}

# PARTE I

| 1. Antecedentes Personales |
| --- |

| **Nombre estudiante** | **Diego Encina Poblete**  **Ignacio Andrés Cabañas Zúñiga**  **Felipe Ignacio Tapia Diaz** |
| --- | --- |
| **Rut** | **21.092.939-8**  **21.510.487-7**  **21.126.460-8** |
| **Carrera** | **INGENIERIA EN INFORMATICA** |
| **Sede** | **Sede Duoc UC San Bernardo** |

| 2. Descripción Proyecto APT |
| --- |

| **Nombre del proyecto** | **CI-MUN** |
| --- | --- |
| ***Área (s) de desempeño(s)*** | Desarrollo de software, inteligencia artificial aplicada, ingeniería web, experiencia de usuario |
| ***Competencias*** | * Desarrollar aplicaciones informáticas con integración de IA. * Implementar soluciones tecnológicas seguras, escalables y centradas en el usuario. * Diseñar e integrar sistemas de comunicación eficientes. * Aplicar metodologías ágiles en el desarrollo de software innovador. * Diseñar y optimizar aplicaciones web seguras, escalables y responsivas. |

| 3. Fundamentación Proyecto APT |
| --- |

| **Relevancia del proyecto APT** | La plataforma CI-MUN, utilizada por distintos municipios en Chile, enfrenta desafíos como:   * Procesamiento manual de incidencias con tiempos de respuesta extensos. * Un sistema de mensajería limitado en personalización y canales de contacto. * Una plataforma web que requiere mejoras en **usabilidad, accesibilidad y rendimiento**, especialmente para ciudadanos que acceden desde dispositivos móviles.   El proyecto propone incorporar IA, mejorar la mensajería e implementar un rediseño de la plataforma web, respondiendo así a la necesidad de modernizar la gestión municipal y fortalecer la relación entre municipio y ciudadanía. |
| --- | --- |
| **Descripción del Proyecto APT** | *Dentro del proyecto se trabajará*   * ***Módulo de IA****:* * ***Mensajería avanzada****:* * ***Optimización de la plataforma web*** |
| **Pertinencia del proyecto con el perfil de egreso** | Este proyecto requiere competencias claves de un informático:   * Diseño y desarrollo de software junto con una plataforma web moderna. * Integración de IA con aplicaciones en producción. * Mejora de la experiencia de usuario y comunicación digital. * Aplicación de buenas prácticas de seguridad y rendimiento en desarrollo web. |
| **Relación con los intereses profesionales** | El proyecto se alinea con el interés en transformación digital aplicada al sector público, aportando experiencia práctica en:   * Inteligencia Artificial aplicada a servicios reales. * Desarrollo web con impacto social. * Comunicación digital avanzada para mejorar la interacción con usuarios. |
| **Factibilidad de desarrollo del Proyecto APT** | [**Link estudio de Factibilidad – Plataforma CI-MUN**](https://docs.google.com/document/d/12ztlqyYgQpIe1mR3i2eadB05JtZRaRQp5TRl-dBgS7Y/edit?usp=sharing) |

