



MI FACTURA PR

MIKHAEL HASHEM K. TIBA
JOHNNY MERCADO RUIZ
YENIEL G PÉREZ MERCADO

FECHA: 20 DE MAYO DE 2025

INGENIERÍA DE SOFTWARE II
COMP 3402

INTRODUCCION

Este proyecto consiste en el desarrollo de una aplicación web para la gestión de facturas de servicios públicos (agua y luz). El objetivo principal es facilitar a los usuarios el acceso seguro y organizado a sus facturas mediante una interfaz amigable, accesible desde cualquier dispositivo con conexión a internet.

¿QUÉ PROBLEMA RESUELVE NUESTRO PROYECTO?

- Actualmente los usuarios deben acceder diferentes plataformas para consultar y gestionar las facturas de AAA y LUMA. Esto genera confusión, pérdida de tiempo y aumenta riesgo de errores al tener que recordar múltiples credenciales.
- Por otro lado, sin una herramienta centralizada es común olvidar las fechas de vencimientos de facturas, lo que puede conllevar a retrasos en los pagos y acumulación de recargos.

ROLES DE CADA INTEGRANTE

- Yeniel G Perez (Frontend): Diseñó y programó la interfaz visual del sistema, asegurando una experiencia de usuario clara y funcional.
- Mikael Hashem K. Tiba (Backend y datos): Desarrolló la lógica del servidor y la base de datos, gestionando el procesamiento y almacenamiento de la información.
- Johnny Mercado (Backend y datos): Colaboró en la programación del servidor y la base de datos, facilitando la conexión entre la interfaz y los datos.

ARQUITECTURA DEL SOFTWARE

Descripción general:

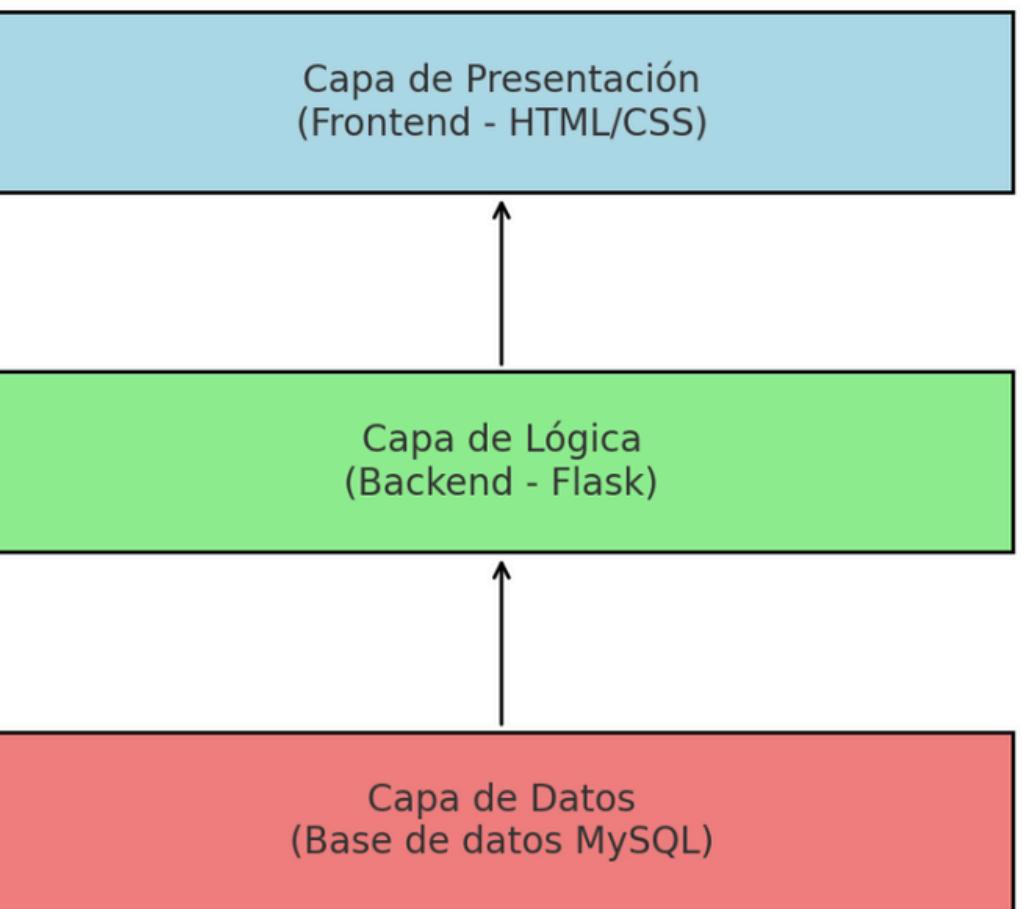
- El sistema utiliza una arquitectura 3-Tier (Tres capas), separando claramente la interfaz de usuario (frontend), la lógica de negocio (backend con Flask) y la capa de datos (base de datos MySQL).

