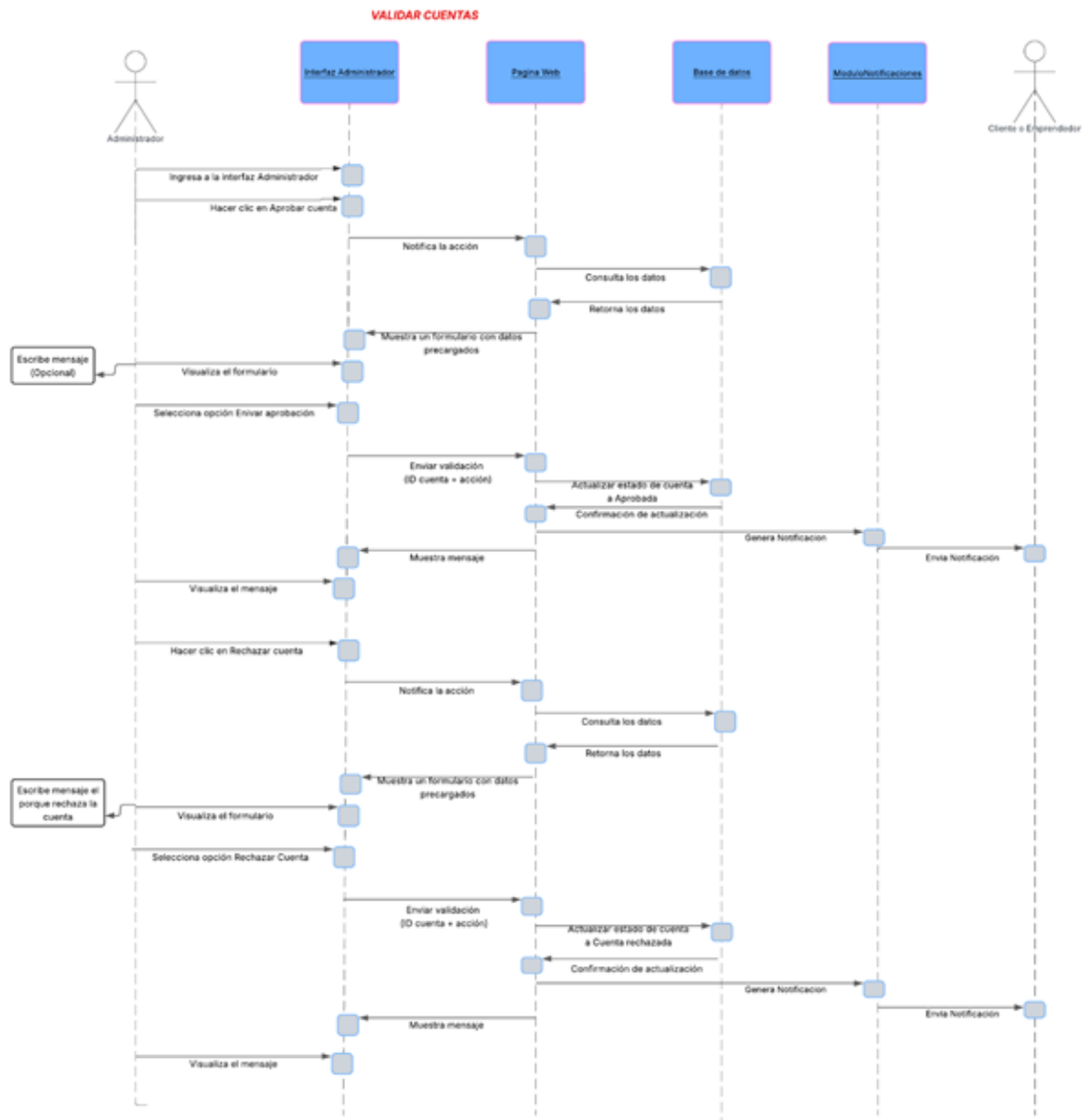


Ilustración 18. Diagrama de secuencia administrador (validar cuentas)

