

Guía1. Definición Proyecto APT

Asignatura Capstone

A. PARTE I

1. Antecedentes Personales

A continuación, se presenta una tabla en la que debes completar la información solicitada.

Nombre estudiante	Tihare Maldonado, Janis Montecino y Matias Morales
Rut	21.116.737-8, 21.088.684-2 y 21.503.300-7
Carrera	Ingeniería en 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	Plataforma de Gestión Digital para el Ingreso de Vehículos al Taller – PepsiCo Chile
Área (s) de desempeño(s)	<ul style="list-style-type: none"> • Gestión de proyectos informáticos • Análisis y evaluación de soluciones informáticas • Desarrollo de software • Aseguramiento de calidad de software
Competencias	<ul style="list-style-type: none"> • Presentar soluciones informáticas mediante un análisis exhaustivo de los procesos y necesidades. • Administrar proyectos tecnológicos facilitando el proceso de toma de decisiones de la entidad.

	<ul style="list-style-type: none">• Diseñar la estructura arquitectónica de una solución técnica que cumpla con los estándares sectoriales.• Elaborar software aplicando los principios de desarrollo y mantenimiento más recomendados.• Ejecutar soluciones sistémicas completas con el fin de automatizar operaciones comerciales.• Llevar a cabo evaluaciones de certificación que garanticen la calidad y la seguridad del software.
--	---

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 justifiquen su relevancia y pertinencia.

<p>Relevancia del proyecto APT</p>	<p>PepsiCo Chile administra una amplia flota de camiones y actualmente gestiona su ingreso a talleres mediante hojas de cálculo y comunicación por WhatsApp. Este método manual genera dificultades de control, trazabilidad y coordinación, además de retrasos en los tiempos de respuesta.</p> <p>El proyecto plantea el desarrollo de una plataforma web centralizada que permita digitalizar y automatizar este proceso, asegurando mayor eficiencia, mejor comunicación entre choferes, supervisores y mecánicos, y un registro confiable de la información.</p> <p>La implementación tendrá un impacto directo en la sucursal de Maipú, donde se coordina la flota, aportando rapidez y orden en la operación. Además, se proyecta como un modelo escalable que podría replicarse en otras instalaciones de la compañía, fortaleciendo la gestión tecnológica y alineándose con los desafíos de la Ingeniería en Informática.</p>
<p>Descripción del Proyecto APT</p>	<p>El proyecto busca desarrollar una plataforma web centralizada para gestionar el ingreso de vehículos al taller de PepsiCo Chile, optimizando tiempos, reduciendo errores y garantizando la trazabilidad de la información.</p> <p>Actualmente, este proceso se realiza de forma manual con planillas y mensajería, lo que provoca desorden y falta de control. La solución propuesta permitirá registrar y programar ingresos de vehículos, gestionar estados y pausas, almacenar documentación y generar reportes automáticos en tiempo real.</p> <p>El desarrollo se abordará con una metodología tradicional, avanzando en fases de planificación, análisis, diseño, construcción, implementación y cierre. Cada etapa contará con entregables concretos, asegurando un control riguroso del proyecto y una solución alineada con los objetivos de la empresa y el perfil profesional de Ingeniería en Informática.</p>
<p>Pertinencia del proyecto con el perfil de egreso</p>	<p>El Proyecto APT se vincula directamente con el perfil de egreso de Ingeniería en Informática, ya que contempla analizar, diseñar,</p>