Justificación:

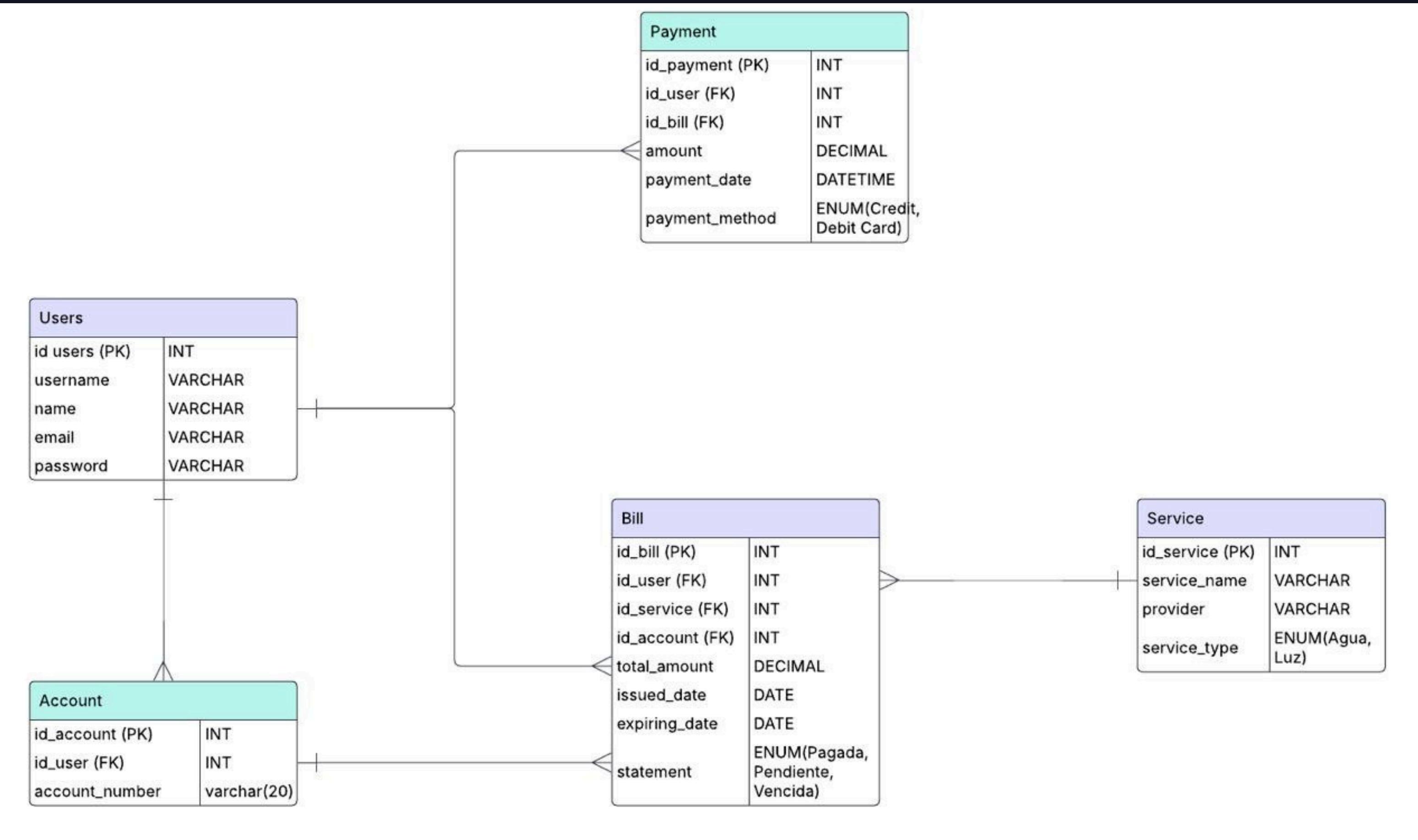
- Se eligió la arquitectura 3-Tier por su simplicidad, mantenimiento claro y separación de responsabilidades. Esta estructura permite un desarrollo más organizado, facilita el trabajo en equipo y hace más sencillo escalar o modificar componentes individuales sin afectar todo el sistema.

