**FASE 2**

**DISEÑO E IMPLEMENTACIÓN DEL PROYECTO**

**Nombre de Proyecto**

Escuela de Informática y Telecomunicaciones

Mes 2025

1. Identificación del Proyecto

| **Nombre de Proyecto** |
| --- |
| Corredora |

1. Integrantes del Equipo de Trabajo

| **N°** | **Rut** | **Apellidos** | **Nombres** |
| --- | --- | --- | --- |
| 1 | 20.919.002-8 | Balboa | Fabián |
| 2 | 20.481.688-3 | Zamora | Cristián |
| 3 | 21.452.371-k | Rocha | Joaquín |

1. Registro de Control de Cambios

| **Revisión** | **Fecha** | **Paginas** | **Descripción del Cambio** | **Autor** |
| --- | --- | --- | --- | --- |
| 1 | 14-09-25 | Todas | Cambios según primera revisión docente (alcances y objetivos). | Joaquín Rocha |
| 2 | 29-09-25 | Todas | Cambios generales en el ATP, tiempo verbal, objetivos específicos, definir criterios de probabilidad | Cristián Zamora |
| 3 | 07-10-25 | Todas | Últimos cambios de formalidad del informe y agregación de índices | Fabián  Balboa |

**ÍNDICE DE CONTENIDO**

[LISTA DE TABLAS 4](#_heading=h.yrnotmfgkhdu)

[LISTA DE FIGURAS 5](#_heading=h.eyy6nq6gizj6)

[LISTA DE DIAGRAMAS 6](#_heading=h.6okkd9muocuq)

[**GLOSARIO 7**](#_heading=h.kdm67scrj4tj)

[1 Diseño e Implementación del Proyecto 8](#_heading=h.css8ibtk49tc)

[1.1 Resumen 8](#_heading=h.r9ylnrkd2bov)

[1.2 Abstract 9](#_heading=h.25nxpo2unnpq)

[1.2 Introducción 10](#_heading=h.o1hqha6p226i)

[1.3 Problemática a solucionar o necesidad a satisfacer 11](#_heading=h.xt1ifqn1ewqn)

[1.4 Objetivos del Proyecto (general y específicos) 12](#_heading=h.jv45hvbf5j33)

[1.5 Competencias del Perfil de Egreso 14](#_heading=h.8sizz9ttwydh)

[1.6 Asignación de roles 15](#_heading=h.vryhc3puzavc)

[1.7 Metodología utilizada en el Proyecto. 16](#_heading=h.4qw8fhq3h6gd)

[1.8 Creación de cronograma asociado al Proyecto (Carta Gantt) 18](#_heading=h.ld0t9ok0fu7c)

[1.9 Riesgos Asociados al Proyecto 19](#_heading=h.rphz11hqcsi)

[1.10 Implementación del Proyecto 21](#_heading=h.m76epxiztl4m)

[Diseño y Arquitectura de la solución (Caso de uso de Software o plataforma de gestión) 21](#_heading=h.4sp3so3hlpz0)

[Requerimientos técnicos 21](#_heading=h.gh40555ycswx)

[Desarrollo de la solución 21](#_heading=h.qoz6yl5wukd4)

[Resultados de la solución 22](#_heading=h.cx9cwnv3ieah)

[1.11 Definición de Recursos y Costos asociados al Proyecto 23](#_heading=h.tttmjivvoi1y)

[CONCLUSIÓN 24](#_heading=h.dxtsdq9vr0y7)

[BIBLIOGRAFÍA 25](#_heading=h.z2yia39dn2u6)

[ANEXOS 26](#_heading=h.p1bevpfbm3je)

# LISTA DE TABLAS

[Matriz de Riesgos 17](#_heading=h.bz1snhdumag8)

[Justificación de probabilidad de riesgos 18](#_heading=h.fl114nr8mbtl)

# LISTA DE FIGURAS

# LISTA DE DIAGRAMAS

[Carta Gantt 18](#_heading=h.3sz7fyxxm1x6)

# GLOSARIO

**Automatización:** Proceso mediante el cual se sustituyen tareas manuales por procedimientos tecnológicos que permiten ahorrar tiempo, reducir errores y aumentar la eficiencia.

**Base de datos relacional:** Sistema que organiza la información en tablas relacionadas entre sí, lo que facilita la integridad y la consulta de datos.

