**FASE 2**

**DISEÑO E IMPLEMENTACIÓN DEL PROYECTO**

**SubwayOtakin**

Escuela de Informática y Telecomunicaciones

Octubre 2025

1. Identificación del Proyecto

|  |
| --- |
| **Nombre de grupo** |
| **SubwayOtakin** |

1. Integrantes del Equipo de Trabajo

|  |  |  |  |
| --- | --- | --- | --- |
| **N°** | **Rut** | **Apellidos** | **Nombres** |
| 1 | 21.542.763-3 | Fernández Valdés | Matías Ignacio |
| 2 | 21.381.143-6 | Torres Jaime | Kevin Alejandro |
| 3 | 20.549.865-6 | Isami Letelier | Rodrigo Alonso |

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). | MIFV |
| 2 | 15-09-25 | Todas | Cambios según primera revisión docente (alcances y objetivos). | RD |
| 3 | 22-9-25 | Todas | arreglar resumen, introducción | KATJ |

**INDICE DE CONTENIDO**

[LISTA DE TABLAS 4](#_heading=h.pb2i88lczqvx)

[LISTA DE FIGURAS 5](#_heading=h.z8kgdrhvn4q1)

[LISTA DE DIAGRAMAS 6](#_heading=h.pfrsvnou1man)

[GLOSARIO 8](#_heading=h.ri52wgs2oby7)

[1 Diseño e Implementación del Proyecto 10](#_heading=h.u1e3g1kjp8l6)

[1.1 El proyecto Subway Otakin se desarrolló como una propuesta innovadora 10](#_heading=h.s3p5ni43ys2h)

[1.2 Abstract 11](#_heading=)

[1.3 Introducción 12](#_heading=h.xffj95kdjcmy)

[1.4 Problemática a solucionar o necesidad a satisfacer 12](#_heading=h.uvnwoxx3spps)

[1.5 Objetivos del Proyecto (general y específicos) 14](#_heading=h.k004z9ahsy8i)

[1.6 Competencias del Perfil del Egreso 15](#_heading=h.j6qja9j5a76b)

[1.7 Asignación de roles 15](#_heading=)

[1.8 Metodología utilizada en el Proyecto. 16](#_heading=h.e1ubnc4d1inb)

[1.9 Creación de cronograma asociado al Proyecto (Carta Gantt) 17](#_heading=h.ec834u6cxufq)

[1.10 Riesgos Asociados al Proyecto 17](#_heading=h.xe3dmn6l97uj)

[1.11 Implementación del Proyecto 18](#_heading=h.j5vsydbjcu20)

[Diseño y Arquitectura de la solución (Caso de uso de Software o plataforma de gestión) 18](#_heading=h.54wepw5v260)

[Requerimientos técnicos 18](#_heading=h.2a64l1dqgx9)

[Desarrollo de la solución 18](#_heading=h.78xrzre4lkc1)

[Resultados de la solución 18](#_heading=h.w9ioh9clvwtg)

[1.12 Definición de Recursos y Costos asociados al Proyecto 18](#_heading=h.yc8jrtd5q08i)

[CONCLUSIÓN 19](#_heading=h.ian0odl1oi4t)

[BIBLIOGRAFÍA 20](#_heading=h.290wdsqaym39)

[ANEXOS 21](#_heading=h.bh4jb4ma19p7)

# LISTA DE TABLAS

[Caso de uso 1 6](#_Toc210687937)

[Vista Fisica 1 7](#_Toc210687990)

[Diagrama de notificaciones de Incidentes 1 8](#_Toc210688135)

[Diagrama 4+1 1 9](#_Toc210688177)

[Diagrama Despliege 1 7](#_Toc210686425)

[Tabla 1 Carta Gantt 18](#_Toc210668575)

[Riesgos Proyecto 1 20](#_Toc210687004)

# LISTA DE FIGURAS

* **Figura 1 -** Logotipo del Proyecto Subway Otakin
* **Figura 2 -** Interfaz de Inicio de la Aplicación Móvil
* **Figura 3 -** Pantalla de Notificaciones en Tiempo Real
* **Figura 4 -** Mapa Interactivo de Estaciones y Rutas
* **Figura 5 -** Dashboard de Supervisión y Reportabilidad
* **Figura 6 -**  Resultados de Usabilidad del MVP