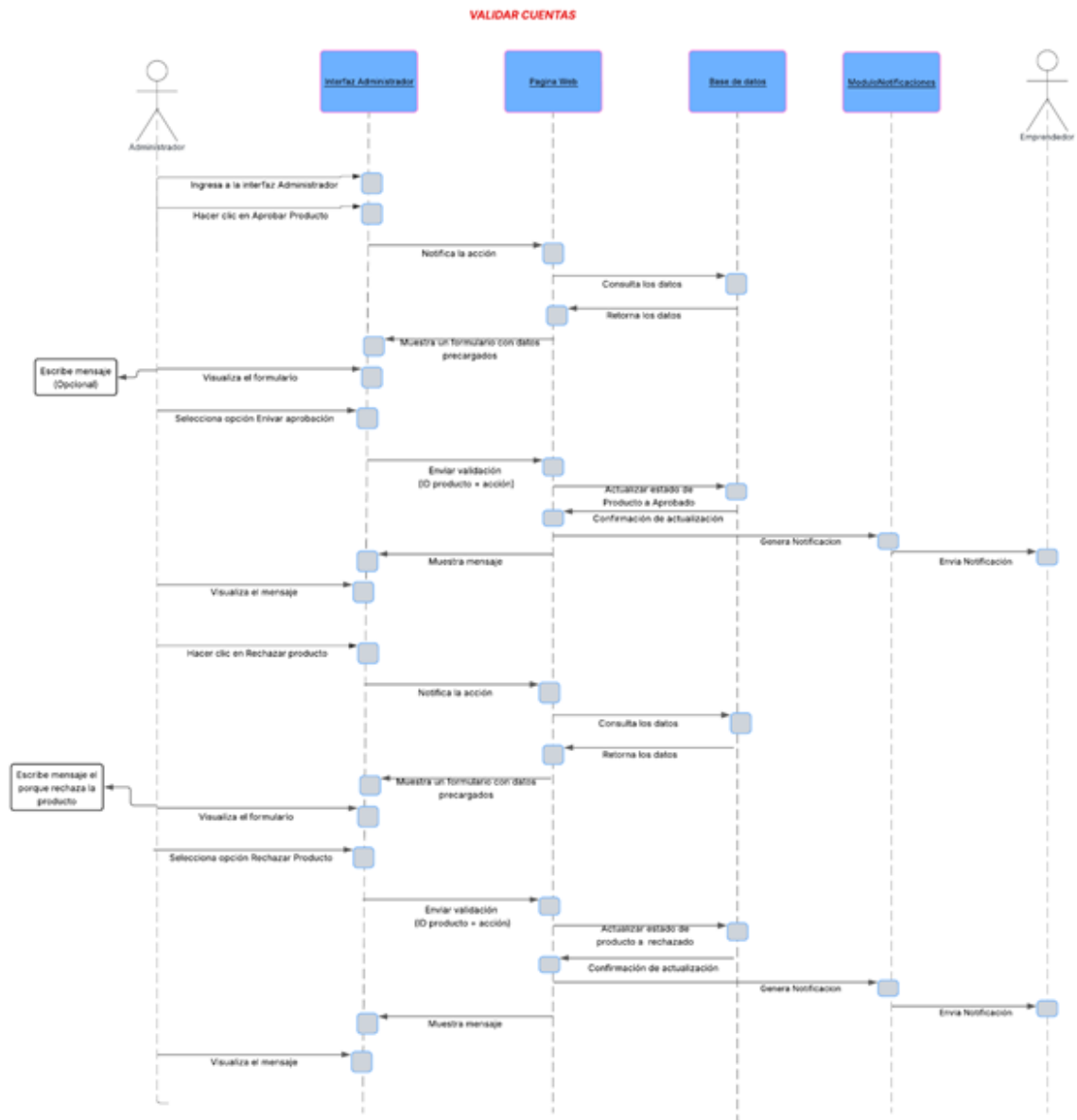
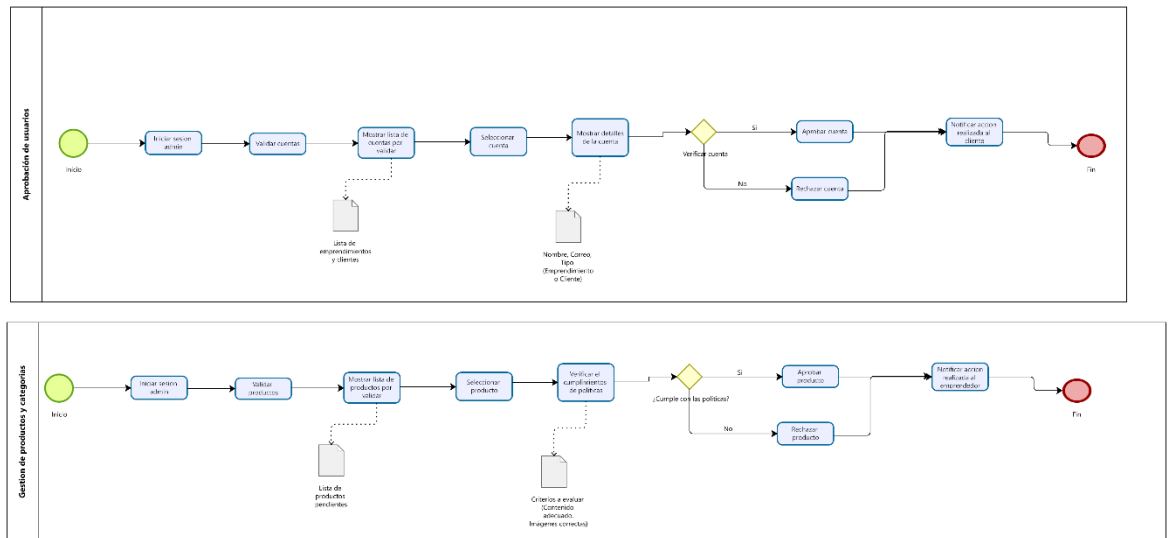