# PARTE II

| 4. Objetivos |
| --- |

| **Objetivo general** | Diseñar e implementar un sistema mejorado para CI-MUN incorporando Inteligencia Artificial, mensajería avanzada y optimización la plataforma web, con foco en eficiencia, usabilidad y comunicación ciudadana. |
| --- | --- |
| **Objetivos específicos** | **Objetivos Específicos:**   * Desarrollar un módulo de Inteligencia Artificial capaz de clasificar, priorizar y asignar automáticamente las incidencias ciudadanas, reduciendo el tiempo de procesamiento manual. * Integrar un sistema de mensajería multicanal **(WhatsApp y notificaciones push)** para enviar actualizaciones en tiempo real a los ciudadanos sobre el estado de sus incidencias. * Implementar de dashboards de gestión para visualizar métricas clave y evaluar el impacto de las nuevas funcionalidades en la gestión municipal. * Establecer un sistema de seguimiento y métricas para evaluar la eficiencia de la plataforma CI-MUN y el impacto de las nuevas funcionalidades en la gestión municipal. * Generar reportes de alto nivel para la toma de decisiones estratégicas, con la condición de consolidar la información relevante del proyecto. * Definir un conjunto de indicadores de rendimiento (KPIs) y reglas de negocio, con la condición de estandarizar el manejo de las incidencias desde el inicio hasta el fin. * Diseñar un dashboard de scoring para mostrar puntuaciones de capacidad de cada área y el estado de las tareas, con la condición de reflejar el rendimiento operativo de manera clara.   **Funcionalidades del Sistema:**   * Automatizar el envío de notificaciones directas a través de Google Firebase Cloud Messaging, con la condición de estandarizar la comunicación a tiempo real. * Configurar el sistema para enviar notificaciones directas a través de la API de WhatsApp Business, con la condición de ampliar los canales de comunicación con los ciudadanos. * Crear un sistema de alertas en tiempo real, con la condición de notificar automáticamente cuando los KPIs o los resultados del scoring lo requieran. |

| 5. Metodología |
| --- |

**Metodología Scrum CI-MUN (Gestión Documental)**

Metodología Scrum encargada de gestionar el proyecto de la mejor manera la plataforma CI-MUN. El cronograma se ajusta a 6 Sprints de tres semanas cada uno, completando las 18 semanas de práctica profesional.

### **A.Contexto del Proyecto y Asignación de Roles**

Proyecto enfocado en optimizar los procesos internos de gestión documental municipal, integrando IA para la clasificación, resumen, y el despacho con notificaciones multicanal.

| **Rol** | **Integrante** | **Horas Asignadas (Brutas)** | **Enfoque Principal** |
| --- | --- | --- | --- |
| Dev 1 – Backend | Felipe | 360 hh | Modelos, Lógica Documental, Integración IA, Seguridad. |
| Dev 2 – Frontend | Diego | 360 hh | Interfaces, Visualizadores (Dashboards), UI/UX Responsive, Notificaciones. |
| Dev 3 – Integraciones & QA | Ignacio | 360 hh | Firma Digital, APIs de Notificación (Push/Correo), Pruebas y Activación. |

### **B. Capacidad y Velocidad del Equipo**

Esta sección formaliza el tiempo total disponible y la capacidad de desarrollo del equipo.

* **Integrantes del equipo:** 3
* **Duración del Proyecto:** 18 semanas (6 Sprints 3 semanas/Sprint).
* **Horas Totales Brutas (Equipo):** 3×360=1080 horas.
* **Deducción por Gestión y Ceremonias (30%):** 324 horas.
* **Horas Netas de Desarrollo (70%):** 756 horas (Tiempo efectivo de codificación y pruebas).

**Velocidad de Sprint (Capacidad Neta)**

* **Duración de 1 Sprint:** 3 semanas (15 días laborables).
* **Capacidad por Sprint (Horas Totales Netas):** 756 horas netas/6 Sprints=126 horas netas por Sprint.
* **Objetivo de Velocidad (Bruta):** El equipo buscará completar el esfuerzo bruto por Sprint (60 hh/Dev).

### **C.Principios Fundamentales de Scrum**

* **Transparencia (Transparency):** El estado del proyecto, el trabajo pendiente (Backlog) y el proceso de desarrollo deben ser visibles y entendidos por todos los participantes.
* **Inspección (Inspection):** El equipo debe inspeccionar frecuentemente los artefactos y el progreso del Sprint (Daily Scrum, Revisión del Sprint).
* **Adaptación (Adaptation):** Si los resultados de la inspección indican que uno o más aspectos se desvían de límites aceptables, el proceso o el producto deben ajustarse (Retrospectiva del Sprint).