# 

# GLOSARIO

* **API (Application Programming Interface):** Conjunto de funciones y procedimientos que permiten la interacción entre diferentes sistemas o aplicaciones.
* **Android Studio:** Entorno de desarrollo integrado (IDE) oficial para la creación de aplicaciones Android.
* **Arquitectura de Software:** Estructura general de un sistema que define sus componentes, relaciones y principios de diseño.
* **AWS (Amazon Web Services):** Plataforma de servicios en la nube que ofrece almacenamiento, bases de datos, cómputo y análisis de datos.
* **Backend:** Parte del desarrollo de software que se encarga de la lógica, procesamiento de datos y conexión con la base de datos.
* **Benchmarking:** Proceso de comparación de un producto o servicio con otros similares para identificar oportunidades de mejora.
* **Base de Datos:** Conjunto estructurado de información que puede ser consultada y gestionada mediante sistemas especializados como PostgreSQL.
* **Carta Gantt:** Herramienta gráfica de planificación que permite visualizar las actividades de un proyecto, sus duraciones y dependencias.
* **Dashboard:** Panel de control interactivo que muestra información relevante en tiempo real mediante indicadores y gráficos.
* **Demo Day Metro 2025:** Convocatoria o desafío de innovación impulsado por Metro de Santiago para fomentar soluciones tecnológicas aplicadas al transporte público.
* **Desarrollo de Software:** Proceso de diseño, codificación, prueba y mantenimiento de sistemas informáticos.
* **Frontend:** Parte del desarrollo que define la interfaz visual y la experiencia del usuario en una aplicación.
* **Gestión de Proyectos:** Conjunto de prácticas para planificar, ejecutar y controlar las actividades necesarias para cumplir con los objetivos de un proyecto.
* **Ingeniería en Informática:** Disciplina que aplica principios de la ingeniería y la computación para desarrollar soluciones tecnológicas.
* **Metodología Ágil SCRUM:** Marco de trabajo basado en iteraciones cortas llamadas sprints, donde se realizan entregas incrementales y se fomenta la colaboración continua.
* **MVP (Minimum Viable Product):** Versión inicial de un producto que incluye las funcionalidades mínimas necesarias para validar su utilidad con los usuarios.
* **Notificaciones Push:** Mensajes enviados desde un servidor hacia dispositivos móviles o navegadores, incluso cuando la aplicación no está abierta.
* **Perfil de Usuario:** Conjunto de características o preferencias que definen a un tipo de usuario dentro de un sistema.
* **PostgreSQL:** Sistema de gestión de bases de datos relacional de código abierto.
* **Prototipo:** Modelo preliminar de un sistema o producto que permite evaluar su funcionamiento y recibir retroalimentación antes del desarrollo final.
* **Reportabilidad:** Capacidad de un sistema para generar informes automáticos con datos útiles para la gestión o toma de decisiones.
* **Seguridad Informática:** Conjunto de medidas y prácticas destinadas a proteger la información y los sistemas frente a accesos no autorizados o fallas.
* **Sistema Escalable:** Sistema que puede crecer en capacidad o funcionalidad sin perder rendimiento ni estabilidad.
* **Sprint:** Periodo corto de trabajo dentro de la metodología SCRUM en el que se desarrolla un conjunto específico de tareas.
* **UI/UX (User Interface / User Experience):** Áreas del diseño de software enfocadas en la apariencia visual (UI) y la facilidad de uso y satisfacción del usuario (UX).
* **Visual Studio Code:** Editor de código fuente ligero y multiplataforma utilizado para el desarrollo de aplicaciones.

# Diseño e Implementación del Proyecto

## Resumen

El proyecto **Subway Otakin** se desarrolló como una propuesta innovadora para enfrentar una de las principales problemáticas del Metro de Santiago: la falta de soluciones digitales que entregaran a los pasajeros información confiable, personalizada y en tiempo real. Aunque existían aplicaciones relacionadas con el transporte público, estas no lograban responder a las necesidades específicas de los usuarios, lo que generaba deficiencias en la experiencia de viaje y en la gestión operativa.

Los pasajeros del Metro debían lidiar con largas esperas sin claridad en los tiempos, notificaciones imprecisas o inexistentes sobre contingencias, y herramientas que no consideraban perfiles diferenciados como turistas, estudiantes, adultos mayores o usuarios frecuentes. Esta situación afectaba la satisfacción de los pasajeros y disminuía la confianza en el sistema.

Frente a este escenario, **Subway Otakin se implementó como una solución tecnológica** que transformó la interacción de los pasajeros con el Metro de Santiago, entregando información en tiempo real, experiencias personalizadas y herramientas de gestión para la administración del servicio. El proyecto respondió a los desafíos definidos en la convocatoria *Demo Day Metro 2025*, en la línea “La nueva era de la app, personalización y conexión para el pasajero”.