**Comisión:** Porcentaje de dinero que recibe un corredor de propiedades por la venta o arriendo de un inmueble.

**Contrato:** Documento legal que establece las condiciones de una compraventa o arriendo entre clientes y la corredora.

**Escalabilidad:** Capacidad de un sistema de software para adaptarse al crecimiento de la empresa, aumentando usuarios, datos o funcionalidades sin perder rendimiento.

**Gestión de propiedades:** Conjunto de actividades relacionadas con la administración, control y registro de bienes raíces disponibles para arriendo o venta.

**Interfaz de usuario (UI):** Medio visual a través del cual los usuarios interactúan con el software, compuesto por pantallas, botones y formularios.

**Metodología en cascada:** Modelo de desarrollo de software lineal y secuencial, compuesto por fases claramente definidas (análisis, diseño, implementación, pruebas y despliegue).

**PostgreSQL:** Sistema de gestión de bases de datos relacional de código abierto, conocido por su estabilidad, seguridad y capacidad de manejar grandes volúmenes de datos.

**Reportes de gestión:** Documentos o visualizaciones generadas por el sistema que resumen información clave para apoyar la toma de decisiones en la empresa.

**Software de escritorio:** Aplicación informática que se instala y ejecuta directamente en un computador, a diferencia de las aplicaciones web o móviles.

**Trazabilidad:** Capacidad de seguir el historial y el estado de la información o procesos dentro del sistema, asegurando control y transparencia en las operaciones.

# 

# Diseño e Implementación del Proyecto

## Resumen

El proyecto **“Corredora”** tuvo como objetivo optimizar la gestión de una corredora de propiedades en **La Calera, Región de Valparaíso**. La empresa enfrentaba dificultades porque gran parte de sus procesos eran manuales, lo que generaba duplicidad de información, pérdida de tiempo y limitaciones en el acceso a datos. Para solucionarlo, se propuso un **software de escritorio** que centralizara y automatizara la administración de propiedades, clientes, RRHH y finanzas, además del control de comisiones y la generación de reportes de gestión.

Los principales requerimientos del sistema fueron:

* Centralizar la gestión de propiedades y clientes en una sola plataforma.
* Registrar y administrar operaciones de ventas y arriendos.
* Incorporar herramientas de control y seguimiento de procesos.
* Generar reportes internos para apoyar la toma de decisiones.
* Utilizar una base de datos **PostgreSQL** que asegurara integridad, seguridad y escalabilidad.

Para cumplir con estos requerimientos, el proyecto se desarrolló bajo la **metodología en cascada**, avanzando de forma ordenada a través de etapas definidas:

* **Análisis de requerimientos:** levantamiento de información y definición de necesidades del negocio.
* **Diseño:** construcción del modelo de datos en PostgreSQL y creación de prototipos de interfaz.
* **Implementación:** desarrollo de los módulos principales (gestión de propiedades, clientes, contratos y reportes).
* **Pruebas:** verificación de calidad mediante pruebas unitarias e integradas.
* **Despliegue:** entrega de la solución y validación en el entorno real.

El proyecto se planificó con una duración aproximada de **tres meses**, utilizando recursos accesibles como computadores personales, PostgreSQL y herramientas de programación. Se proyectó entregar una solución confiable que mejorara la productividad, redujera los tiempos de gestión y garantizara la seguridad de la información.

En conclusión, esta iniciativa respondió a una necesidad concreta de modernización tecnológica y constituyó un espacio de aplicación práctica de competencias en programación, bases de datos, aseguramiento de calidad y gestión de proyectos.

## 1.2 Abstract

The **“Corredora”** project aimed to improve the management of a real estate agency in **La Calera, Valparaíso Region**. The company faced challenges due to largely manual processes, which led to data duplication, time inefficiencies, and limited data access. To address this, a **desktop software** solution was proposed to centralize and automate the management of properties, clients, human resources, and finances, while also providing commission control and management report generation.

The main system requirements were:

* Centralize property and client management on a single platform.
* Register and manage sales and rental operations.
* Incorporate process monitoring and control tools.
* Generate internal reports to support decision-making.
* Use a **PostgreSQL** database to ensure integrity, security, and scalability.

To meet these requirements, the project followed the **waterfall methodology**, progressing through well-defined stages:

* **Requirements analysis.**
* **Design** of the database and user interfaces.
* **Implementation** of core modules (properties, clients, contracts, and reports).
* **Testing** through unit and integration tests.
* **Deployment** and validation in the real environment.