	<p>desarrollar e implementar un sistema que resuelve una necesidad real en PepsiCo Chile: la gestión del ingreso de vehículos al taller.</p> <p>La propuesta integra funciones clave del ingeniero informático, como la automatización de procesos, la integración de tecnologías y el aseguramiento de la calidad del software. Las competencias aplicadas incluyen: diseñar soluciones tecnológicas para digitalizar procesos manuales, analizar requerimientos funcionales para responder a las necesidades de choferes y supervisores, aplicar metodologías de gestión de proyectos para cumplir plazos y garantizar calidad y trazabilidad de datos.</p> <p>En síntesis, este proyecto ofrece una instancia práctica para demostrar las competencias adquiridas en la carrera, aportando una solución tecnológica con valor real para la organización y reflejando el perfil profesional del Ingeniero en Informática.</p>
Relación con los intereses profesionales	<p>El Proyecto APT se relaciona directamente con los intereses profesionales del equipo, enfocados en el desarrollo de software, la gestión de proyectos informáticos y el análisis de sistemas, áreas clave en la formación de Ingeniería en Informática.</p> <p>El proyecto refleja estos intereses al requerir planificación, análisis, diseño e implementación de una solución tecnológica real, aplicando metodologías estructuradas y entregables definidos. Esto permite fortalecer competencias prácticas como la gestión formal de proyectos, la documentación técnica y la validación de soluciones mediante pruebas.</p> <p>De esta forma, el proyecto contribuye al desarrollo profesional de los estudiantes al consolidar conocimientos técnicos y demostrar la capacidad de enfrentar problemáticas reales de la industria, reforzando su perfil como futuros Ingenieros en Informática capaces de aportar soluciones confiables y de valor para las organizaciones.</p>
Factibilidad de desarrollo del Proyecto APT	<p>El desarrollo del Proyecto APT es factible dentro de las 12 semanas del semestre académico, ajustándose a las horas asignadas y a los plazos de la asignatura. Se utilizarán tecnologías de código abierto y herramientas gratuitas o de bajo costo, como VS Code, Git/GitHub, Figma y gestores de bases de datos, lo que asegura que el proyecto se mantenga dentro del presupuesto.</p> <p>Entre los factores que facilitan el desarrollo destacan la existencia de un caso real en PepsiCo, la disponibilidad de documentos y reportes</p>

	<p>actuales para levantar requerimientos, y la aplicación de una metodología estructurada que brinda control y reduce incertidumbre.</p> <p>Como posibles dificultades se consideran la falta de integración inicial con sistemas ERP y la resistencia de usuarios al cambio de procesos manuales. Para mitigarlas, se propone una versión web responsiva con funcionalidades críticas y el apoyo de manuales y capacitaciones.</p> <p>En conclusión, el proyecto es viable en tiempo, recursos y alcance, contando con las competencias necesarias y un plan estructurado que asegura su implementación exitosa.</p>
--	---

B. 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	Desarrollar una plataforma tecnológica centralizada que permita gestionar de forma eficiente el ingreso de vehículos al taller de PepsiCo Chile, con el propósito de optimizar los tiempos de registro y control, reducir errores en la coordinación de tareas y asegurar la trazabilidad de la información, aportando una solución digital confiable a un proceso totalmente manual.
Objetivos específicos	<ul style="list-style-type: none"> ● Diseñar un sistema de agenda para programar y controlar el ingreso de vehículos. ● Implementar perfiles de usuario diferenciados (chofer, supervisor, mecánico). ● Incorporar funciones de carga y consulta de documentos, fotos e informes. ● Automatizar notificaciones y generación de reportes. ● Evaluar el impacto en la reducción de tiempos y errores del proceso.

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

El desarrollo del proyecto se llevará a cabo utilizando la metodología tradicional de gestión y desarrollo de software, en la cual las fases se realizan de manera secuencial y estructurada, permitiendo planificar, organizar y controlar cada etapa del ciclo de vida del sistema.

Fases principales del proyecto:

1. **Planificación inicial** → Definir el alcance del proyecto, elaborar el acta de constitución, la estructura de desglose de trabajo (EDT), la carta Gantt y la matriz de riesgos.
2. **Análisis y diseño** → Levantamiento y documentación de requerimientos, modelado de procesos de negocio, elaboración de casos de uso y diagramas UML, diseño de base de datos y mockups de interfaz.
3. **Construcción** → Configuración del ambiente de desarrollo, programación de los módulos del sistema, creación de la base de datos y realización de pruebas unitarias.
4. **Implementación** → Integración de componentes, ejecución de pruebas finales, validación de requerimientos y generación de manuales de usuario y administrador.
5. **Cierre** → Capacitación de los usuarios finales, despliegue del sistema en ambiente productivo, entrega de documentación y plan de soporte inicial.