La propuesta incluyó el desarrollo de una aplicación móvil y web con distintos módulos:

* **Gestión de perfiles de usuario:** permitió diferenciar entre estudiantes, turistas, adultos mayores y usuarios frecuentes, adaptando la experiencia de la aplicación.
* **Notificaciones en tiempo real:** el sistema envía alertas sobre retrasos, incidentes y contingencias operativas a través de notificaciones push y datos confiables.
* **Mapas interactivos y accesibilidad:** los usuarios visualizan recorridos, estaciones cercanas y tiempos estimados, además de contar con funciones inclusivas como lectura de texto, contraste ajustable y selección de idioma.
* **Dashboards de supervisión:** los administradores accedieron a reportes y análisis en tiempo real para mejorar la toma de decisiones.

El desarrollo se organizó bajo la metodología **Ágil SCRUM**, en sprints semanales con entregas incrementales. Durante el semestre se construyó un **prototipo mínimo viable (MVP)** que permitió validar tempranamente las funcionalidades críticas.

La factibilidad técnica estuvo respaldada por la experiencia del equipo en desarrollo de software y análisis de datos, además del uso de herramientas como Visual Studio Code, Android Studio, PostgreSQL y AWS. El tiempo disponible fue suficiente para implementar un prototipo funcional que demostró la escalabilidad de la solución.

El impacto del proyecto fue doble:

* **Para los usuarios:** mejoró la experiencia de viaje en el Metro de Santiago, reduciendo la incertidumbre y aumentando la confianza en el servicio.
* **Para los administradores:** entregó herramientas de monitoreo y análisis que optimizaron la operación y redujeron los tiempos de respuesta frente a contingencias.

Finalmente, el proyecto constituyó un aporte real a la innovación en el Metro de Santiago, además de ser un desafío académico en la carrera de Ingeniería en Informática. La solución demostró un alto potencial de escalabilidad y abrió la posibilidad de ser replicada en otras ciudades del país, impulsando una movilidad más eficiente, inclusiva y digital.

## Abstract

The Subway Otakin project emerges as an innovative proposal to address one of the main challenges currently affecting urban mobility in Santiago, Chile: the lack of comprehensive digital solutions that provide passengers with reliable, personalized, and real-time information. Although there are existing public transport applications, most of them fail to adapt to the specific needs of different users, resulting in inefficiencies in travel experience and operational management.

Subway Otakin aims to transform the way passengers interact with Santiago’s transportation system by offering real-time information, personalized experiences, and management tools for operating entities. The project is aligned with the challenges defined in the Metro Demo Day 2025 initiative, particularly in the line “The new era of the app, personalization and connection for the passenger,” ensuring strategic alignment with Metro’s innovation goals.

The proposed solution consists of a mobile and web application that integrates several modules:

* User profile management (students, tourists, senior citizens, frequent users).
* Real-time notifications for delays, incidents, and route changes using push alerts.
* Interactive maps and accessibility features, including text-to-speech, high contrast, and multilingual support (Spanish/English).
* Automated reporting and dashboards for administrators, enabling real-time monitoring, data-driven decision-making, and improved operational efficiency.

The project will be developed using the Agile SCRUM methodology, organized in weekly sprints with incremental deliverables. This approach ensures flexibility, early validation of critical features, and adaptability to changes. A minimum viable product (MVP) will be delivered within the semester, demonstrating the technical feasibility and scalability of the system.

The expected impact of Subway Otakin is twofold:

* For passengers: enhancing their travel experience in Santiago’s Metro by reducing uncertainty and improving trust through personalized and reliable information.
* For administrators: providing monitoring and reporting tools that reduce response times to contingencies and optimize overall operational processes.

This project not only contributes academically, by applying competencies in software development, data modeling, quality assurance, and project management, but also offers a real-world solution that supports sustainable and inclusive mobility. With strong potential for scalability, Subway Otakin could be replicated in other cities across Chile, becoming a key digital platform to drive innovation in public transportation.

.

## Introducción

**Introducción**

Tal como se describió anteriormente, el transporte público constituye uno de los pilares fundamentales en el desarrollo urbano y en la calidad de vida de las personas. En el caso de Santiago, el Metro fue el eje central de la movilidad, movilizando a millones de usuarios diariamente y desempeñando un rol estratégico en la conexión de la ciudad. La magnitud de este sistema lo convirtió en un espacio clave para la innovación tecnológica y la búsqueda de soluciones orientadas a mejorar la experiencia de viaje de los pasajeros y la eficiencia en la gestión de los recursos.