Ilustración 19. Diagrama de secuencia administrador (validar productos)

11.3.2 Diagrama de actividades

- **Administrador**



Powered by
BPM
Modeler

Ilustración 20. Diagrama de actividades administrador

- **Emprendedor**

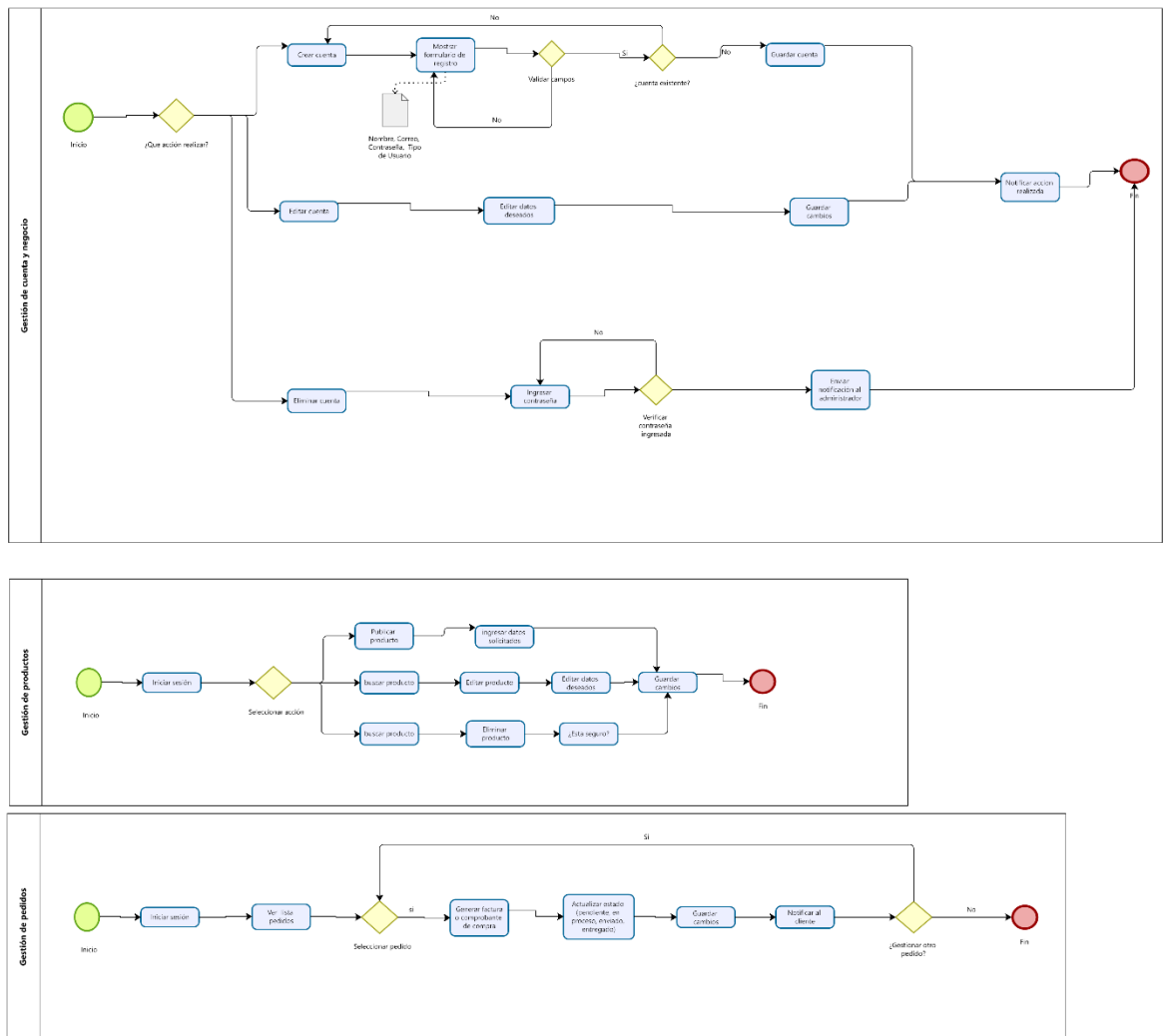


Ilustración 21. Diagrama de actividades emprendedor

- **Cliente**

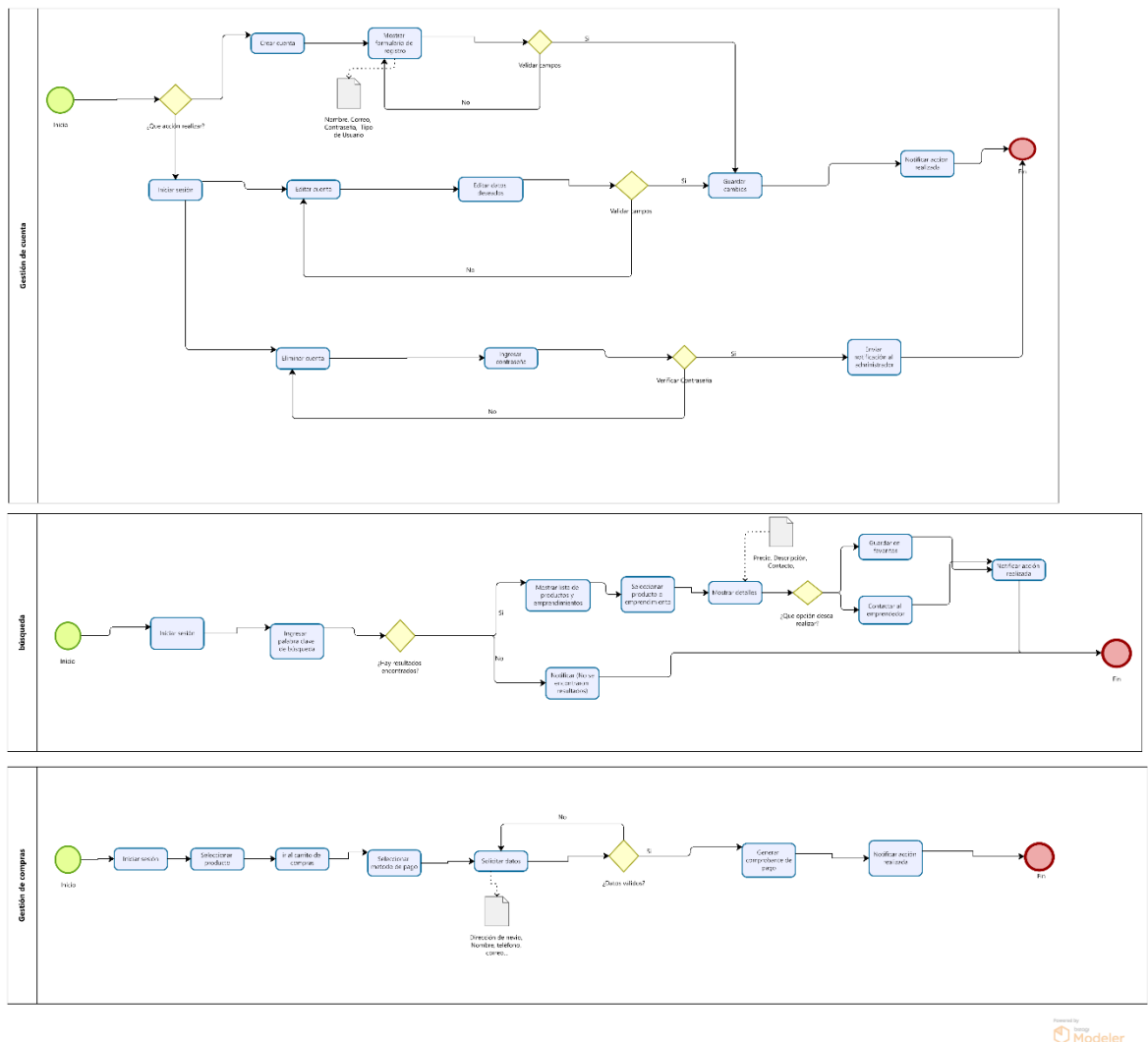


Ilustración 22. Diagrama de actividades cliente

11.4 Vista física

Muestra la infraestructura de despliegue. La aplicación se ejecuta en un servidor que aloja el backend en Python y la interfaz web. La base de datos PostgreSQL se encuentra en un servidor independiente, conectado de forma segura al backend. Los clientes acceden a través de navegadores web (Chrome, Firefox, Edge) mediante el protocolo HTTPS. Esta separación garantiza seguridad en la transmisión de datos y permite escalar el sistema a medida que crezca la demanda.