Definición de roles:

1. Jefe de Proyecto / Project Manager (Janis Montecino)

- Planifica, organiza y controla todas las fases del proyecto.
- Elabora el cronograma (Carta Gantt), la EDT, la matriz de riesgos y asegura que se cumplan plazos y costos.
- Es el “dueño” del control y seguimiento.

2. Analista de Negocios / Analista Funcional (Janis Montecino)

- Se encarga del **levantamiento de requerimientos** con los usuarios.
- Traduce las necesidades del negocio en especificaciones técnicas (documento de requerimientos, casos de uso).
- Es el puente entre los usuarios y el equipo técnico.

3. Arquitecto de Software (Tihare Maldonado)

- Define la **arquitectura del sistema** (diagrama de arquitectura, modelo de datos, diseño de clases).
- Se asegura de que el diseño sea escalable, seguro y cumpla estándares.

4. Diseñador UX/UI (Tihare Maldonado)

- Elabora **mockups, diagramas de interacción y prototipos de interfaz**.
- Se centra en la experiencia del usuario y la usabilidad del sistema.

5. Desarrolladores / Programadores (Matias Morales)

- Implementan la solución siguiendo los diseños y especificaciones.
- Crean la base de datos, programan las funcionalidades y realizan pruebas unitarias.

6. Tester / QA (Aseguramiento de Calidad) (Matias Morales)

- Diseña y ejecuta los planes de pruebas (iniciales y finales).
- Detecta errores y asegura que el software cumpla con los requerimientos y la calidad esperada.

7. Usuarios Finales / Stakeholders

- Participan validando requerimientos, revisando prototipos y probando el sistema.
- Son los beneficiarios directos de la solución (ej: choferes, supervisores y mecánicos en tu caso).

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 inicial que define objetivos, alcance y responsables.	Formaliza el inicio del proyecto y alinea expectativas.
Avance	EDT – Hitos de desarrollo y diccionario	Estructura de Desglose del Trabajo con hitos definidos.	Facilita la organización y control de actividades.
Avance	Carta Gantt	Cronograma visual con las actividades del proyecto.	Permite planificar tiempos y recursos.
Avance	Definición de Responsabilidades (RAM o RACI)	Matriz que asigna roles a cada tarea.	Asegura claridad en la responsabilidad de los involucrados.
Avance	Matriz de Riesgos	Identificación y análisis de riesgos potenciales.	Ayuda a anticipar problemas y planificar mitigaciones.
Avance	Plan de Pruebas Inicial	Estrategia de pruebas preliminares del sistema.	Permite definir cómo se validará la calidad del proyecto.
Avance	Modelo de Proceso de Negocio	Representación gráfica del flujo actual del proceso.	Da contexto y base para el rediseño.
Avance	Documento de Especificación de Requerimientos	Detalle de las funcionalidades esperadas del sistema.	Garantiza que el sistema responda a necesidades reales.
Avance	Casos de Uso Específicos UML	Diagramas de interacción entre usuarios y sistema.	Aseguran que se contemplen todos los escenarios.
Avance	Diagrama de Arquitectura	Representación general de la solución tecnológica.	Define la estructura técnica de la plataforma.
Avance	Mockups Interfaz de Sistemas completo	Prototipos visuales de pantallas y flujos.	Facilitan la validación temprana con usuarios.
Avance	Diagrama de Actividad UML	Flujo de actividades del sistema.	Complementa la visión de procesos.