Los avances en digitalización y conectividad abrieron nuevas oportunidades para transformar la forma en que los pasajeros interactúan con el Metro. Sin embargo, persistían desafíos relacionados con la entrega de información en tiempo real, la personalización de la experiencia y la accesibilidad para distintos grupos de usuarios. Estas limitaciones reducían la confianza de los pasajeros, generaban incertidumbre durante los desplazamientos y restringen la capacidad de los administradores para tomar decisiones basadas en datos.

En este contexto, se desarrolló el proyecto **Subway Otakin**, una aplicación digital móvil y web diseñada para el Metro de Santiago. Su propósito fue ofrecer un sistema integral que entregará a los pasajeros información personalizada y en tiempo real, y que a la vez brindará a los administradores herramientas de reportabilidad y visualización para fortalecer la operación. La solución respondió a los desafíos planteados en la convocatoria *Demo Day Metro 2025*, alineándose con las tendencias de innovación en transporte urbano y con las necesidades reales de los usuarios.

Este proyecto no solo representó una respuesta tecnológica a una problemática concreta, sino que también permitió el desarrollo de competencias profesionales en Ingeniería en Informática, integrando áreas como desarrollo de software, modelado de datos, aseguramiento de calidad y gestión de proyectos. Además, se proyectó como una solución escalable que podría aplicarse en otros sistemas de transporte urbano del país, aportando al objetivo de avanzar hacia una movilidad más sostenible, inclusiva y digital.

## Problemática a solucionar o necesidad a satisfacer

La ciudad de Santiago concentra uno de los sistemas de transporte público más extensos de Chile, siendo el Metro el principal eje articulador de la movilidad urbana, con millones de pasajeros transportados diariamente. Sin embargo, pese a los avances en infraestructura y cobertura, existe una brecha digital significativa en lo relativo a las herramientas tecnológicas que acompañan la experiencia de viaje de los usuarios.

Actualmente, los pasajeros cuentan con aplicaciones de transporte que entregan información básica de recorridos y horarios, pero estas presentan limitaciones importantes:

* Ausencia de notificaciones personalizadas en tiempo real que consideren la ubicación del pasajero, sus hábitos de viaje, contingencias operativas y sus preferencias individuales.
* Experiencia homogénea que no distingue entre distintos perfiles de usuario, como turistas, estudiantes, adultos mayores o trabajadores frecuentes, lo que disminuye la accesibilidad y el valor percibido del servicio.
* Limitada integración de datos de operación interna, lo que dificulta la generación de reportes automáticos, la trazabilidad de eventos y la toma de decisiones estratégicas en tiempo oportuno.
* Baja capacidad de prevención de incidentes y comunicación inmediata frente a situaciones de riesgo en estaciones o líneas, lo que puede impactar en la seguridad de los pasajeros.

Estas carencias repercuten directamente en la experiencia de los usuarios del Metro de Santiago, quienes enfrentan incertidumbre en sus desplazamientos, mayores tiempos de espera y una menor percepción de calidad en el servicio. Al mismo tiempo, los administradores del sistema cuentan con información dispersa y poco integrada, lo que genera retrasos en la gestión y en la detección temprana de problemas.

La problemática se hace aún más relevante considerando que el transporte público es el pilar de la movilidad sostenible en las grandes ciudades, y que la innovación digital es una herramienta clave para aumentar la confianza, mejorar la satisfacción de los pasajeros y optimizar los recursos operativos.

En este contexto, surge la necesidad de contar con una plataforma digital integral que conecte a los pasajeros con el sistema de transporte en tiempo real, adaptándose a sus perfiles y necesidades específicas, al mismo tiempo que provea a los administradores con herramientas de monitoreo y análisis que fortalezcan la toma de decisiones.

La propuesta Subway Otakin se plantea como respuesta directa a esta problemática, al integrar notificaciones personalizadas, segmentación de usuarios, mapas interactivos, funcionalidades de accesibilidad y dashboards de gestión, alineándose con los desafíos planteados en la convocatoria Demo Day Metro 2025 y proyectándose como una solución escalable y de impacto social.

## Objetivos del Proyecto (general y específicos)

**Objetivo General**

Desarrollar una aplicación digital móvil y web denominada *Subway Otakin*, orientada a mejorar la experiencia de los usuarios del Metro de Santiago mediante la entrega de información personalizada y en tiempo real, la integración de funcionalidades de accesibilidad y la incorporación de herramientas de reportabilidad y monitoreo para la gestión operativa.  
Esta solución busca optimizar la supervisión de las operaciones, reducir los tiempos de respuesta frente a contingencias, fortalecer la comunicación con los usuarios y aumentar la satisfacción de los pasajeros, promoviendo la confianza y fidelización hacia el servicio.