The project was planned to be completed in approximately **three months**, using accessible resources such as personal computers, PostgreSQL, and programming tools. It was expected to deliver a reliable solution that improved productivity, reduced management times, and ensured data security.

In conclusion, this initiative addressed a concrete need for technological modernization and constituted a practical opportunity to apply competencies in programming, databases, software quality assurance, and project management.

## Introducción

El proyecto “Corredora” se desarrolló en el contexto de una agencia inmobiliaria ubicada en La Calera, Región de Valparaíso, dedicada a la gestión de propiedades para venta y arriendo. La empresa enfrentaba una serie de dificultades operativas debido a que gran parte de sus procesos se ejecutaban de forma manual, mediante registros en hojas de cálculo, documentos físicos y distintos sistemas sin integración entre sí. Esta situación generaba una administración poco eficiente, con duplicidad de datos, pérdida de información relevante y una limitada capacidad para acceder a datos actualizados en tiempo real.

La necesidad de modernización tecnológica se hizo evidente a medida que la corredora experimentó un aumento sostenido en el número de propiedades y clientes gestionados. Los procedimientos manuales comenzaron a representar una barrera significativa para mantener la calidad del servicio y responder oportunamente a las demandas del mercado inmobiliario. Además, la ausencia de herramientas automatizadas dificultaba la trazabilidad de las operaciones, el control de comisiones y la generación de reportes de gestión necesarios para la toma de decisiones estratégicas.

Ante este escenario, el proyecto tuvo como propósito diseñar e implementar un software de escritorio que centralizará la información y automatizará los principales procesos internos de la corredora. Este sistema buscó optimizar la administración de propiedades, clientes, contratos, recursos humanos y finanzas, mejorando la eficiencia operativa y la confiabilidad de los datos. Al incorporar una base de datos relacional en PostgreSQL, se garantizó la integridad, seguridad y escalabilidad de la información, permitiendo además que el sistema pueda adaptarse al crecimiento futuro de la empresa.

El desarrollo del proyecto se estructuró bajo la metodología tradicional en cascada, la cual permitió avanzar de manera ordenada y secuencial a través de las etapas de análisis, diseño, implementación, pruebas y despliegue. Esta metodología resultó adecuada debido a la naturaleza del proyecto, que requería un control riguroso de las fases y una validación completa antes de pasar a la siguiente etapa, reduciendo la probabilidad de errores y facilitando el seguimiento del progreso.

Durante el proceso, se aplicaron diversas competencias técnicas propias del perfil del ingeniero informático, como el levantamiento de requerimientos, el diseño de modelos de datos, la programación orientada a objetos, el manejo de consultas SQL y la aplicación de pruebas de calidad. Asimismo, se utilizaron herramientas de desarrollo modernas como Python con PySide6, Visual Studio Code, Git/GitHub para control de versiones, y Supabase como entorno de respaldo de base de datos, garantizando un flujo de trabajo profesional y eficiente.

La ejecución del proyecto representó una experiencia integral de aplicación de conocimientos teóricos y prácticos, orientada a resolver una necesidad real del sector inmobiliario. La implementación del software permitió reducir tiempos de gestión, minimizar errores humanos, mejorar la trazabilidad de la información y facilitar la toma de decisiones basadas en datos confiables. En términos generales, este proyecto contribuyó al fortalecimiento tecnológico de la corredora, posicionándose de manera más competitiva dentro del mercado local y demostrando la relevancia del desarrollo de soluciones informáticas personalizadas para pequeñas y medianas empresas.

## Problemática a solucionar o necesidad a satisfacer

La corredora de propiedades “La Calera” enfrenta múltiples dificultades en su gestión diaria debido a la prevalencia de procesos manuales y desorganizados, lo que afecta directamente su eficiencia operativa y la calidad del servicio ofrecido a los clientes. Entre los principales problemas se identifican:

1. **Duplicidad y pérdida de información:** La ausencia de un sistema centralizado provoca que los datos de propiedades, clientes y contratos se registren en distintos documentos o formatos, ya sea en hojas de cálculo, archivos físicos o registros dispersos. Esta situación genera errores frecuentes, pérdida de información valiosa y dificultades para mantener un historial confiable de las operaciones, afectando la continuidad de los procesos y la trazabilidad de los datos.
2. **Ineficiencia operativa:** Muchas tareas administrativas, como el registro de ventas, arriendos, control de contratos o cálculo de comisiones, requieren un consumo de tiempo excesivo por parte del personal. Esta situación no solo reduce la productividad general, sino que también incrementa la posibilidad de errores humanos y retrabajos, limitando la capacidad de la empresa para enfocarse en actividades estratégicas o de atención al cliente.
3. **Limitaciones en la toma de decisiones:** La información clave sobre propiedades, clientes y contratos no siempre se encuentra disponible de manera rápida ni confiable, lo que dificulta la planificación estratégica, el seguimiento de resultados y la identificación de oportunidades de negocio. La falta de datos precisos puede generar decisiones basadas en información incompleta o desactualizada, afectando la competitividad de la empresa.
4. **Ausencia de control y reportes precisos:** Actualmente no existe un mecanismo automatizado que permita calcular comisiones, generar reportes de gestión o supervisar el desempeño de las distintas áreas. Esta carencia dificulta la evaluación del desempeño del personal, la supervisión de las operaciones y la generación de indicadores clave que apoyen la mejora continua de la corredora.

Estos problemas generan retrasos, errores frecuentes y dificultades para ofrecer un servicio eficiente y confiable a los clientes, afectando la imagen y competitividad de la empresa en el mercado inmobiliario. Por esta razón, se evidencia la necesidad de implementar un software de escritorio que centralice y automatice la administración de propiedades, clientes, contratos, recursos humanos y finanzas. La adopción de esta herramienta permitirá mejorar la eficiencia operativa, garantizar la integridad y disponibilidad de la información, reducir los errores humanos y facilitar la toma de decisiones estratégicas basadas en datos confiables. Además, contribuirá a optimizar el tiempo del personal, mejorar la organización interna y fortalecer la calidad del servicio ofrecido a los clientes, asegurando un crecimiento sostenible de la corredora.

## Objetivos del Proyecto (general y específicos)

#### Objetivo general

El proyecto tuvo como objetivo mejorar la eficiencia operativa y administrativa de la corredora de propiedades “La Calera” mediante la implementación de un software de escritorio que permitió centralizar la información de clientes, propiedades, contratos y reportes de gestión. Gracias a esta solución, la empresa redujo los tiempos de trabajo manual, disminuyó los errores asociados a la duplicidad de datos y fortaleció la trazabilidad de la información, facilitando una toma de decisiones más rápida y basada en información confiable.

En términos estratégicos, el software contribuyó a incrementar la productividad, optimizar los recursos internos y profesionalizar los procesos de gestión, otorgando a la corredora una ventaja competitiva frente a empresas del rubro que aún utilizan métodos tradicionales.

#### Objetivos específicos

Para alcanzar este propósito general y garantizar resultados medibles para la empresa, se establecieron los siguientes objetivos específicos:

1. **Optimizar la administración de la información interna**, consolidando en una única plataforma digital los registros de clientes, propiedades, contratos y comisiones.
2. **Reducir la carga operativa del personal administrativo**, automatizando tareas repetitivas y procesos manuales relacionados con la gestión de ventas y arriendos.
3. **Aumentar la confiabilidad de los datos** mediante el diseño e implementación de un modelo de base de datos relacional en PostgreSQL, que garantizó integridad, seguridad y escalabilidad.
4. **Fortalecer la capacidad de análisis y control** a través de la generación automatizada de reportes e indicadores que apoyaron la toma de decisiones estratégicas.
5. **Mejorar la trazabilidad y transparencia de las operaciones**, facilitando el seguimiento de cada proceso desde el registro inicial hasta la finalización del contrato.
6. **Impulsar la transformación digital de la corredora**, estableciendo una base tecnológica que permita la evolución futura del sistema y la incorporación de nuevas funcionalidades.

## Competencias del Perfil de Egreso

En este apartado debe insertar todas las competencias técnicas asociadas a su perfil de egreso y que se encuentran directamente relacionada al desarrollo del proyecto.

Debe insertar un mínimo de tres competencias técnicas.