11.4.1 Diagrama de despliegue

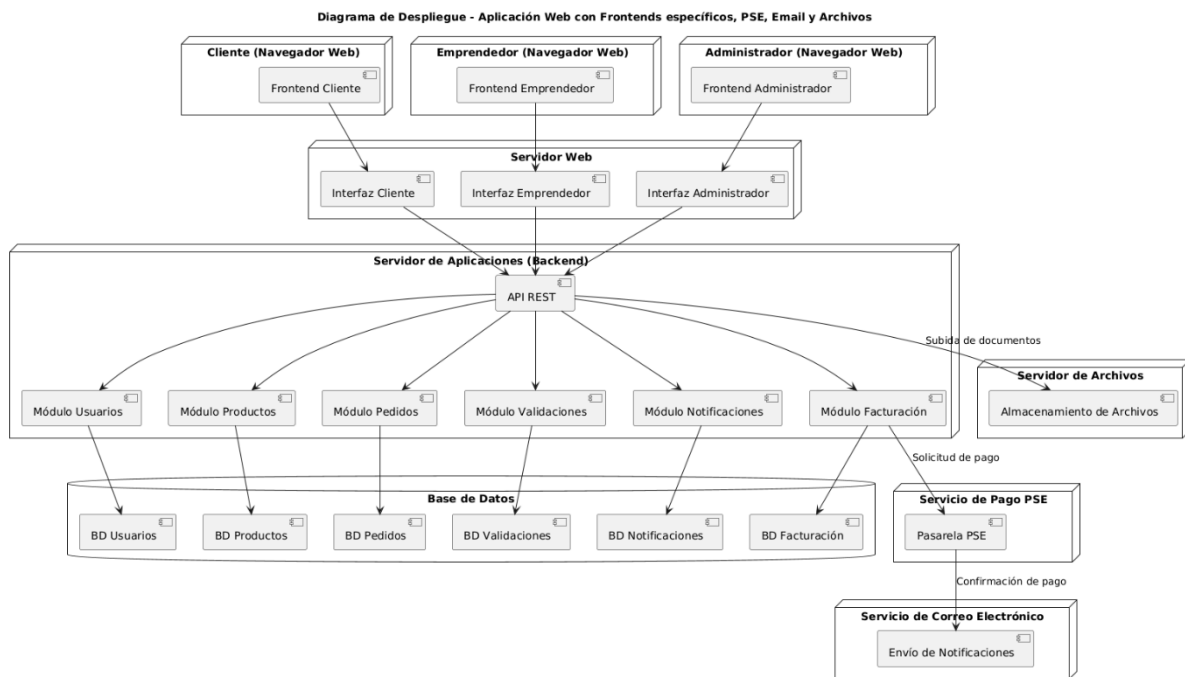


Ilustración 23. Diagrama de despliegue

11.5 Vista de escenarios

Representa la interacción de los actores con el sistema. El diagrama de casos de uso detalla cómo cada rol (cliente, emprendedor, administrador) accede a diferentes funcionalidades.

- Clientes: registrarse, iniciar sesión, buscar productos, gestionar carrito, generar pedidos y pagar.
- Emprendedores: publicar y administrar productos, consultar pedidos asociados a sus ventas.
- Administradores: gestionar usuarios, monitorear transacciones y mantener la seguridad del sistema.

Este diagrama sintetiza las funciones clave del sistema y muestra cómo se cumplen los objetivos de los usuarios finales.