❖ Analizar y documentar los requerimientos funcionales y no funcionales del sistema, considerando las necesidades de los pasajeros y de los administradores internos.

❖ Diseñar la arquitectura de software y el modelo de datos que soporten de manera escalable las funcionalidades del sistema, garantizando seguridad, accesibilidad y adaptabilidad futura.

❖ Implementar el módulo de gestión de usuarios y perfiles personalizados, permitiendo la segmentación en distintos tipos de pasajeros (frecuentes, estudiantes, adultos mayores, turistas).

❖ Desarrollar e integrar el sistema de notificaciones personalizadas en tiempo real para informar a los usuarios sobre horarios, contingencias, retrasos y alertas de seguridad.

❖ Incorporar mapas interactivos y funciones de accesibilidad que aseguren la inclusión de usuarios con diferentes necesidades (lectura de texto, alto contraste, tamaño de letra ajustable e idiomas español e inglés).

❖ Construir dashboards de supervisión y reportabilidad automática que permitan a los administradores generar informes y visualizar indicadores de uso y operación en tiempo real.

❖ Aplicar pruebas de calidad de software (unitarias, de integración y de usabilidad) para asegurar que el sistema cumpla con los requerimientos establecidos y funcione de manera confiable en distintos entornos.

❖ Elaborar documentación técnica y funcional que registre el análisis, diseño, implementación, pruebas y resultados obtenidos, garantizando la trazabilidad del proyecto.

❖ Validar el prototipo mínimo viable (MVP) en función de los objetivos definidos y evaluar su potencial de escalabilidad dentro del Metro de Santiago.

## Competencias del Perfil del Egreso

El desarrollo del proyecto *Subway Otakin* permitió la aplicación integral de diversas competencias asociadas al perfil de egreso de la carrera, fortaleciendo tanto las habilidades técnicas como las de gestión.

Entre las principales competencias ejercitadas se destacan:

* **Desarrollo de software con buenas prácticas:** aplicación de metodologías de programación estructurada y orientada a objetos, empleando estándares de calidad, documentación adecuada y control de versiones para garantizar la eficiencia y mantenibilidad del código.
* **Modelamiento y manipulación de datos:** diseño y gestión de bases de datos relacionales mediante el uso de herramientas y lenguajes especializados, asegurando la integridad, consistencia y disponibilidad de la información.
* **Gestión de proyectos informáticos y aseguramiento de calidad:** planificación, coordinación y control de actividades técnicas utilizando metodologías ágiles, junto con la implementación de estrategias de prueba y verificación para garantizar la confiabilidad del producto final.

En conjunto, estas competencias reflejan el **sello profesional del egresado**, orientado a la resolución eficiente de problemáticas tecnológicas mediante el uso de herramientas modernas, el trabajo colaborativo y el compromiso con la calidad del software desarrollado.

*.*

## Asignación de roles

* **Matías Fernández – Jefe de Proyecto / Arquitecto de Software**

Matías asume el liderazgo general del proyecto, supervisando el cumplimiento de los plazos, la correcta coordinación del equipo y la toma de decisiones estratégicas. Es responsable de elaborar el plan de trabajo, distribuir tareas de acuerdo con las habilidades de cada integrante y gestionar la comunicación con el profesor o entidad solicitante.

Como Arquitecto de Software, Matías define la estructura técnica del sistema, selecciona las tecnologías a utilizar y establece los lineamientos para garantizar la escalabilidad, mantenibilidad y seguridad de la solución propuesta. Además, valida que la integración entre los módulos Frontend y Backend se realice de manera correcta y eficiente.

* **Kevin Torres:** Desarrollo Frontend, Experiencia de Usuario (UI/UX).

Kevin tiene a su cargo el diseño e implementación de la interfaz gráfica con la que interactuarán los usuarios finales. Se encarga de construir el Frontend utilizando tecnologías modernas, asegurando que el sistema sea visualmente atractivo, intuitivo y accesible.

Dentro de su rol de UI/UX, realiza maquetas, prototipos y pruebas de usabilidad para validar que la experiencia del usuario sea fluida y satisfactoria. Trabaja en conjunto con el equipo para recibir retroalimentación y aplicar mejoras constantes en el diseño

* **Rodrigo Letelier:** Desarrollo Backend, Modelado de Datos.

Rodrigo es el encargado de desarrollar la lógica interna del sistema, programando los servicios y funcionalidades que operan del lado del servidor. Define y construye la estructura de la base de datos, asegurando la integridad y consistencia de la información.

