}

1. **PARTE I**

| **1. Antecedentes Personales** |
| --- |
| A continuación, se presenta una tabla en la que debes completar la información solicitada. |

| Nombre estudiante | **Guillermo Almendra - Sebastián Miranda** |
| --- | --- |
| Rut | **20.058.625-5 - 20.147.550-3** |
| Carrera | **Ing. Informática** |
| Sede | **Plaza Oeste** |

| **2. Descripción Proyecto APT** |
| --- |
| En la descripción debes señalar brevemente el nombre de tu proyecto APT y las competencias del perfil de egreso que vas a poner en práctica. Si en tu carrera están definidas las áreas de desempeño, también menciona a qué áreas de desempeño está vinculado el proyecto. |

| Nombre del proyecto | Centro de mantención Camiones PepsiCo*.* |
| --- | --- |
| Área (s) de desempeño(s) | Programación de base de datos, Base de datos, Gestión de proyecto |
| Competencias | Gestionar proyectos informáticos, ofreciendo alternativas para la toma de decisiones de acuerdo a los requerimientos de la organización.  Desarrollar la transformación de grandes volúmenes de datos para la obtención de información y conocimiento de la organización a fin de apoyar la toma de decisiones y la mejora de los procesos de acuerdo a las necesidades y conocimiento de la organización  Construir la solución de arquitectura de una solución sistémica que soporte los procesos de negocio de acuerdo a los requerimientos de la organización y los estándares de la industria.  Desarrollar un sistema de software utilizando técnicas que permitan sistematizar los procesos de desarrollo y utilizar los recursos de forma óptima para lograr los objetivos.  Implementar soluciones sistémicas integrales para automatizar y optimizar los procesos de negocio de acuerdo a las necesidades de la organización.  Resolver las vulnerabilidades sistémicas para asegurar que el software construido cumple las normas de seguridad exigidas por la industria. |

| **3. Fundamentación Proyecto APT** |
| --- |
| A continuación, se presentan distintos campos que debes completar con la información solicitada. Esta sección busca que describas en detalle tu proyecto y justifiques su relevancia y pertinencia. |