### **D. Artefactos de Scrum**

| **Artefacto** | **Descripción** | **Responsable** |
| --- | --- | --- |
| Product Backlog | Lista ordenada de todas las funcionalidades, requerimientos y mejoras del proyecto (total 756 horas netas). | Jefe práctica |
| Sprint Backlog | Subconjunto de elementos del Product Backlog seleccionados para el Sprint actual, más el plan para entregar el Incremento. | Equipo de Desarrollo |
| Incremento | La suma de todos los elementos completados en el Sprint y los anteriores. Debe cumplir con la Definición de Terminado (DoD). | Equipo de Desarrollo |

### 

### **E. Eventos de Scrum (Ajustados a 3 Semanas)**

Estos eventos estructuran el trabajo y consumen el 30% del tiempo total (324 horas).

| **Evento** | **Duración** | **Propósito** | **Frecuencia** |
| --- | --- | --- | --- |
| Duración del Sprint | 3 semanas (15 días laborables) | 6 Sprints en total | Entregar un Incremento de producto utilizable. |
| Planificación del Sprint | 6 horas | Cada 3 semanas | Definir el Objetivo y seleccionar el Sprint Backlog. |
| Daily Scrum | 15 minutos | Diario | Sincronizar actividades y crear un plan diario. |
| Revisión del Sprint | 3 horas | Cada 3 semanas | Inspeccionar el Incremento y obtener *feedback* de los Stakeholders. |
| Retrospectiva del Sprint | 1.5 horas | Cada 3 semanas | Inspeccionar la efectividad del equipo y planificar mejoras. |

### 

### **F. Product Backlog (Priorizado y Estimado)**

Product Backlog basado en las funcionalidades de Gestión Documental, IA y Notificaciones del plan de trabajo.

### **G. Cronograma de Desarrollo (6 Sprints - 18 Semanas)**

El trabajo se divide para asegurar la entrega de un incremento funcional cada 3 semanas, alineado con la capacidad neta de ∼126 horas netas por Sprint.

### 

### 

### **H. Definición de Terminado (DoD) - Criterios de Calidad**

Cada tarea solo se considera "Terminada" (Done) si cumple con *todos* los siguientes criterios:

1. **Código Completo:** El código de la funcionalidad ha sido escrito y revisado por pares.
2. **Pruebas Unitarias:** Se han implementado y pasado las pruebas unitarias.
3. **Pruebas de Integración:** La funcionalidad se integra correctamente con los módulos existentes.
4. **Validación de Requisito:** Cumple con la Historia de Usuario o el Requisito definido.
5. **Despliegue en Entorno de Pruebas:** El código es funcional en el entorno de Staging/Pruebas.
6. **Documentación Mínima:** El código está debidamente comentado y la API/función nueva ha sido documentada.