También implementa APIs y mecanismos de comunicación entre el Frontend y el Backend, garantizando que los datos se procesen de manera eficiente y segura. Además, colabora en pruebas técnicas para detectar fallos y optimizar el rendimiento del sistema.

## Metodología utilizada en el Proyecto.

Para la ejecución del proyecto *Subway Otakin* se adoptó la metodología ágil **Scrum**, dado que el desarrollo de una solución tecnológica orientada a distintos tipos de usuarios —como estudiantes, turistas y adultos mayores— requiere flexibilidad, iteración constante y capacidad de adaptación a nuevos requerimientos conforme avanza el proceso. Scrum permitió organizar el trabajo de manera colaborativa y progresiva, facilitando la validación temprana del prototipo mínimo viable (MVP) y el ajuste oportuno de funcionalidades según las necesidades detectadas.

El trabajo se estructuró en **sprints semanales**, cada uno acompañado de reuniones de planificación (*Sprint Planning*) donde se definían los objetivos y tareas específicas de cada miembro del equipo. Durante la semana se realizaron instancias de seguimiento para evaluar el progreso y resolver posibles obstáculos, mientras que al cierre de cada sprint se llevó a cabo una **revisión del avance (Sprint Review)** y una **retrospectiva (Sprint Retrospective)** para identificar mejoras en el proceso.

El cronograma de desarrollo se organizó de la siguiente manera:

* **Sprint 1:** Levantamiento de requerimientos y benchmarking de soluciones existentes en el sector del transporte público.
* **Sprint 2:** Diseño de la arquitectura del sistema y modelado de datos.
* **Sprints 3 al 5:** Desarrollo incremental de los módulos principales: interfaz de usuario (Frontend), servicios y lógica interna (Backend) e implementación del sistema de notificaciones en tiempo real.
* **Sprint 6:** Pruebas de integración, detección y corrección de errores.
* **Sprint 7:** Documentación final y preparación de la entrega del MVP.

La elección de Scrum se consideró más adecuada que metodologías tradicionales como el **Modelo en Cascada**, el cual se basa en un desarrollo lineal donde cada fase debe completarse totalmente antes de avanzar a la siguiente. En proyectos como *Subway Otakin*, donde los requerimientos evolucionan según la retroalimentación de usuarios y las necesidades operativas del Metro de Santiago, un enfoque rígido como Cascada limitaría la capacidad de reacción ante cambios.

En cambio, **Scrum permitió realizar entregas parciales, probar funcionalidades rápidamente y ajustar el rumbo cuando fue necesario**, lo que garantizó un desarrollo más eficiente, colaborativo y alineado con los objetivos planteados.

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

Debe insertar la carta Gantt del proyecto donde se debe evidenciar las principales etapas, recursos asociados a cada tarea y visualizar las fechas de cada una de ellas.

Tabla 1 Carta Gantt

Imagen que contiene Aplicación

El contenido generado por IA puede ser incorrecto.