| Relevancia del proyecto APT | Problema y relevancia para la carrera  PepsiCo Chile gestiona una flota de camiones a nivel nacional. El proceso de ingreso de vehículos al taller y su seguimiento hoy se ejecuta manualmente, utilizando hojas de cálculo y WhatsApp. Esto genera brechas de control, trazabilidad, coordinación y medición de tiempos. El proyecto propone digitalizar y centralizar el flujo con una plataforma web responsiva, lo que es directamente relevante para Ingeniería en Informática: análisis y modelamiento de procesos, diseño de datos, desarrollo web, control de accesos/seguridad, reportabilidad y despliegue en la nube  Contexto y localización El piloto se implementará en el Taller Santa Marta (4 naves), donde el 98% de la flota es eléctrica (con tótem de carga) y ~2% diésel. La patente del vehículo actuará como llave para abrir/relacionar su información. El enfoque local permite probar y refinar antes de escalar a otros talleres a nivel nacional.  A quiénes afecta:   * Choferes: registran llegada y estado de ingreso. * Guardia/Portería: registra entrada/salida por patente y hora. * Mecánicos/Administrativos: ejecutan tareas, suben fotos, dejan observaciones. * Supervisor/Coordinador/Jefe de taller: asigna y monitorea trabajo, controla estados y tiempos. * Indicadores/Bodega/Administración: visibilidad de KPI y repuestos.   La solución impacta en tiempos de atención, coordinación entre áreas y visibilidad del avance.  Aporte de valor:   * Eficiencia operativa: meta de reducción en tiempos de registro; menos errores y duplicidades. * Trazabilidad en tiempo real: estados, pausas y evidencias por vehículo/OT; tablero de taller y bitácora de portería. * Mejor decisión: reportes automáticos (entradas/salidas por día, horas-hombre, duración por etapa/pausa). * Gobierno de datos y seguridad: acceso por perfil, mínima exposición; retención acotada 45–90 días para datasets manejables; operación en la nube. * Escalabilidad: base tecnológica web que permite incorporación futura de cámaras/NFC y otros sitios de mantenimiento. |
| --- | --- |
| Descripción del Proyecto APT | Objetivo Desarrollar una plataforma web responsiva que gestione de manera eficiente y centralizada el ingreso de vehículos al taller, optimizando tiempos, mejorando la comunicación entre actores y asegurando la trazabilidad de la información.  Se implementará un MVP para el Taller Santa Marta que: (1) registra y programa ingresos por patente como llave de datos; (2) opera con perfiles de usuario (guardia/portería, coordinador, mecánico, indicadores, admin); (3) permite crear OT/tareas con inicio, pausas con motivo, reanudaciones y evidencias fotográficas; (4) muestra un tablero por estados (en cola, en proceso, pausado, listo) y una bitácora de portería con entradas/salidas; (5) centraliza documentos y fotos del vehículo; y (6) genera reportes automáticos (entradas vs. salidas, horas‑hombre efectivas, duración por etapa/pausa). Primera versión sin integraciones a ERP/finanzas y enfocada en operación en la nube, con retención acotada de datos.  Cómo aborda la problemática.  Digitaliza un proceso hoy manual (Excel/WhatsApp), eliminando duplicidades y pérdidas de información; entrega visibilidad en tiempo real del avance por vehículo, mejora la coordinación entre choferes, guardia, mecánicos y supervisión, y reduce tiempos de registro. |
| Pertinencia del proyecto con el perfil de egreso | Justificación general El proyecto aborda la digitalización del proceso de ingreso de vehículos al taller y su trazabilidad de punta a punta, lo que requiere análisis de requerimientos, modelado de procesos/datos, desarrollo web, seguridad por perfiles y reportabilidad.  Vinculación con competencias técnicas de Ingeniería en Informática  • Levantamiento y análisis de requerimientos: se modela el flujo Portón→Recepción→Ejecución→Cierre, con estados y criterios de aceptación. • Modelamiento de datos y arquitectura: uso de la \*\*patente\*\* como llave y tablas mínimas (Vehículos, Órdenes, Tareas, Pausas, Evidencias, Usuarios/Roles, BitácoraPortería). • Desarrollo de aplicaciones web: primera versión web responsiva (no móvil nativa), accesible por navegador. • Seguridad y control de accesos: perfiles con visibilidad acotada según rol; gobierno de datos y retención limitada. • Reportabilidad y analítica: indicadores de entradas/salidas, horas‑hombre y duración por etapa/pausa. • Pruebas y aseguramiento de calidad: definición de criterios de aceptación y plan de pruebas para pausas/flujo en paralelo. |
| Relación con los intereses profesionales | Intereses profesionales declarados Desarrollo de soluciones web y en la nube orientadas a procesos reales, con datos y trazabilidad.  Cómo el proyecto refleja esos intereses • Diseño e implementación de un backend/servicios REST y una interfaz web simple y multiplataforma. • Implementación de RBAC (roles/permisos) y manejo de archivos/imágenes como evidencias. • Construcción de reportes automáticos (tiempos, productividad, entradas/salidas) para apoyar decisiones. • Operación en cloud, con almacenamiento de objetos, base de datos y monitoreo básico.  Aporte a desarrollo profesional Consolida competencias clave para roles de desarrollador full‑stack o analista de soluciones: levantamiento de requerimientos, modelado de datos, desarrollo web, seguridad, pruebas y entrega continua. Además, entrega evidencia concreta (demo y métricas) de impacto en eficiencia operativa y coordinación. |
| Factibilidad de desarrollo del Proyecto APT | 1) Duración del semestre (supuesto)Se asume un semestre de16 semanas. El MVP se planifica en 12 semanas, dejando 4 semanas para pruebas, cierre, documentación y defensa.  2) Horas asignadas a la asignatura (estimación) Supuesto de dedicación: 4 h/semana en aula **+** 6 h/semana de trabajo autónomo = 10 h/semana**.** Total estimado: 16 × 10 = 160 horas (ajusta según la carga real de tu malla).  3) Materiales requeridos • Notebook personal con navegador moderno. • Cuenta **c**loud para BD, almacenamiento de objetos (fotos) y despliegue web. • Repositorio Git y sistema de issues (GitHub/GitLab). • Acceso coordinado alTaller Santa Marta para levantamiento/validación (visitas breves). • Plantillas de ingreso/salida y autorización para capturar evidencias fotográficas**.**  4) Factores externos que facilitan• Alcance acotado a un piloto (un taller, flujo definido) y sin integraciones ERP en el MVP. • Apoyo de contraparte (validación de flujo y acceso a información no sensible). • Proceso manual actual bien conocido (Excel/WhatsApp) que permite medir mejoras. • Free tiers de nube suficientes para un MVP (BD, storage, hosting).  5)Factores externos que dificultan y cómo solucionarlos•Acceso al sitio y disponibilidad del personal: agendar con anticipación sesiones cortas (30–45 min) y usar pruebas remotas con datos de ejemplo. • Seguridad/privacidad en fotos y patentes: usar ambientes de prueba con patentes ficticias y difuminar imágenes; retención acotada (90 días). • Conectividad inestable en el taller: capturas simples y ligeras; reintentos y cola de subida de evidencias. • Scope creep (agregar módulos/integraciones no planificadas): bloquear alcance MVP y documentar “futuras iteraciones” en backlog. • Curva de aprendizaje y tiempos: reutilizar componentes/plantillas, priorizar casos críticos y automatizar despliegues. |

