Contents

[Introducción 2](#_Toc197326030)

[Requerimientos Funcionales 3](#_Toc197326031)

[Frontend: 3](#_Toc197326032)

[Backend 3](#_Toc197326033)

[Requerimientos Generales 4](#_Toc197326034)

[Tecnologías implementadas en el Proyecto 5](#_Toc197326035)

[Frontend: 5](#_Toc197326036)

[Backend: 5](#_Toc197326037)

[Plan de Trabajo 6](#_Toc197326038)

[Semana 1: 6](#_Toc197326039)

[Semana 2: 6](#_Toc197326040)

[Semana 3: 6](#_Toc197326041)

[Semana 4 7](#_Toc197326042)

[Semana 5 7](#_Toc197326043)

[Semana 6 7](#_Toc197326044)

[Semana 7 7](#_Toc197326045)

[Semana 8 8](#_Toc197326046)

[Semana 9 8](#_Toc197326047)

[Semana 10 8](#_Toc197326048)

[Semana 11-12 8](#_Toc197326049)

# Introducción

La **Aplicación Web de Gestión Residencial** es una solución integral diseñada para optimizar la administración de complejos residenciales. Esta herramienta facilita la gestión de pagos, la atención de reclamos, la generación de reportes de averías y el control financiero, proporcionando una plataforma centralizada y eficiente para administradores y residentes.

# Requerimientos Funcionales

## Frontend:

* Panel de usuario para registrar pagos y consultar su historial.
* Sección para enviar reclamos con evidencia (texto, imágenes)
* Notificaciones de mantenimiento programado y recordatorio de pagos
* Dashboard administrativo para gestionar reportes y reclamos
* Gráficos interactivos sobre ingresos y gastos residenciales

## Backend

* Base de datos para pagos, reclamos, reportes y usuarios.
* Algoritmo para calcular cuotas mensuales y generar recibos automáticos
* API para integrar servicios de pago en línea
* Sistema de estadística para evaluar el estado financiero de la comunidad
* Gestión de roles (administrador, residente)

# Requerimientos Generales

* Diseño Responsivo (adaptado a cada pantalla)
* Login y Registro de Usuario
* Aplicar integración de inteligencia artificial
* Manual técnico y Manual de Usuario (PDF)
* Documentación del Progreso del desarrollo (inicio, fin)
* CRUD de todas las entidades de la aplicación ej.: Categoría, Roles, Usuario)
* Permisos de Usuario
* UI/ UX
* Diseño de reportes o informe de resultado en cada proyecto, Ejemplo: ver las ventas del último mes…

# Tecnologías implementadas en el Proyecto

## Frontend:

* **Tecnologías base:**
  + HTML
  + CSS
  + JavaScript
* **Librerías y herramientas:**
  + Tailwind CSS
  + Flowbite
  + DaisyUI
  + Google Fonts
  + Google Icons
  + Heroicons
  + SailboatUI
  + MerakiUI
  + Alpine.js
  + Undraw
  + Jinja2

## Backend:

* **Tecnologías base:**
  + Flask (Python)
  + MySQL
* **Librerias y Herramientas**
  + Bcrypt
  + Mysql-connector
  + Gmail (API)

# Plan de Trabajo

## Semana 1:

En la primera semana del desarrollo, estuve analizando los requisitos mas profundamente, y tratando de determinar como seria el flujo de mi aplicación, además de las tecnologías y librerías que iba usar para UI y UX, esta semana fue para pensar sobre la estructura de mi proyecto.

## Semana 2:

En esta semana, estaba desarrollando como seria la autenticación que es verificación, identificación y acceso que tendrán los usuarios a mi aplicación, y la autorización que son los permisos que les dare a los supuestos usuarios.

Cree las primeras 3 tablas(users,roles,user\_roles), esta almacena los usuarios, los roles, y los roles que se le asignan a cada uno.

Y para culminar la semana hice el login, y las 2 pantallas que tendrá cada rol, para el rol de admin hice un Panel Administrativo que me permitirá gestionar mejor los datos, y diseñando como será la pantalla del residente.

## Semana 3:

Continuando de lo hecho en la segunda semana, le dedique tiempo tanto al diseño usando modales, y componentes para una mejor experiencia de usuario como a la logica, manipulado los datos de una manera eficiente para la gestión de usuarios con un CRUD(Create,Read,Update,Delete), pero antes de eliminar cada usuario, los deshabilito para mantener los datos y no desaparecerlos, trabajando de una manera un poco mas profesional.

## Semana 4

Una semana un poco menos ocupada, desarrollando una de las funcionalidades principales de mi aplicación, las residencias. Estuve trabajando en la creación de las tablas apartments y houses, y en la asignación de residencias si los usuarios contienen el rol residente, y al remover el rol, desalojarlos de su residencia, además de eso inicié a trabajar con los crud de las residencias.

## Semana 5

Culminando las residencias, empecé a trabajar en la pantalla del residente, y lo primero que hice es que el pueda editar su información, correo y número de teléfono. Además de eso el podía editar la contraseña con una validación para contraseña segura. De inmediato empecé a trabajar con los reclamos.

## Semana 6

Haciendo los reclamos, el residente me podía enviar texto y evidencia(imágenes) del supuesto reclamo. Los reclamos por defecto al crearlos el estado eran “Pendiente”, luego de que el administrador lo atendía y ponía un día y una fecha y tiempo de inicio, y se actualizaba a “Atendido”, cuando llegaba la hora de inicio y la fecha, pasaba a “En Progreso”, y el admin podía culminarla. En cada cambio de estado del reclamo, le llegaba una notificación al residente.

## Semana 7

Empezando los pagos, tenía como requisito los pagos en efectivo y por transferencia con evidencia y pude analizar que antes de pagar, los residentes debían tener deudas, así que mis 3 tablas creadas fueron payments(pagos), debts(deudas), transfer\_requests (solicitudes de transferencia), y pude empezar registrando pagos en efectivo, insertando las deudas manualmente.

## Semana 8

Continuando los pagos, pude hacer que un residente mandara un comprobante de transferencia y el admin lo recibía y manualmente seleccionaba las deudas que tenía para marcarlas como pagadas. Además de eso aplique facturas a los pagos, y cuando el residente paga por transferencia le llega su factura al correo.

## Semana 9

Casi culminando el proyecto, estaba haciendo algo extra en la sección de reclamos del lado del residente, que cuando su reclamo termine, pueda calificarlo con un componente de rating con estrellas, además de eso, hice que un admin pueda registrar un egreso y le dedique tiempo a la sección principal del admin donde se mostrara un resumen de todos los datos de la aplicación como, Balance, Reclamos pendientes, Solicitudes de transferencia pendientes, estadísticas de ingreso y egreso, promedio del rating de los reclamos y recordatorios de pago.

## Semana 10

En esta semana estaba haciendo la sección de configuración donde el administrador puede editar el logo de la app, los precios de los pagos que se van a generar, y su información, además de eso me mantuve haciendo optimizaciones en la app.

## Semana 11-12

Hice que los pagos se generaran mensualmente y cuando yo agregara a un residente, o editara a algún admin a residente también, y también hice el forgot your password, se basa en que pones tu correo en un form y te llega una confirmación al correo donde debes poner tu nueva contraseña y confirmarla, con un token único y temporal por 30 minutos.