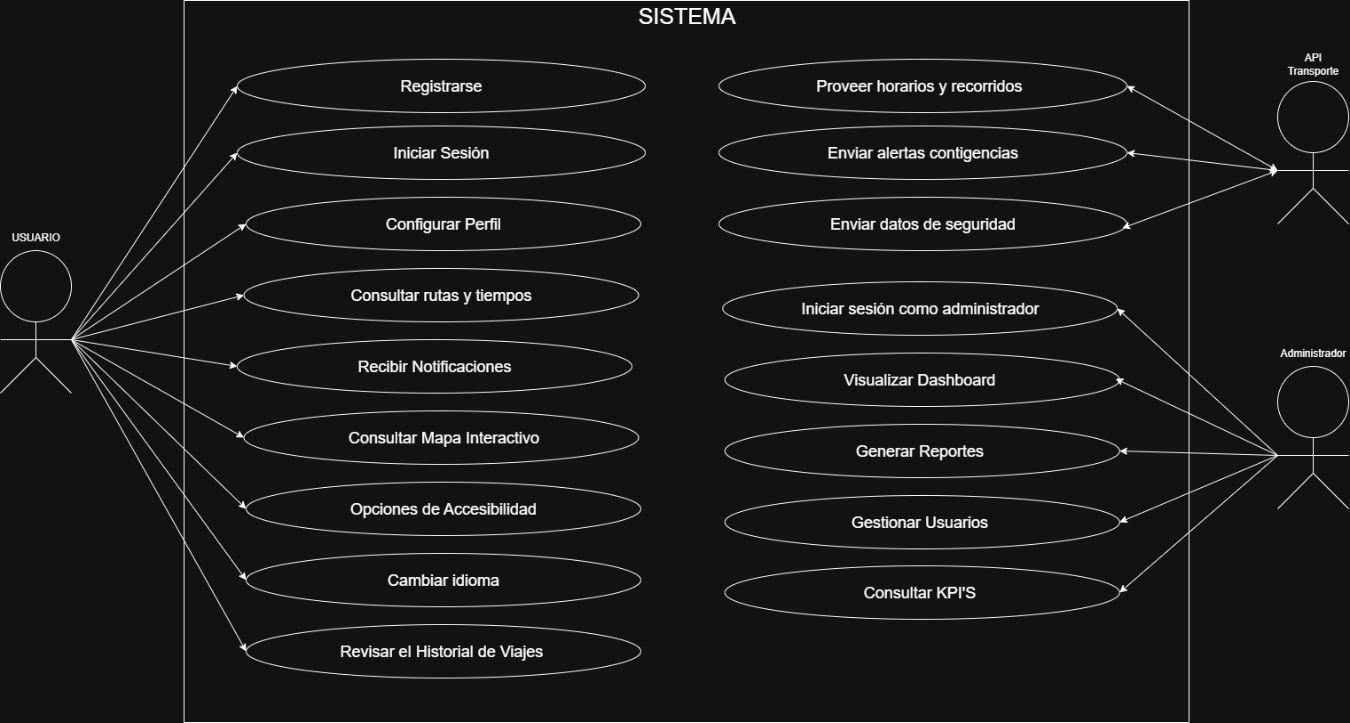
## Riesgos Asociados al Proyecto

# LISTA DE DIAGRAMAS

* **Diagrama 1 -** Diagrama de Casos de Uso

Caso de uso 1

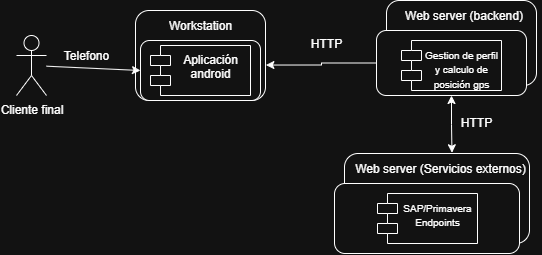
* + Este diagrama representa las interacciones entre los actores del sistema (usuarios o sistemas externos) y las funcionalidades principales que ofrece. Corresponde a la **vista lógica**, ya que se centra en mostrar el comportamiento del sistema desde la perspectiva del usuario, identificando los casos de uso sin entrar en detalles técnicos o de infraestructura.



* **Diagrama 2 -** Diagrama de Vista Física

Vista física 1

* + El diagrama de vista física muestra la distribución e interconexión de los componentes físicos del sistema, como servidores, dispositivos y redes. Este diagrama pertenece a la **vista de desarrollo o infraestructura**, ya que describe cómo se implementan físicamente los componentes del sistema y cómo se comunican entre ellos para garantizar el funcionamiento del software en un entorno rea