1. **PARTE II**

| **4. Objetivos** |
| --- |
| En este apartado debes definir objetivos generales y específicos del Proyecto APT. Es importante aclarar que los objetivos se deben plantear en forma clara, concisa y sin dar mayores explicaciones, es decir, deben entenderse por sí solos. Se sugiere redactarlos utilizando un verbo en infinitivo, pues ello obliga a precisar acciones concretas. |

| Objetivo general | Diseñar, desarrollar y desplegar un MVP web responsivo que digitalice el proceso de ingreso de vehículos al taller  **Portería → Recepción → Ejecución → Cierre**  , con trazabilidad por patente, control de roles y permisos**,** tablero de estados y reportes operativos, logrando una reducción en los tiempos de registro respecto del proceso actual. |
| --- | --- |
| Objetivos específicos | • Levantar y validar con la contraparte el flujo completo, estados y reglas de negocio; entregar diagrama y criterios de aceptación firmados. • Definir el modelo de datos con la patente como llave (Vehículos, Órdenes de Trabajo, Tareas, Pausas, Evidencias, Usuarios, Roles, Bitácora de Portería). • Implementar el registro de ingresos por patente y la gestión de OT/tareas con inicio, pausas con motivo, reanudación y cierre, incluyendo evidencias fotográficas**.** • Desarrollar un tablero de visualización por estados y una bitácora de portería (entradas/salidas por día y hora). • Generar reportes automáticos de entradas vs. salidas, horas-hombre y duración por etapa/pausa, con exportación (CSV/PDF). • Aplicar seguridad por roles (RBAC), auditoría básica y retención acotada de datos (90 días) en un entorno cloud**.** • Realizar pruebas funcionales/UX con casos representativos y documentar resultados; corregir hallazgos críticos antes de la entrega final. • Entregar documentación técnica y manual de usuario y ejecutar una capacitación de 30–45 minutos a usuarios clave. |

| **5. Metodología** |
| --- |
| En el siguiente apartado deberás describir la metodología, propia de tu disciplina, que utilizarás para resolver el proyecto APT antes descrito, incluyendo las etapas y métodos de trabajo. |