11.5.1 Diagrama de casos de uso

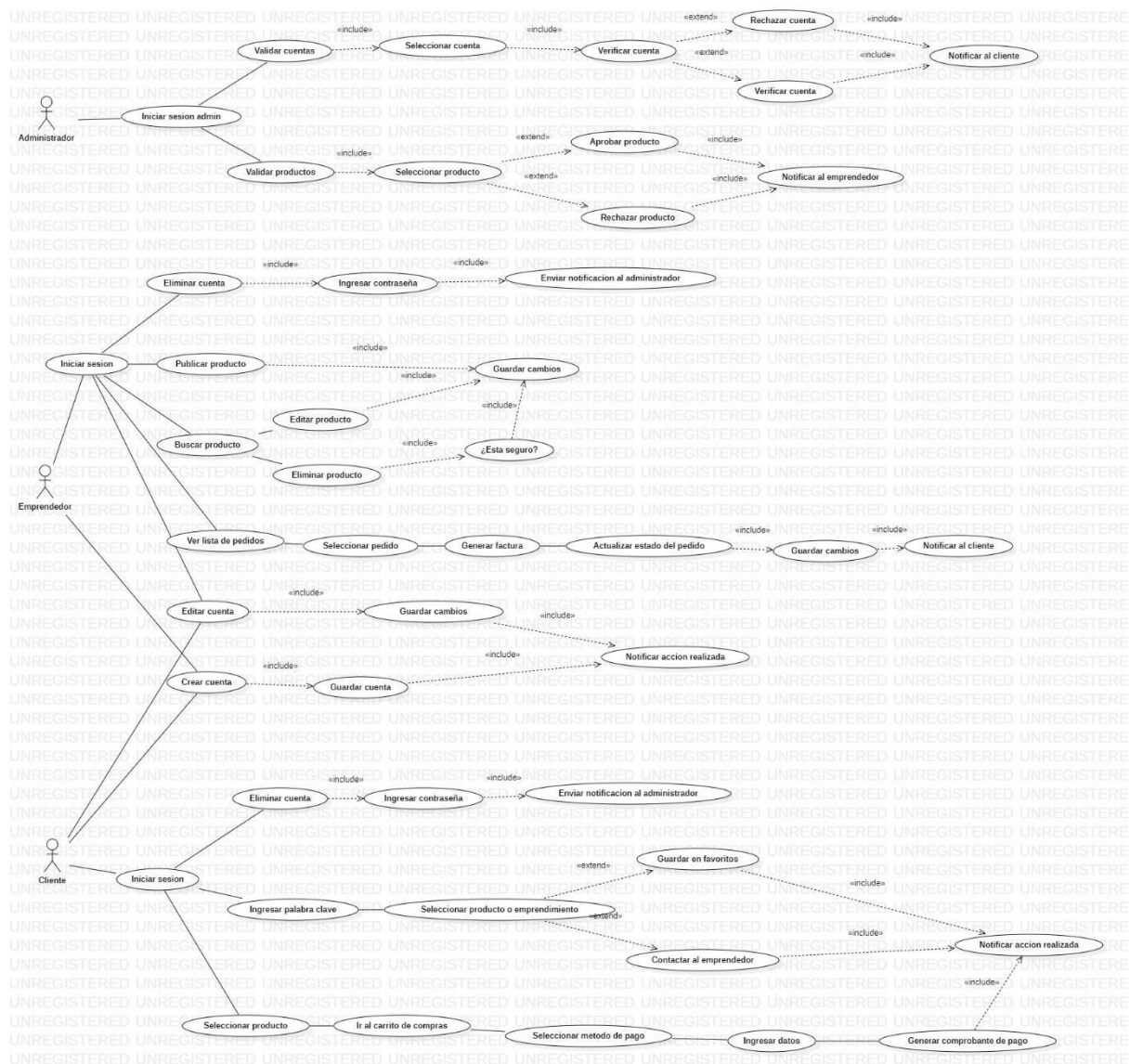


Ilustración 24. Diagrama de casos de uso

12. AMBIENTES IMPLEMENTADOS

El sistema contará con diferentes ambientes de ejecución, cada uno orientado a una etapa específica del ciclo de desarrollo. Estos ambientes permiten garantizar la calidad del software, realizar pruebas controladas y finalmente poner en operación la plataforma para los usuarios finales.

ID	Nombre	Actividades
AMB-01	Desarrollo	<ul style="list-style-type: none"> - Cada desarrolladora implementa y prueba funcionalidades en su propio entorno local con Python, HTML, CSS, JS y PostgreSQL. - Uso de repositorio compartido para control de versiones.
AMB-02	Pruebas (QA)	<ul style="list-style-type: none"> - Validación de requerimientos funcionales y no funcionales. - Ejecución de pruebas manuales y automáticas. - Pruebas de seguridad (validación de sesiones, cifrado de contraseñas). - Pruebas de carga básicas para validar desempeño en consultas y generación de pedidos.
AMB-03	Aceptación de Usuario (UAT)	<ul style="list-style-type: none"> - Validación del sistema por parte de la profesora y revisión de funcionalidades principales. - Retroalimentación para ajustes antes de pasar a producción. - Pruebas de casos de uso completos (registro, inicio de sesión, gestión de carrito, generación de pedidos y facturación).