El desarrollo del proyecto APT permite aplicar diversas competencias técnicas asociadas al perfil de egreso de Ingeniería en Informática, directamente relacionadas con la creación de soluciones de software eficientes y la gestión de información dentro de una organización. Entre las principales competencias técnicas destacadas se encuentran:

1. **Análisis de procesos y propuesta de soluciones informáticas:** Capacidad para identificar problemas, levantar requerimientos y proponer soluciones tecnológicas que optimicen los procesos internos de la empresa.
2. **Desarrollo de software aplicando buenas prácticas:** Habilidad para diseñar, implementar y mantener aplicaciones de escritorio confiables, escalables y de fácil mantenimiento, siguiendo estándares de calidad en programación.
3. **Construcción y administración de modelos de datos en PostgreSQL:** Competencia para estructurar y gestionar bases de datos relacionales, asegurando integridad, seguridad y accesibilidad de la información.
4. **Programación de consultas y rutinas SQL:** Capacidad para desarrollar procedimientos, consultas y rutinas que faciliten el acceso, manipulación y análisis de datos.
5. **Pruebas de calidad en software:** Habilidad para verificar que los módulos del sistema cumplan con los requerimientos funcionales y no funcionales, asegurando la confiabilidad del software.
6. **Gestión de proyectos informáticos:** Competencia para planificar, organizar y controlar las fases del proyecto, asegurando la correcta ejecución y cumplimiento de objetivos.

## Asignación de roles

Cada integrante del grupo debe tener asignado un rol dentro del proyecto en donde se debe describir las funciones y tareas que ejecutarán.

**Jefe de Proyecto: Cristian Zamora** Es responsable de la planificación, organización y supervisión general del proyecto. Sus funciones principales incluyen:

* Coordinar las actividades del equipo y asignar tareas según las competencias de cada integrante.
* Establecer el cronograma de trabajo y asegurar el cumplimiento de plazos y objetivos.
* Supervisar la correcta aplicación de la metodología en cascada en cada fase del desarrollo.
* Revisar avances, gestionar riesgos y resolver problemas que puedan surgir durante la ejecución.
* Mantener comunicación con las partes interesadas para informar sobre el progreso y resultados del proyecto.

**Desarrollador: Joaquín Rocha**

Su función principal es el diseño y desarrollo del software, con énfasis en la implementación técnica. Entre sus tareas se encuentran:

* Colaborar en el diseño de la base de datos PostgreSQL y de la arquitectura del sistema.
* Programar los módulos funcionales del software.
* Implementar rutinas y consultas SQL para garantizar el acceso eficiente a la información.
* Participar en la integración de los distintos componentes del sistema.
* Apoyar en la fase de pruebas para verificar la calidad y funcionalidad del software.

**Desarrollador: Fabián Balboa**

Se encarga del desarrollo de módulos y apoyo en la documentación del sistema. Sus tareas principales son:

* Implementar funcionalidades relacionadas con reportes e interfaz de usuario.
* Colaborar en la codificación y pruebas unitarias para asegurar la estabilidad del sistema.
* Contribuir en la elaboración de manuales de usuario y guías técnicas para la corredora.
* Apoyar en la detección y corrección de errores durante la fase de pruebas.
* Participar en el despliegue final del software y en la capacitación a los usuarios.

## Metodología utilizada en el Proyecto.

Durante el desarrollo del proyecto se evaluaron distintas metodologías de trabajo comúnmente empleadas en la ingeniería de software, con el propósito de seleccionar aquella que se ajusta de mejor manera a las condiciones del equipo, los requerimientos del cliente y la naturaleza del sistema a desarrollar.

Entre las metodologías analizadas se consideraron las siguientes:

* **Scrum:** metodología ágil basada en iteraciones cortas y entregas parciales del producto, que requiere una comunicación constante con el cliente para redefinir prioridades y ajustar funcionalidades. Aunque Scrum favorece la flexibilidad y la adaptación al cambio, su implementación demanda reuniones frecuentes, roles definidos (Scrum Master, Product Owner, equipo de desarrollo) y una alta participación del cliente, condiciones que no se presentaban en este proyecto.
* **Kanban:** enfoque visual de gestión de tareas centrado en la mejora continua y la reducción de cuellos de botella en el flujo de trabajo. Si bien Kanban resulta eficiente para proyectos con requerimientos cambiantes o en evolución constante, su naturaleza flexible no se alineaba con la planificación estructurada que se había definido desde el inicio del proyecto.
* **Prototipado:** metodología que se basa en construir versiones iniciales del sistema (prototipos) para obtener retroalimentación temprana del cliente. Este enfoque es útil cuando los requerimientos no están completamente definidos o cuando se busca validar conceptos antes de la implementación definitiva. En este caso, los requerimientos fueron establecidos con claridad desde la primera reunión con el cliente, por lo que no fue necesario aplicar esta metodología.
* **Espiral:** modelo que combina elementos de desarrollo iterativo con análisis de riesgos continuos. Aunque es una metodología robusta, su nivel de complejidad y los recursos que demanda resultaban excesivos para el alcance y tiempo de ejecución del presente proyecto.