Avance	Modelo E-R (Entidad Relación)	Diseño conceptual de la base de datos.	Organiza las entidades y relaciones de la información.
Avance	Modelo Relacional Normalizado	Versión técnica del modelo de datos.	Asegura consistencia y eficiencia en almacenamiento.
Avance	Diccionario de Datos	Definición de cada atributo de la base de datos.	Entrega detalle técnico para implementación.
Avance	Diagrama de Clases	Estructura de objetos y sus relaciones.	Fundamenta el diseño orientado a objetos.
Avance	Plan de Calidad	Estrategia para asegurar calidad en procesos y entregables.	Permite monitorear estándares de calidad.
Avance	Plan de Costos	Estimación y control de gastos asociados.	Da visibilidad al uso de recursos.
Avance	Plan de Riesgos	Estrategia formal de mitigación de riesgos.	Complementa la matriz de riesgos inicial.
Avance	Plan de Comunicación	Definición de canales y frecuencia de comunicación.	Asegura coordinación entre participantes.
Avance	Plan de Adquisiciones	Definición de compras o insumos requeridos.	Permite anticipar los recursos necesarios.
Avance	Definición de Actividades detalladas (EDT)	Desglose detallado de tareas.	Entrega mayor control de ejecución.
Final	Implementación de ambiente de desarrollo	Configuración del entorno técnico de trabajo.	Permite iniciar el desarrollo de la plataforma.
Final	Base de datos, tablas y scripts	Creación de tablas y estructuras de datos.	Implementa el modelo relacional diseñado.
Final	Minuta Control de la Programación	Registro de avances en el desarrollo.	Da seguimiento a las actividades programadas.
Final	Matriz seguimiento Status del Proyecto	Documento con estado actualizado del proyecto.	Asegura control de hitos y progreso.
Final	Verificación de Alcances	Validación del cumplimiento de requerimientos.	Garantiza que se cumpla lo comprometido.
Final	Matriz Gestión Control de Cambio	Registro de modificaciones en alcance o requerimientos.	Mantiene trazabilidad de cambios.
Final	Plan de Pruebas Final	Estrategia de validación completa del sistema.	Asegurar la calidad antes de la entrega.
Final	Reporte estatus final del Proyecto	Documento que resume resultados y avances.	Refleja cumplimiento y cierre formal.

Final	Plan de Implantación	Estrategia para poner en marcha la solución.	Facilita la transición del desarrollo al uso real.
Final	Plan de Soporte y Mantenimiento	Definición de actividades de soporte post-implementación.	Garantiza continuidad del sistema.
Final	Plan de Capacitación	Estrategia de formación de usuarios.	Facilita la adopción de la plataforma.
Final	Manuales de Usuario y Administrador	Documentación operativa y técnica del sistema.	Permiten uso eficiente y soporte de la solución.

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	Observaciones
1.1 Presentar soluciones informáticas mediante un análisis exhaustivo de los procesos y necesidades.	Levantamiento de requerimientos.	Reuniones de stakeholders para documentar procesos actuales y necesidades.	Entrevistas, planillas Excel, documento de requerimientos.	1 semana.	Analista de Negocios (Janis Montecino)	La disponibilidad de los stakeholders puede retrasar la información.
1.2 Presentar soluciones informáticas mediante un análisis exhaustivo de los procesos y necesidades.	Análisis de procesos y documentación.	Elaborar diagramas de flujo y mapas de procesos para detectar ineficiencias.	Herramientas de diagramación (Draw.io, Lucidchart).	1 semana.	Analista de Negocios (Janis Montecino)	Validar procesos con stakeholders para evitar errores de interpretación.

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.

		y áreas de mejora.				
2.1 Administrar proyectos tecnológicos facilitando el proceso de toma de decisiones de la entidad.	Elaboración de EDT y Carta Gantt.	Planificación y calendarización de actividades con desglose detallado de tareas.	Plantillas EDT, MS Project o Excel.	1 semana.	Jefe de Proyecto / PM (Janis Montecino)	Requiere constante actualización frente a cambios del proyecto.
2.2 Administrar proyectos tecnológicos facilitando el proceso de toma de decisiones de la entidad.	Monitoreo y control del proyecto.	Revisar avances, identificar retrasos y ajustar cronogramas según sea necesario.	MS Project, reportes de avance, reuniones de seguimiento.	Durante todo el proyecto.	Jefe de Proyecto / PM (Janis Montecino)	Depende de la comunicación efectiva con el equipo y stakeholders.
3.1 Diseñar la estructura arquitectónica de una solución técnica que cumpla con los estándares sectoriales.	Diseño de la arquitectura del sistema.	Creación de diagramas de arquitectura, modelo entidad-relación y diseño de clases.	Herramientas de diagramación, laptop.	2 semanas.	Arquitecto de Software (Tihare Maldonado)	Diseño debe garantizar escalabilidad, seguridad e integración futura.
3.2 Diseñar la estructura arquitectónica de una solución técnica que cumpla con los estándares sectoriales.	Revisión de estándares y buenas prácticas.	Asegurar que la arquitectura cumpla normas de seguridad, rendimiento y mantenimiento.	Documentación técnica, frameworks, plantillas de buenas prácticas.	1 semana.	Arquitecto de Software (Tihare Maldonado)	Involucra coordinación con desarrolladores y QA.