AMB-04	Producción	<ul style="list-style-type: none"> - Implementación del sistema en un servidor con acceso web. - Habilitación de comunicación segura (HTTPS). - Base de datos PostgreSQL configurada con copias de seguridad periódicas. - Acceso a usuarios finales (clientes, emprendedores y administradores).

Tabla 11. Ambientes implementados

13. DIMENSIONAMIENTO – CRECIMIENTO

El dimensionamiento del sistema contempla la estimación del crecimiento en datos y usuarios, con el fin de anticipar necesidades de rendimiento y almacenamiento en la base de datos PostgreSQL.

Criterios definidos para el proyecto:

- Usuarios: se estima inicialmente un promedio de 25–60 usuarios (clientes, emprendedores y administradores). El sistema está preparado para escalar a más de 300 usuarios con optimizaciones de base de datos y recursos de servidor.
- Productos: la base de datos iniciará con un aproximado de 100 productos publicados. Se proyecta un crecimiento mensual en el número de productos conforme más emprendedores utilicen la plataforma.
- Pedidos: cada cliente puede generar múltiples pedidos. Se estima un crecimiento mensual en el volumen de pedidos.
- Base de datos: el repositorio de información (usuarios, productos, pedidos y facturas) podría crecer un 5–10% mensual en promedio. PostgreSQL permite gestionar este crecimiento manteniendo la integridad y rendimiento.

Medidas de soporte:

- Copias de seguridad completas semanales.
- Posibilidad de migrar a un servidor con mayor capacidad de almacenamiento si el crecimiento supera las métricas estimadas.
- Indexación de tablas críticas (Usuarios, Productos, Pedidos) para optimizar consultas frecuentes.