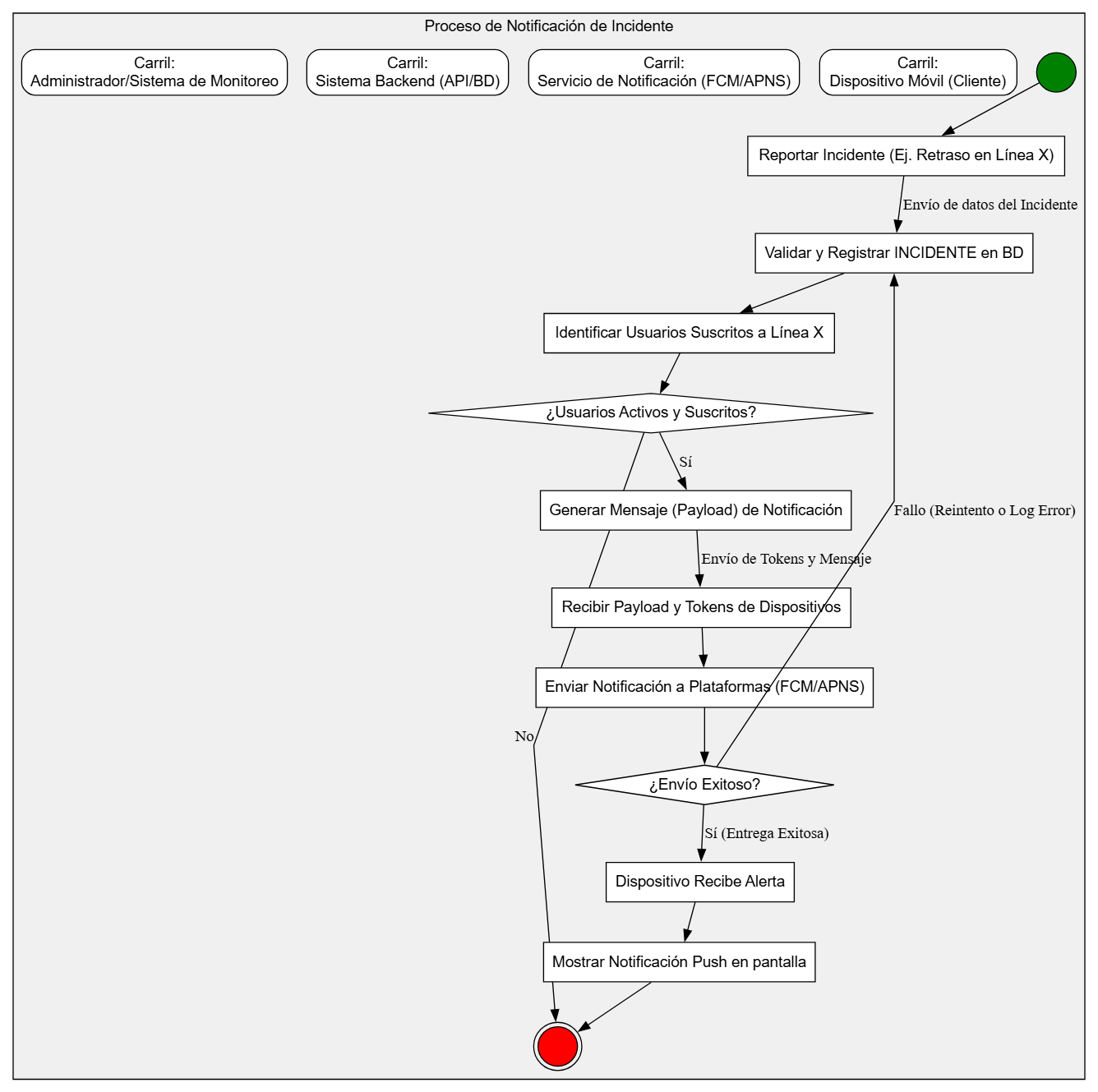


* **Diagrama 3 –** Diagrama de notificaciones de Incidentes

Diagrama de notificaciones de Incidentes 1

* + Este diagrama describe el flujo de comunicación y procesamiento de las notificaciones dentro del sistema cuando ocurre un incidente. Puede interpretarse como un **diagrama de flujo de procesos**, ya que muestra cómo se manejan los eventos, las condiciones y las respuestas dentro del sistema. Por su enfoque en la interacción y ejecución de los procesos en tiempo real, corresponde a la **vista de proceso**, que representa cómo los

distintos componentes del sistema colaboran durante su ejecución



* **Diagrama 4 -** Diagrama 4+1

Diagrama 4+1 1

* + El diagrama del modelo 4+1 combina las distintas vistas arquitectónicas del sistema (lógica, de desarrollo, de proceso y física) junto con la vista de escenarios. Este diagrama ofrece una **visión global de la arquitectura del sistema**, integrando los diferentes enfoques para proporcionar una representación completa y coherente del proyecto. Por tanto, sirve como marco integrador que conecta todas las vistas anteriores.

Diagrama

El contenido generado por IA puede ser incorrecto.

* **Diagrama 5 -** Diagrama de Despliegue (Infraestructura Técnica)
  + El diagrama de despliegue muestra cómo se distribuyen los componentes del sistema en la infraestructura técnica, como servidores, redes y otros recursos de hardware. Este diagrama encaja directamente en la vista física, que se centra en cómo el sistema se despliega y se distribuye a nivel de infraestructura, incluyendo la conectividad y la distribución de componentes a través de la red, los dispositivos y el hardware.

Diagrama Despliege 1

Interfaz de usuario gráfica, Aplicación, Word

El contenido generado por IA puede ser incorrecto.

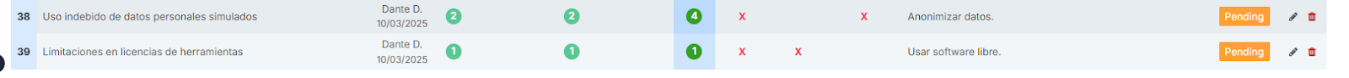
Riesgos Proyecto 1

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.



## Implementación del Proyecto

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

* Describir en profundidad el modelo de datos que permiten soportar los requerimientos de la organización de acuerdo al diseño definido que sea escalable en el tiempo.

### Requerimientos técnicos

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

### 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.