Tras este análisis comparativo, se determinó que la **metodología tradicional en cascada** era la más adecuada para el desarrollo del software. Esta metodología se basa en un proceso secuencial y estructurado, donde cada fase se completa antes de avanzar a la siguiente, asegurando un control riguroso de los entregables y facilitando la trazabilidad de cada etapa.

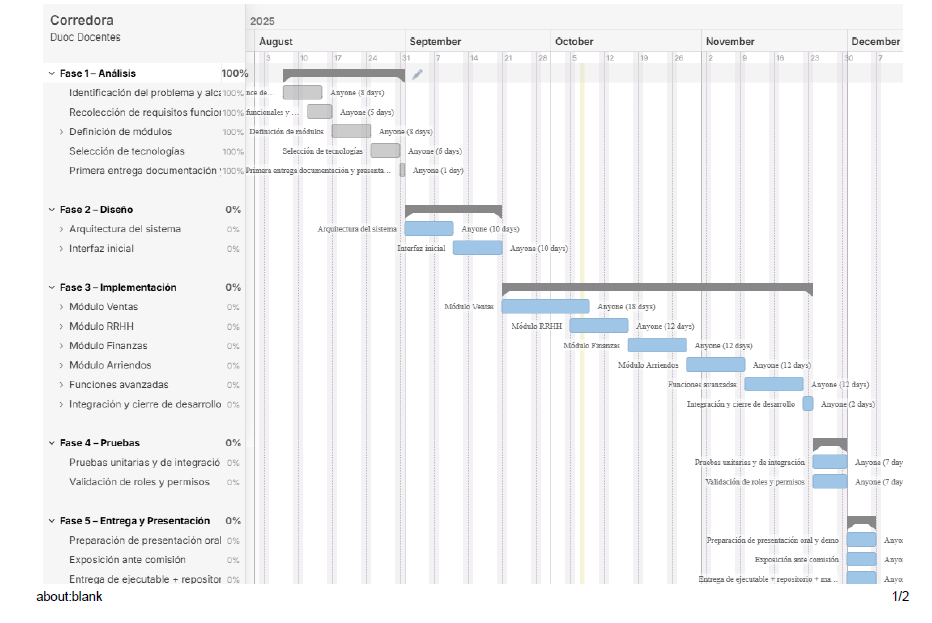
La elección del modelo en cascada se justificó principalmente porque los requerimientos fueron definidos de forma clara, completa y estable desde la primera reunión con el cliente, sin la necesidad de realizar iteraciones o ajustes frecuentes. Además, el contacto con el cliente durante la ejecución del proyecto fue limitado, lo que hacía menos viable la aplicación de metodologías ágiles que dependen de la retroalimentación continua.

El modelo en cascada permite mantener **una planificación ordenada, un control eficiente de los plazos y una documentación completa** en cada fase, lo que resultó fundamental para un equipo de desarrollo pequeño y con roles bien definidos. Las etapas aplicadas fueron las siguientes:

1. **Análisis:** recopilación y documentación de los requerimientos funcionales y no funcionales del sistema.
2. **Diseño:** elaboración del modelo de datos, la arquitectura del sistema y los prototipos de interfaz.
3. **Implementación:** desarrollo de los módulos del software y configuración de la base de datos.
4. **Pruebas:** validación del correcto funcionamiento del sistema mediante pruebas unitarias e integradas.
5. **Despliegue:** instalación y entrega del sistema en el entorno operativo del cliente, junto con su documentación técnica y de usuario.