| Descripción de la Metodología |
| --- |
| Enfoque general (Metodología Tradicional – SDLC)  Se aplicará un ciclo de vida en fases secuenciales con puertas de control. Cada fase produce artefactos formales y tiene criterios de salida antes de pasar a la siguiente. Se prioriza la trazabilidad, la planificación y la evidencia documental.  Etapas, métodos y entregables (plan base 16 semanas)   * Fase 1 – Requisitos (Semanas 1–2): entrevistas breves y observación del proceso actual; diagrama del flujo Portería→Recepción→Ejecución→Cierre; ERS (especificación de requisitos), RACI, matriz de riesgos; Gantt v1. * Fase 2 – Diseño (Semanas 3–5): modelo de datos (patente como llave), arquitectura web, mockups; plan de pruebas inicial; criterios de aceptación por módulo. * Fase 3 – Construcción (Semanas 6–10): implementación del MVP web (ingreso por patente, OT/tareas, pausas con motivo, evidencias fotográficas, tablero y bitácora); control de versiones y minutas de avance. * Fase 4 – Pruebas (Semanas 11–13): pruebas funcionales/UX y con usuarios clave; reporte de defectos y corrección; verificación de métricas objetivo. * Fase 5 – Implementación y Cierre (Semanas 14–16): despliegue en la nube, manuales y capacitación (30–45 min), plan de soporte; acta de cierre y reporte final de métricas.   Criterios de salida por fase   * Requisitos: ERS aprobado y criterios de aceptación firmados; riesgos priorizados. * Diseño: modelo de datos y arquitectura validados; mockups revisados. * Construcción: funcionalidades del MVP completas y estables; cobertura de pruebas unitarias básica. * Pruebas: 0 defectos críticos abiertos; ≥90% de casos críticos ejecutados y aprobados. * Implementación y Cierre: despliegue operativo, manuales entregados, capacitación realizada, acta de cierre firmada.   Roles y responsabilidades   * Jefatura de Proyecto (PM): planificación (Gantt), coordinación, minutas, riesgos y comunicación con contraparte. * Analista/UX: levantamiento y validación de requisitos, mockups, criterios de aceptación y pruebas con usuarios. * Arquitecto/DBA: modelo de datos, seguridad y desempeño; políticas de retención y respaldo. * Desarrollador/a Full‑Stack: frontend web y API; integración con base de datos; control de versiones. * QA: plan y ejecución de pruebas; reporte y seguimiento de defectos. * DevOps: despliegue, monitoreo básico y respaldos.   Calidad, riesgos y comunicación   * Calidad: definición de DoR/DoD por módulo, checklist de revisiones, estándares de código y control de versiones. * Riesgos: matriz con probabilidad/impacto y planes de respuesta (conectividad, disponibilidad de usuarios, privacidad de imágenes/patentes, alcance). * Comunicación: reunión semanal con contraparte (30–45 min), minutas y tablero de seguimiento; demostraciones en semanas 6, 10 y 14.   Integrantes del equipo   * Guillermo Almendra * Sebastián Miranda   Asignación de roles (principal / apoyo)   * Jefatura de Proyecto / Coordinación con contraparte: Sebastián (principal), Guillermo (apoyo) * Analista de procesos y UX: Guillermo (principal), Sebastián (apoyo) * Arquitectura y modelo de datos / DBA: Sebastián (principal), Guillermo (apoyo) * Desarrollo Frontend Web: Sebastián (principal),Guillermo (apoyo) * Desarrollo Backend / API: Guillermo (principal), Sebastián (apoyo) * Seguridad y RBAC: Sebastián (principal), Guillermo (apoyo) * Reportes y analítica operativa: Sebastián (principal), Guillermo (apoyo) * QA / Pruebas funcionales y de usuario: Guillermo (principal), Sebastián (apoyo) * DevOps / Despliegue: Guillermo (principal), Sebastián (apo yo) * Documentación y capacitación a usuarios: Sebastián (principal), Guillermo (apoyo) |