ARQUITECTURA DE LA APLICACIÓN

(3-Tier)



BASE DE DATOS

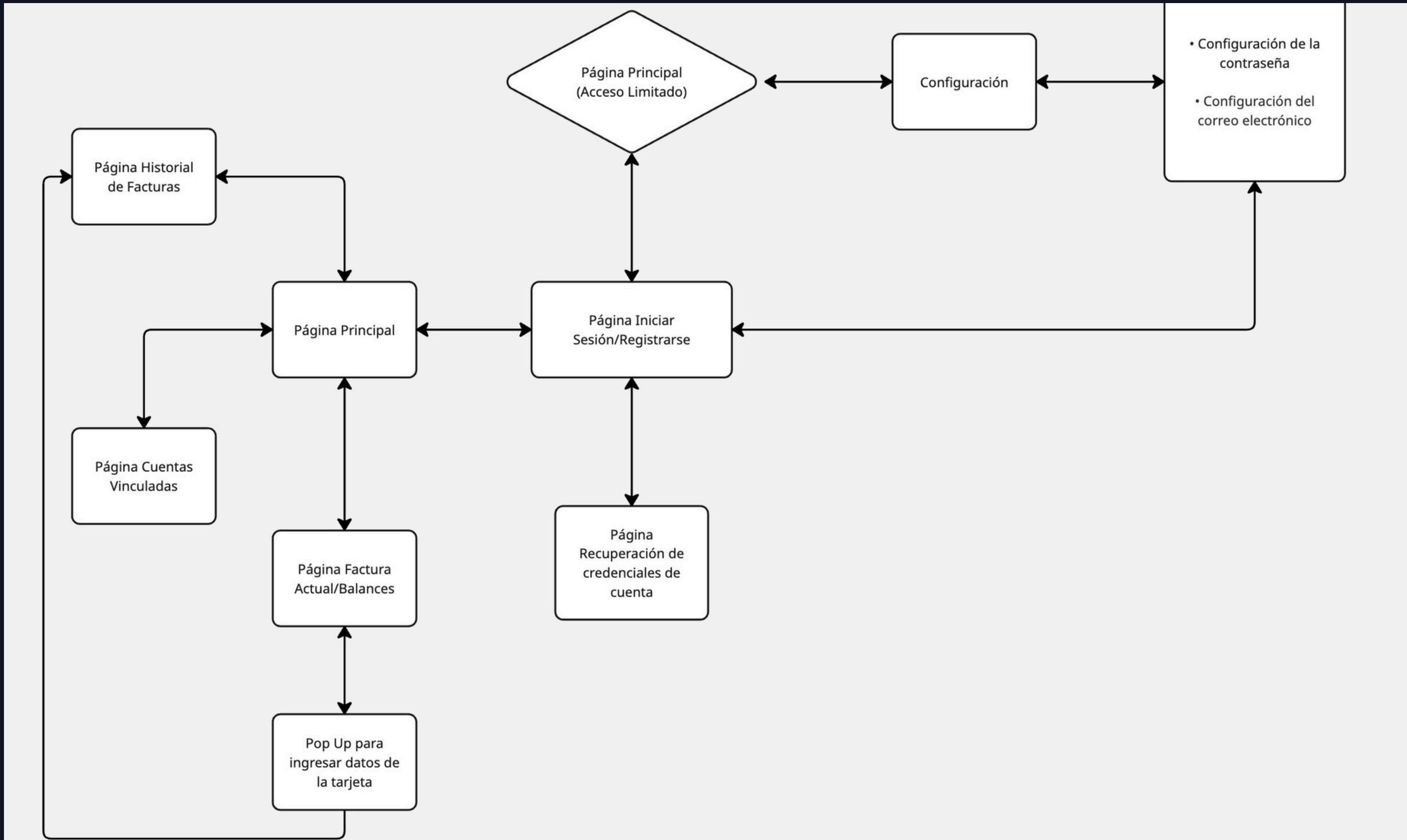


BASE DE DATOS

EXPLICACIÓN DE TABLAS PRINCIPALES

- USERS: GUARDA INFORMACIÓN DE LOS USUARIOS REGISTRADOS (NOMBRE, EMAIL, USERNAME, PASSWORD).
- ACCOUNT: REPRESENTA LAS CUENTAS ASOCIADAS A CADA USUARIO, IDENTIFICADAS POR UN NÚMERO ÚNICO.
- BILL: CONTIENE LAS FACTURAS DE CADA CUENTA, INDICANDO EL MONTO, FECHA DE EMISIÓN, VENCIMIENTO Y ESTADO (PAGADA, PENDIENTE, VENCIDA).
- SERVICE: DEFINE LOS SERVICIOS DISPONIBLES (EJ. AGUA O LUZ), SU NOMBRE Y PROVEEDOR.
- PAYMENT: REGISTRA LOS PAGOS REALIZADOS POR LOS USUARIOS, INCLUYENDO MONTO, MÉTODO Y FACTURA ASOCIADA.