### 

#### 

### 

### 

### 

### 

### 

### 

**Descripción de la MetodologíaDe**

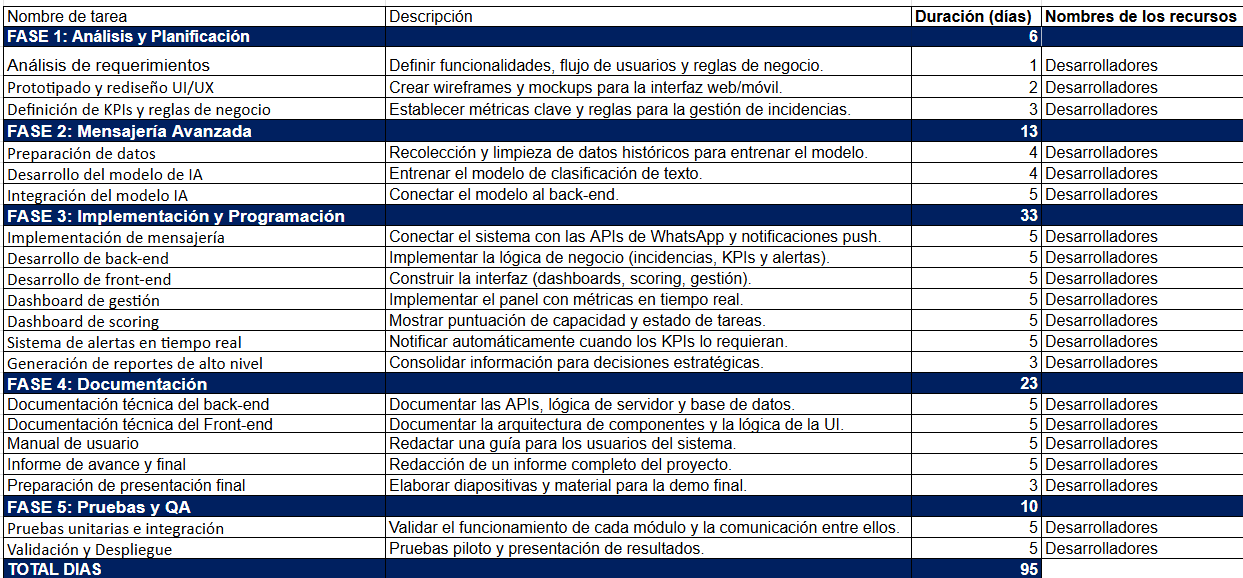
| 6. Evidencias |
| --- |
| A continuación, describe qué evidencias serán evaluadas en el informe de avance y en el informe final de tu proyecto APT. Estas evidencias deben ser acordadas con tu docente. Se entenderá por evidencia los productos que se desarrollen durante el proyecto y cuyo propósito sea visibilizar o documentar cómo se ha implementado el trabajo. |

| **Tipo de evidencia**  **(avance o final)** | **Nombre de la evidencia** | **Descripción** | **Justificación** |
| --- | --- | --- | --- |
| **Avance** | **Informe de análisis de la plataforma web actual** | *Evaluación de usabilidad, rendimiento y accesibilidad de CI-MUN.* | **Permite identificar mejoras concretas para el rediseño de la plataforma web.** |
| **Avance** | **Informe de diseño y mockups** | Documento con arquitectura, BD y prototipos de plataforma web. | **Garantiza una planificación inicial sólida del proyecto.** |
| **Avance** | **Registro de mentoría** | Evidencia de asesorías recibidas en áreas técnicas nuevas. | **Refuerza la curva de aprendizaje y justifica decisiones técnicas.** |
| **Avance** | **Prototipo de mensajería** | Implementación inicial del envío de notificaciones push y mensajes de WhatsApp. | **Demostración del progreso en la funcionalidad de mensajería multicanal.** |
| **Avance** | **Prototipo IA** | Demostración inicial de clasificación y manejos de IA. | **Evidencia avance en IA aplicada.** |
| **Final** | **MVP integrado** | Plataforma Web optimizada con IA y mensajería y mejoras de Usabilidad. | **Producto central del proyecto.** |
| **Final** | **Informe final** | Documentación completa del desarrollo y pruebas. | **Sustenta el trabajo académico.** |