| **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 | Acta de Constitución del Proyecto | Documento de inicio con objetivo, alcance alto nivel, stakeholders, riesgos iniciales, supuestos, restricciones y aprobaciones. | Define el marco de gestión del proyecto y habilita un control ordenado (stage‑gates). |
| Avance | EDT – Hitos de Desarrollo y Diccionario | Estructura de desglose del trabajo, hitos principales y diccionario (definiciones) de cada paquete de trabajo. | Define el marco de gestión del proyecto y habilita un control ordenado (stage‑gates). |
| Avance | Carta Gantt | Cronograma de 16 semanas con actividades, responsables (Guillermo/Sebastián), dependencias e hitos. | Define el marco de gestión del proyecto y habilita un control ordenado (stage‑gates). |
| Avance | Matriz de Responsabilidades (RACI) | Asignación de Responsable, Aprobador, Consultado e Informado por actividad clave. | Define el marco de gestión del proyecto y habilita un control ordenado (stage‑gates). |
| Avance | Matriz de Riesgos | Identificación, probabilidad/impacto, disparadores y planes de respuesta; monitoreo por semana. | Define el marco de gestión del proyecto y habilita un control ordenado (stage‑gates). |
| Avance | Plan de Pruebas Inicial | Estrategia de pruebas, alcance, roles, ambientes, criterios de aceptación y casos preliminares. | Define el marco de gestión del proyecto y habilita un control ordenado (stage‑gates). |
| Avance | Documento de Especificación de Requerimientos (ERS) | Requisitos funcionales y no funcionales, reglas de negocio, actores, flujo Portería→Recepción→Ejecución→Cierre y criterios de aceptación. | Asegura claridad del qué y cómo antes de construir; reduce retrabajo. |
| Avance | Modelos de Datos y Arquitectura | Modelo lógico con \*\*patente\*\* como llave; entidades Vehículo/OT/Tareas/Pausas/Evidencias/Usuarios/Roles/Bitácora y arquitectura web en la nube. | Asegura claridad del qué y cómo antes de construir; reduce retrabajo. |
| Avance | Mockups de Interfaz de Sistemas | Prototipos de pantallas para registro por patente, tablero por estados y bitácora de portería. | Asegura claridad del qué y cómo antes de construir; reduce retrabajo. |
| Avance | Plan de Calidad y de Costos | Estándares de código, revisión y métricas; estimación de costos (nube free‑tier, dominio opcional) y criterios de aceptación de calidad. | Asegura claridad del qué y cómo antes de construir; reduce retrabajo. |
| Avance | Código Fuente y Documentación Interna | Repositorio con commits, ramas, README, comentarios y guías de despliegue/entorno. | Evidencia progreso técnico y control de alcance durante la construcción. |
| Avance | Minuta Control de la Programación | Registro de avances, issues, acuerdos y decisiones por semana; evidencias de demos intermedias. | Evidencia progreso técnico y control de alcance durante la construcción. |
| Avance | Verificación de Alcances | Checklist de requisitos implementados vs. ERS; evidencia de pruebas unitarias/funcionales por módulo. | Evidencia progreso técnico y control de alcance durante la construcción. |
| Final | Plan de Pruebas Final y Reporte de Estatus Final | Ejecución de casos, resultados, defectos y correcciones; criterio de salida 0 críticos; métricas objetivo alcanzadas. | Demuestra operación, transferencia y sostenibilidad post‑entrega. |
| Final | Plan de Implantación, Soporte y Mantenimiento | Procedimiento de despliegue, rollback, respaldos, monitoreo básico y mesa de ayuda/responsables. | Demuestra operación, transferencia y sostenibilidad post‑entrega. |
| Final | Manuales de Usuarios y Administrador | Guía paso a paso para usuarios y administración (creación de usuarios, roles, backups, etc.). | Demuestra operación, transferencia y sostenibilidad post‑entrega. |