## Creación de cronograma asociado al Proyecto (Carta Gantt)

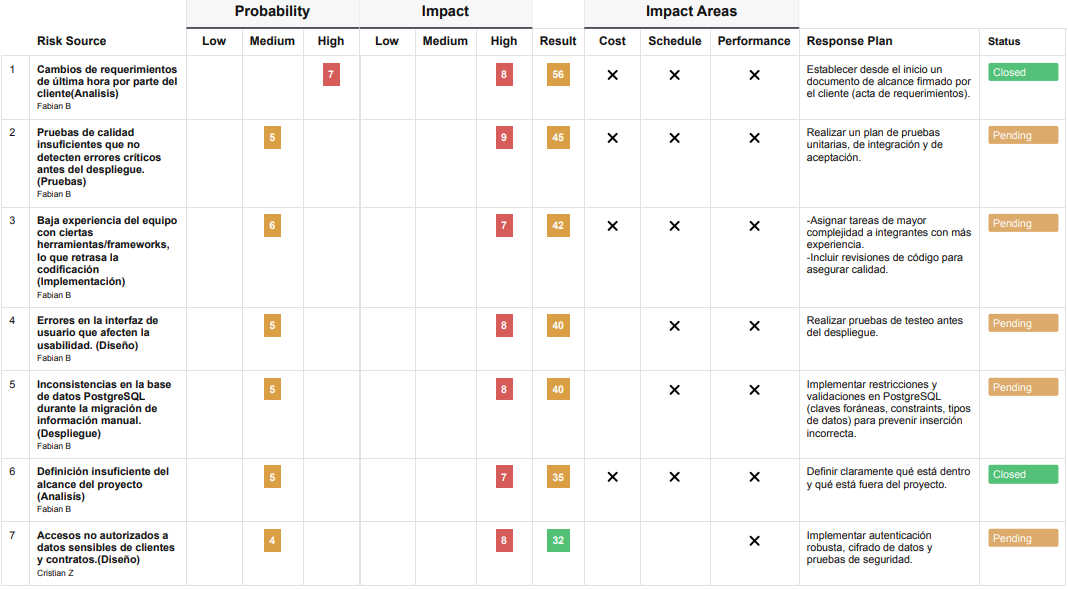
La siguiente Carta Gantt presenta la planificación temporal del proyecto, detallando las principales etapas, actividades y tareas involucradas en su desarrollo. En ella se especifican los recursos asignados a cada fase, así como la duración estimada y las fechas de inicio y término correspondientes, permitiendo visualizar de forma clara la secuencia y dependencia de las actividades.