14. COMPONENTES HABILITADORES DEL DESARROLLO

Los componentes habilitadores corresponden a las herramientas y recursos técnicos que hicieron posible la construcción de la arquitectura del sistema, incluyendo entornos de programación, control de versiones y utilitarios de soporte.

Nombre	Descripción	Licencia	Precio Unitario 2020
Visual Studio Code	IDE utilizado para la programación en Python, HTML, CSS y JavaScript.	Open Source.	Gratuito
Git	Sistema de control de versiones distribuido para gestionar cambios en el código.	Open Source.	Gratuito

GitHub	Plataforma de repositorio remoto para trabajo colaborativo y almacenamiento del proyecto.	Open Source	Gratuito
PostgreSQL	DBMS utilizado para gestionar los datos del sistema.	Open Source	Gratuito
Python Libraries (bcrypt, hashlib)	Librerías adicionales para seguridad y cifrado de contraseñas.	Open Source	Gratuito
Navegadores (Chrome/Opera)	Entornos de ejecución y validación de la aplicación web en distintos navegadores.	Open Source	Gratuito

Tabla 12. Habilitadores de Desarrollo

15. RIESGOS DE LA ARQUITECTURA

Durante el diseño de la arquitectura se identificaron riesgos asociados al funcionamiento y despliegue del sistema. Cada riesgo se documenta con su probabilidad, impacto y estrategia de mitigación.

ID	Descripción	Probabilidad de Ocurrencia	Impacto	Mitigación
RSK-01	Fallo en el servidor donde se aloja la aplicación web.	Media	Alto	Implementar copias de seguridad periódicas de la base de datos y contar con un plan de recuperación ante desastres.
RSK-02	Acceso no autorizado a datos sensibles (usuarios, direcciones, medios de pago).	Media	Alto	Uso de cifrado de contraseñas, conexión segura HTTPS, y controles de autenticación/autorización por roles.

RSK-03	Crecimiento inesperado en el número de usuarios o productos que degrade el rendimiento de la base de datos.	Baja	Medio	Optimización de consultas SQL, creación de índices en tablas críticas y posibilidad de escalar a un servidor más potente.
RSK-04	Errores en el flujo de compra (carrito y pedidos) que afecten la experiencia del usuario.	Media	Medio	Realizar pruebas de integración y validación de escenarios completos en ambiente QA antes del paso a producción.
RSK-05	Dependencia de librerías externas (Tailwind, bcrypt, etc.) que puedan cambiar o discontinuarse.	Baja	Medio	Documentar versiones utilizadas y mantener plan de actualización periódica para garantizar compatibilidad.

Tabla 13. Riesgos de la Arquitectura

16. CONCLUSIÓN

El desarrollo de este documento de arquitectura permitió organizar y dar sentido al proyecto de manera integral, abarcando desde la definición de interesados y requerimientos hasta las decisiones técnicas que soportan la implementación de la plataforma. Documentar cada etapa hizo posible entender que la arquitectura no se limita a elegir un lenguaje o una base de datos, sino que constituye la guía que orienta el diseño, el desarrollo y la futura evolución del sistema.

A lo largo del proceso se logró plasmar cómo los patrones de diseño (Factory, Composite y State) y las tecnologías seleccionadas (Python, HTML, CSS, JavaScript, TailwindCSS y PostgreSQL) se integran para responder a las necesidades de los diferentes usuarios de la aplicación. De igual forma, las vistas arquitectónicas ofrecieron una visión clara de la estructura lógica, el flujo de procesos, la organización de los componentes y la interacción de los actores con el sistema.

Otro aspecto valioso fue la reflexión sobre temas como seguridad, trazabilidad y crecimiento futuro, lo cual permitió anticipar riesgos y proponer medidas que aumentan la confiabilidad del sistema. De esta manera, se sientan bases sólidas no sólo para cumplir con los requerimientos actuales, sino también para que la solución pueda ampliarse y adaptarse en el tiempo.

En conclusión, este trabajo evidenció la importancia de planificar y documentar antes de programar, y de mantener una visión clara de cómo cada decisión técnica impacta en la experiencia del usuario y en la calidad del software. Más que un requisito académico, este documento representa un aprendizaje significativo sobre cómo construir sistemas de forma estructurada, colaborativa y con miras a la sostenibilidad tecnológica.