| **7. Plan de Trabajo** |
| --- |
| En la siguiente tabla define la planificación de tu Proyecto APT de acuerdo a lo requerido. |

| **Plan de Trabajo Proyecto APT** | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| Competencia o unidades de competencias | Nombre de Actividades/Tareas | Descripción Actividades/Tareas | Recursos | Duración de la actividad | Responsable[[1]](#footnote-0) | Observaciones |
| Gestión de proyectos / comunicación | Acta de Constitución del Proyecto | Definir objetivo, alcance alto nivel, stakeholders, supuestos, restricciones y aprobaciones. | Plantilla de acta, reunión con contraparte, Google Docs | Sem 1 | Sebastián (principal) | Puerta de salida para iniciar formalmente. Dependencia: disponibilidad de contraparte. |
| Gestión de proyectos | EDT – Hitos de Desarrollo y Diccionario | Desglosar el trabajo por fases/entregables y definir cada paquete. | Plantilla EDT, Miro/Draw.io | Sem 1 | Sebastián | Base para Gantt y RACI. |
| Gestión de proyectos | Carta Gantt v1 | Cronograma de 16 semanas con dependencias e hitos. | Herramienta Gantt (Sheets/Notion/Project) | Sem 1 | Sebastián | Actualizar en cada hito (v2/v3). |
| Trabajo en equipo | Matriz de Responsabilidades (RACI) | Asignar R/A/C/I por actividad clave. | Plantilla RACI | Sem 1 | Guillermo | Mantener sincronizada con el Gantt. |
| Gestión de riesgos | Matriz de Riesgos | Identificar riesgos, probabilidad/impacto y planes de respuesta. | Plantilla riesgos | Sem 1 | Sebastián | Revisar semanalmente. |
| Calidad / QA | Plan de Pruebas Inicial | Definir alcance, roles, ambientes y criterios de aceptación. | Plantilla plan de pruebas | Sem 1–2 | Guillermo | Se detalla en Fase de Pruebas. |
| Levantamiento y análisis de requisitos | Entrevistas y observación de proceso actual | Recolectar información del flujo Portería→Recepción→Ejecución→Cierre. | Agenda, cuestionario, visitas breves | Sem 1–2 | Guillermo (principal) | Minutas por sesión. |
| Requisitos / documentación | ERS – Especificación de Requerimientos | Definir requisitos funcionales/no funcionales, reglas y criterios de aceptación. | Google Docs, versión controlada | Sem 2 | Sebastián (redacción), Guillermo (revisión) | Aprobación de contraparte (stage‑gate). |
| Modelado de datos / arquitectura | Modelo de Datos y Arquitectura | Modelo lógico con patente como llave; definición de componentes web y BD. | Draw.io/Mermaid, Google Docs | Sem 3 | Sebastián | Validar con impacto en reportes/KPI. |
| UX / diseño de interfaz | Mockups de Interfaz | Prototipos: registro por patente, tablero y bitácora. | Figma/Draw.io | Sem 3–4 | Guillermo | Prueba de concepto con 2–3 usuarios. |
| Calidad / gestión | Plan de Calidad y de Costos | Definir estándares de código, revisiones, estimación cloud (free tier). | Plantilla calidad, calculadora cloud | Sem 4–5 | Sebastián | Ajustar con uso real de storage/fotos. |
| Desarrollo backend | API y lógica de negocio | Servicios para registro por patente, OT/tareas, pausas con motivo, evidencias. | Python/Node/Java, Postman, GitHub | Sem 6–9 | Sebastián (principal) | Definir endpoints y validaciones. |
| Desarrollo frontend | Interfaz web responsiva | Pantallas para registro, tablero y bitácora. | Framework web, GitHub | Sem 6–9 | Guillermo (principal) | Pruebas de usabilidad rápidas. |
| Base de datos | Scripts de creación y seed | DDL y datos de ejemplo; políticas de retención/backups. | Motor BD cloud, SQL | Sem 6–7 | Sebastián | Ambiente de prueba aislado. |
| Seguridad | RBAC y auditoría básica | Roles/permisos por perfil; logging mínimo. | Auth (JWT/OAuth), logs | Sem 8–9 | Sebastián | Revisión de privacidad (patentes/fotos). |
| Reportabilidad | Reportes y exportaciones | Entradas/salidas, horas‑hombre, duración por etapa/pausa (CSV/PDF). | Librería reportes, Sheets/CSV | Sem 9–10 | Sebastián | Validar definiciones con negocio. |
| QA / pruebas | Ejecución de pruebas funcionales/UX | Casos de prueba, registro de defectos y corrección. | Plan de pruebas, tracker de issues | Sem 11–12 | Guillermo (ejecución), Sebastián (fix) | Criterio de salida: 0 críticos abiertos. |
| Demostración de avance | Demo interna y con contraparte | Mostrar MVP, recoger feedback y ajustar. | Ambiente de prueba, minutas | Sem 12–13 | Ambos | Preparar video corto como respaldo. |
| DevOps / operación | Despliegue funcional | Publicación del MVP y verificación post‑deploy. | Cuenta cloud, dominios opcionales | Sem 14 | Sebastián | Plan de rollback y backups. |
| Documentación / formación | Manual de Usuario y Administrador + Capacitación | Guías y sesión de 30–45 min a usuarios clave. | Docs, presentación, lista de asistencia | Sem 15 | Guillermo (manual usuario), Sebastián (admin) | Grabar la sesión (opcional). |
| Cierre del proyecto | Acta de Cierre y Métricas | Comparativo de tiempos antes/después; lecciones aprendidas y backlog futuro. | Plantilla de acta, dashboard KPI | Sem 16 | Sebastián (redacción), Guillermo (datos) | Cierre formal y traspaso. |

| **8. Carta Gantt** |
| --- |

| **Actividad** | **Fase 1** | | | | **Fase 2** | | | | | | | | | | | | | **Fase 3** | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **S 1** | **S 2** | **S 3** | **S 4** | **S 5** | **S 6** | | **S 7** | **S 8** | **S 9** | **S 10** | **S 11** | **S 12** | **S 13** | **S 14** | **S 15** | **S 16** | | **S 17** | **S 18** |
| Acta de Constitución |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| EDT y Diccionario |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Carta Gantt |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Matriz RACI |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Matriz de Riesgos |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Plan de Pruebas Inicial |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Entrevistas y Observación |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| ERS (Requisitos) |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Modelo de Datos y Arquitectura |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Mockups de Interfaz |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Plan de Calidad y Costos |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Scripts BD (DDL/seed) |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| API y lógica de negocio (Backend) |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Interfaz web (Frontend) |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Seguridad RBAC / Auditoría |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Reportes y Exportaciones |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Pruebas funcionales/UX |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Demo interna y con contraparte |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Despliegue funcional |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Manuales y Capacitación |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Acta de Cierre y Métricas |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Buffer / Ajustes / Defensa |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  |

1. En caso de que el Proyecto APT sea grupal, en esta columna deben indicar el nombre de los responsables de cada tarea o actividad. Esto posteriormente permitirá diferenciar la evaluación por cada integrante. [↑](#footnote-ref-0)