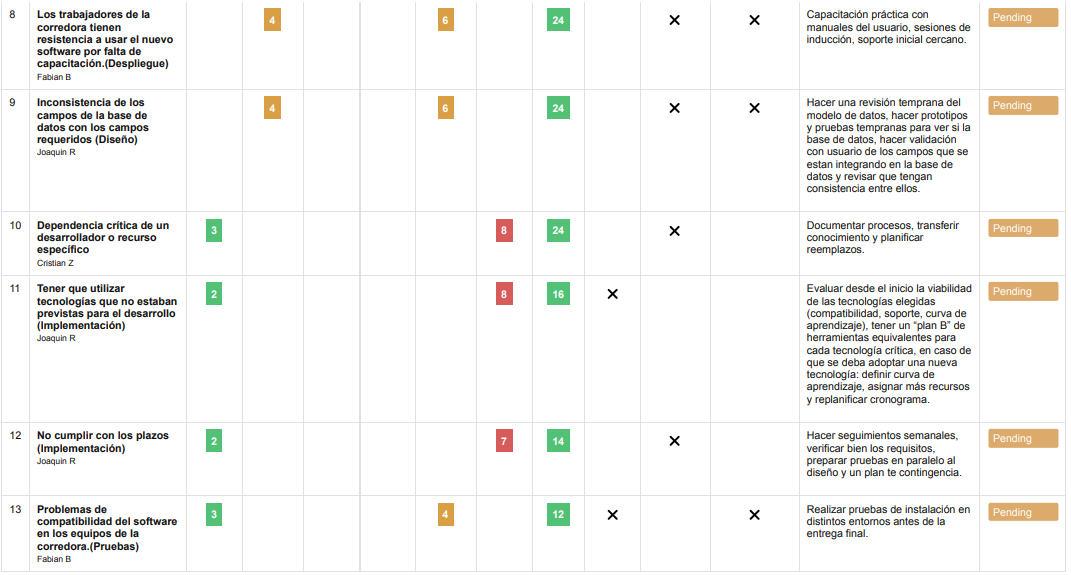




## Riesgos Asociados al Proyecto

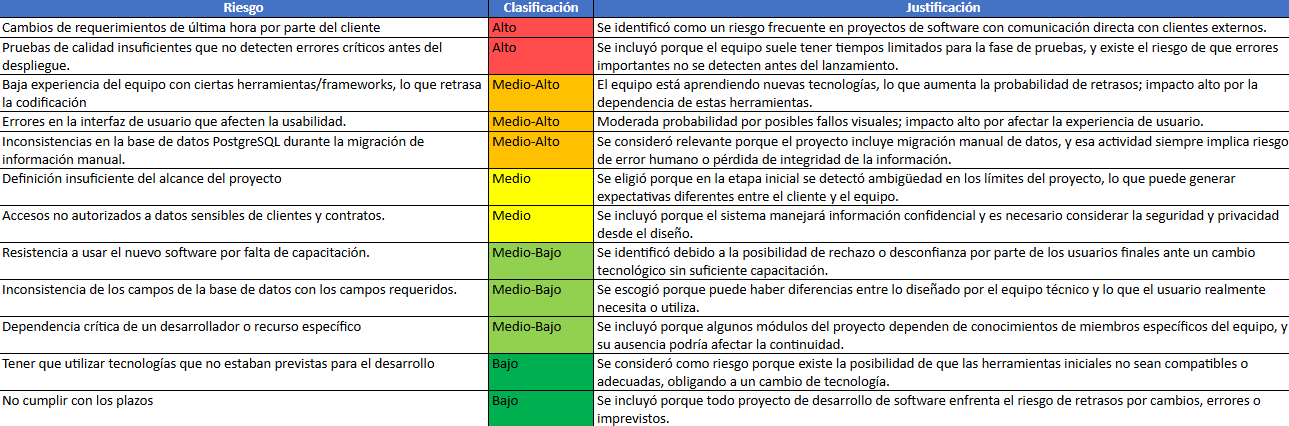
La siguiente matriz de riesgos nos permite identificar, evaluar y priorizar los posibles eventos que podrían afectar el proyecto. Cada riesgo se analiza según su probabilidad, impacto y criticidad, incluyendo además planes de mitigación y contingencia para minimizar problemas durante el desarrollo del sistema.





*Tabla 1*

La siguiente tabla presenta la justificación de la probabilidad asignada a cada uno de los riesgos identificados en el proyecto. Esta clasificación se basa en la frecuencia esperada de ocurrencia y en la experiencia del equipo respecto a proyectos de características similares, permitiendo establecer una base objetiva para la evaluación y priorización de los riesgos.



*Tabla 2*

## Implementación del Proyecto

### Diseño y Arquitectura de la solución (Caso de uso de Software o plataforma de gestión)

* **Tipo de sistema:** Aplicación de escritorio cliente-servidor.
* **Arquitectura:** Multicapa (capa de presentación, lógica de negocio y base de datos).
* **Plataforma de desarrollo:**

Lenguaje: Python con Pyside6 (C++)

Base de datos: PostgreSQL

Control de versiones: Git/GitHub

### Requerimientos técnicos

* Detallar cada uno de los dispositivos, insumos y herramientas que se utilizaran para la implementación del proyecto.

Dispositivos:

* Computadores portatiles personales

Insumos:

* Infraestructura

Herramientas:

* Visual Studio Code
* Supabase
* Pyside6
* Python(Backend)

### Desarrollo de la solución

* Se recomienda insertar capturas de pantalla con una breve explicación de cada uno de los pasos realizados en cada proceso de programación.
* Evidenciar la solución de software utilizando técnicas que permitan sistematizar el proceso de desarrollo y mantenimiento, asegurando el logro de los objetivos del proyecto.
* Integrar los distintos componentes de la solución de software utilizando técnicas que permitan sistematizar el proceso de desarrollo y mantenimiento, asegurando el logro de los objetivos del proyecto.

### Resultados de la solución

* En este apartado debe insertar todas las pruebas que fueron ejecutadas para chequear los resultados de la solución. Puede incorporar capturas de pantalla, videos o demostración en tiempo real de la solución.
* Los resultados deben ser mostrados en la exposición final frente a la comisión.

## Definición de Recursos y Costos asociados al Proyecto

# CONCLUSIÓN

Sintaxis final del documento que permite resumir y/o llegar a conclusiones desde la detección del problema o necesidad, pasando por el desarrollo del proyecto hasta las pruebas finales de funcionamiento.

# BIBLIOGRAFÍA

Para la bibliografía considerar Norma APA 7ma Edición.

# 

# ANEXOS

Debe incorporar como anexos en documento como archivo Word, todo tipo de información que complemente el proyecto.

En la entrega final del proyecto, deben enviar todos los archivos utilizados en la implementación y desarrollo del mismo. El docente mencionará unas clases antes de la entrega final dicho listado a entregar.