4.1 Elaborar software aplicando los principios de desarrollo y mantenimiento más recomendados.	Desarrollo de módulos funcionales.	Programación de funcionalidades como ingreso de vehículos, gestión de estados, carga de documentos y generación de reportes.	Visual Studio Code, frameworks web, GitHub, base de datos SQL.	4 semanas.	Desarrolladores / Programadores (Matías Morales)	Posibles retrasos por cambios en los requerimientos.
4.2 Elaborar software aplicando los principios de desarrollo y mantenimiento más recomendados.	Revisión de código y pruebas unitarias.	Implementar pruebas unitarias y revisiones de código para asegurar calidad.	GitHub, herramientas de testing, documentación de código.	2 semanas (paralelo al desarrollo)	Desarrolladores / Programadores (Matías Morales)	Recomendable revisión cruzada por otro desarrollador.
5.1 Ejecutar soluciones sistémicas completas con el fin de automatizar operaciones comerciales.	Integración de módulos y pruebas internas.	Unificación de frontend, backend y base de datos en ambiente de desarrollo.	Servidor local, software de pruebas, entornos virtuales.	2 semanas.	Desarrolladores / Programadores (Matías Morales)	Posibles incompatibilidades técnicas durante la integración.
5.2 Ejecutar soluciones sistémicas completas con el fin de automatizar operaciones comerciales.	Pruebas de rendimiento y ajustes finales.	Evaluar velocidad, eficiencia y estabilidad del sistema integrado.	Herramientas de testing de carga, scripts de automatización.	1 semana.	Desarrolladores / Programadores (Matías Morales), QA Tester	Requiere coordinación con QA para validar resultados.
6.1 Llevar a cabo evaluaciones de certificación que	Ejecución del Plan de Pruebas Final.	Realización de pruebas unitarias, integración y aceptación	Datos de prueba, checklist QA, manuales de usuario.	2 semanas.	Tester / QA (Matías Morales)	Depende de la disponibilidad y validación de usuarios

garanticen la calidad y la seguridad del software.		junto a usuarios finales.				para certificar la solución.
6.2 Llevar a cabo evaluaciones de certificación que garanticen la calidad y la seguridad del software.	Documentación de resultados y ajustes.	Registrar hallazgos de pruebas y coordinar correcciones de errores antes de la entrega final.	Plantillas de informe QA, herramientas de seguimiento de bugs (Jira, Trello).	1 semana.	Tester / QA (Matías Morales)	Crucial para la entrega formal y aceptación del sistema.
6.3 Llevar a cabo evaluaciones de certificación que garanticen la calidad y la seguridad del software.	Validación con usuarios finales.	Revisar que el sistema cumpla con los requerimientos y expectativas del usuario.	Sesiones de prueba con usuarios finales	1 semana.	Usuarios Finales / Stakeholders	Garantiza que la solución sea funcional y usable.

8. Carta Gantt

Busca un formato de Carta Ganntt que te acomode y organiza en este las actividades planificadas en el punto anterior considerando el periodo asignado para el desarrollo de tu Proyecto APT. Debes mantener la temporalidad del periodo académico en el desarrollo de las tres fases que contempla la Asignatura de Portafolio de Título.

[illegible]

[illegible]

Fase 1



Pruebas Final.																		
Documentación de resultados y ajustes.																		
Validación con usuarios finales.																		