| 7. Plan de Trabajo |
| --- |

| **Plan de Trabajo Proyecto APT** | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Competencia o unidades de competencias** | **Nombre de Actividades/Tareas** | **Descripción Actividades/Tareas** | **Recursos** | **Duración de la actividad** | **Responsable** | **Observaciones** |
| Desarrollo de Software Plataforma Web | Análisis de requerimientos | Definir funcionalidades del sistema, flujo de usuarios y reglas de negocio | Entrevistas, documento de requisitos, Figma, Miro | 2 semanas | **Ignacio** | Requerimientos deben incluir KPIs, scoring y multicanalidad. |
| Experiencia de Usuario | Prototipado y rediseño UI/UX | Crear wireframes y mockups para interfaz web/móvil | Figma, Sketch, feedback usuarios | 2 semanas | **Diego** | Diseñar pantallas para dashboards, scoring y reportes. |
| Inteligencia Artificial | Preparación de datos | Recolección y limpieza de incidencias históricas para entrenar el modelo | Python, Pandas, dataset municipal | 2 semanas | **Felipe** | Datos balanceados para mejorar precisión del modelo. |
| Inteligencia artificial | Desarrollo del modelo de IA | Entrenar un modelo de clasificación de texto para incidencias | TensorFlow, PyTorch, Python | 2 semanas | **Felipe** | Iterar hasta lograr al menos 85% de precisión. |
| Inteligencia Artificial | Integración del modelo IA | Conectar el modelo al back-end para clasificar automáticamente | Laravel PHP, APIs internas, Python | 2 semanas | **Felipe** | Validar con casos de prueba reales. |
| Integración de Sistemas | Implementación de mensajería | Conectar el sistema con la API de WhatsApp y notificaciones push | API WhatsApp, FCM, Laravel PHP, documentación | 2  semanas | **Ignacio** | Gestionar credenciales y estandarizar mensajes en tiempo real. |
| Desarrollo de Software Plataforma Web | Desarrollo de back-end avanzado | Implementar lógica de negocio: incidencias, KPIs y alertas | Laravel PHP, MySQL, Firestore | 3 semanas | **Felipe** | Backend debe disparar alertas cuando KPIs no se cumplan. |
| Desarrollo de Software Plataforma Web | Desarrollo de front-end avanzado | Construir interfaz: dashboards, scoring y gestión de incidencias | React, Tailwind CSS, Chart.js | 3 semanas | **Diego** | Dashboards deben ser interactivos y visuales. |
| Visualización y Gestión | Dashboard de gestión | Implementar panel con métricas en tiempo real | React, Chart.js/Recharts, Laravel APIs | 2 semanas | **Diego** | Mostrar estado de incidencias, tiempos de respuesta y KPIs. |
| Visualización y Gestión | Dashboard de scoring | Mostrar puntuación de capacidad por área y estado de tareas | React, Chart.js, Laravel PHP | 1 semana | **Diego** | Debe reflejar rendimiento operativo de manera clara. |
| Seguimiento y Métricas | Definición de KPIs y reglas de negocio | Establecer métricas clave y reglas para incidencias | Documento de reglas, Notion | 1 semana | **Ignacio** | Necesario para scoring y sistema de alertas. |
| Alertas y Automatización | Sistema de alertas en tiempo real | Notificar automáticamente cuando KPIs o scoring lo requieran | Laravel PHP, FCM, API WhatsApp | 1 semana | **Ignacio** | Alertas automáticas configurables por umbral. |
| Reportes y Estrategia | Generación de reportes de alto nivel | Consolidar información para decisiones estratégicas | Google Data Studio, Laravel export (PDF/Excel) | 1 semana | **Felipe** | Reportes deben incluir gráficos y KPIs resumidos. |
| Pruebas y QA | Pruebas unitarias e integración | Validar funcionamiento de cada módulo y comunicación entre ellos | PHPUnit (Laravel), Jest, Cypress | 2 semanas | **Felipe / Ignacio** | Pruebas de estrés con incidencias masivas. |
| Gestión de Proyectos | Documentación del proyecto | Redacción de informe de avance y final | Google Docs, Notion, Word | 2 semanas | **Diego / Felipe / Ignacio** | Documentar logros, métricas y desafíos. |
| Cierre y Entrega | Validación final y despliegue | Pruebas piloto y presentación de resultados | Servidor demo, Laravel Deploy, manual de usuario | 1 semana | **Equipo completo** | Entrega oficial del sistema CI-MUN. |

| 8. Diagrama EDT |
| --- |



| 8. Carta Gantt |
| --- |

[**Link Carta Gantt**](https://drive.google.com/file/d/1qX0-121Tnow8XLi1JMguvbcU6DbOxDdt/view?usp=drive_link)

| 9. Anexos |
| --- |

* [**Link estudio de Factibilidad – Plataforma CI-MUN**](https://docs.google.com/document/d/12ztlqyYgQpIe1mR3i2eadB05JtZRaRQp5TRl-dBgS7Y/edit?usp=sharing)
* [**Link Carta Gantt**](https://drive.google.com/file/d/1qX0-121Tnow8XLi1JMguvbcU6DbOxDdt/view?usp=drive_link)