FLUJO DE PANTALLAS



EXPLICACIÓN DE FLUJOGRAMA

- INICIO DE SESIÓN / REGISTRO: ACCESO SEGURO AL SISTEMA.
- PÁGINA PRINCIPAL: CENTRO DE NAVEGACIÓN CON ACCESO A FACTURAS, CUENTAS Y CONFIGURACIÓN.
- FACTURAS: VISUALIZACIÓN DE BALANCES ACTUALES Y FACTURAS ANTERIORES.
- PAGOS: POP-UP PARA INGRESAR DATOS DE TARJETA.
- CUENTAS VINCULADAS: MANEJO DE MÚLTIPLES CUENTAS DE USUARIO.
- CONFIGURACIÓN: CAMBIO DE CONTRASEÑA Y CORREO ELECTRÓNICO.
- RECUPERACIÓN DE CUENTA: SOPORTE PARA CREDENCIALES OLVIDADAS.

TECNOLOGIAS UTILIZADAS

La aplicación tiene una arquitectura monolítica basada en Flask y sigue el patrón MVC (Modelo-Vista-Controlador). El backend está desarrollado en Python, utiliza MySQL como base de datos y cuenta con una interfaz construida con HTML, CSS y JavaScript.

Lenguaje de Programación

- Frontend: HTML, CSS, Bootstrap, Jinja2 y JavaScript
- Backend: Flask (Python)
- Database: MySQL/MariaDB

Programas Utilizados

- Figma
- Visual Studio Code

REFLEXIÓN DEL TRABAJO

DESAFÍOS ENCONTRADOS

- INTEGRAR CORRECTAMENTE EL FRONTEND CON EL BACKEND USANDO RUTAS DE FLASK.
- MANEJO DE ERRORES DE AUTENTICACIÓN Y VALIDACIÓN DE USUARIOS.
- CONEXIÓN ESTABLE Y SEGURA CON LA BASE DE DATOS MYSQL, ESPECIALMENTE AL MANEJAR MÚLTIPLES USUARIOS

SOLUCIONES IMPLEMENTADAS

- SE APLICÓ LA ARQUITECTURA 3-TIER PARA SEPARAR RESPONSABILIDADES Y FACILITAR EL DESARROLLO EN EQUIPO.
- SE UTILIZÓ FLASK-LOGIN Y SESIONES PARA CONTROLAR EL ACCESO DE USUARIOS.
- SE IMPLEMENTÓ UNA ESTRUCTURA MODULAR EN EL CÓDIGO PARA FACILITAR PRUEBAS Y MANTENIMIENTO.

RELACIÓN CON LOS TEMAS DE CLASE

- APLICAMOS PRINCIPIOS DE INGENIERÍA DE SOFTWARE, COMO MODULARIDAD, SEPARACIÓN DE CAPAS Y TRABAJO COLABORATIVO.
- USAMOS HERRAMIENTAS Y CONCEPTOS DISCUTIDOS COMO SCRUM, BACKLOG DE PRODUCTO Y CONTROL DE VERSIONES.
- SE REFORZÓ LA IMPORTANCIA DE DOCUMENTAR EL CÓDIGO Y PLANIFICAR EL DESARROLLO, TAL COMO SE ENFATIZÓ EN EL CURSO.

CONCLUSIÓN

En conclusión el desarrollo de “Mi Factura PR” nos permitió aplicar de manera práctica los conocimientos adquiridos en el curso de Ingeniería de Software II. A través del trabajo colaborativo, logramos integrar nuestras habilidades individuales en un proyecto común, demostrando la importancia de la planificación, la comunicación efectiva y la asignación clara de roles. Al implementar herramientas como el sprint planning y enfocarnos en la usabilidad y experiencia del usuario, pudimos diseñar una solución que responde a una necesidad real: centralizar la gestión de facturas de servicios esenciales. Una implementación o actualización que queremos añadir la gestión de notificaciones. Esta experiencia no solo fortaleció nuestras competencias técnicas, sino también nuestra capacidad de trabajar como un equipo sólido orientado a resultados.



MI FACTURA PR