

Informe de Autoevaluación

Definición del Proyecto APT

Integrantes: Joaquin Marin

Profesor: Fabian Alvarez

Curso: Capstone

Fecha: 28/08/2025

## **1. Descripción del Proyecto APT**

El Proyecto APT consiste en el desarrollo de una plataforma web centralizada para digitalizar y optimizar el proceso de ingreso de vehículos al taller de mantenimiento de PepsiCo Chile. Actualmente, este proceso se realiza de forma manual mediante planillas Excel y comunicación por WhatsApp, lo que genera problemas de control, trazabilidad y eficiencia. La plataforma permitirá programar ingresos, gestionar estados en tiempo real, subir documentación y generar reportes automáticos, mejorando significativamente la coordinación entre choferes, supervisores y mecánicos.

Este proyecto es altamente relevante para el campo laboral de la ingeniería informática, ya que aborda un problema real de logística y gestión de flotas, aplicando conocimientos en desarrollo web, bases de datos, gestión de proyectos y automatización de procesos. Además, se alinea con la demanda actual de digitalización en la industria.

## **2. Relación con las competencias del perfil de egreso**

Este proyecto se relaciona directamente con las siguientes competencias del perfil de egreso:

* **Gestionar proyectos informáticos:** Se aplicará en la planificación, ejecución y control del desarrollo del software, asegurando que cumpla con los requisitos y el plazo establecido.
* **Construir modelos de datos:** Se diseñará e implementará una base de datos relacional escalable para gestionar vehículos, ingresos, tareas y documentos.
* **Desarrollar una solución de software:** Se utilizarán metodologías ágiles y técnicas de desarrollo modernas para construir una plataforma funcional y mantenible.
* **Pruebas y aseguramiento de calidad**: Diseñar pruebas de validación del sistema, verificando su correcto funcionamiento en escenarios reales.

## **3. Relación con mis intereses profesionales**

Mi interés profesional se centra en el desarrollo de software aplicado a resolver problemas empresariales concretos. Este proyecto me permite integrar conocimientos técnicos en un contexto real, trabajar con metodologías tradicionales y mejorar habilidades de gestión de proyectos. Además, me interesa especializarme en el desarrollo de sistemas de información que optimicen procesos operativos, lo cual se alinea perfectamente con el objetivo de esta plataforma.

## **4. Argumento del por qué el proyecto es factible de realizarse dentro de la asignatura**

El proyecto es factible por las siguientes razones:

* **Tiempo:** El plazo de 18 semanas es suficiente para desarrollar una versión funcional (MVP) utilizando tecnologías ágiles y un stack moderno pero accesible.
* **Recursos:** Se utilizarán tecnologías de código abierto y gratuitas, lo que elimina costos de licencias.
* **Alcance delimitado:** El alcance inicial está bien definido y no incluye integraciones complejas con ERPs ni módulos financieros.
* **Habilidades:** Cuento con los conocimientos técnicos necesarios en programación, bases de datos y desarrollo web, adquiridos durante la carrera.  
    
  En caso de posibles dificultades (como retrasos en alguna funcionalidad), se priorizarán las features críticas para cumplir con el objetivo mínimo viable.

## **5. Objetivos claros y coherentes**

### **Objetivo General:**

Desarrollar una plataforma web que gestione eficientemente el ingreso de vehículos al taller, mejorando la trazabilidad, comunicación y control del proceso.

### **Objetivos Específicos:**

1. Digitalizar la programación y registro de ingresos de vehículos.
2. Implementar perfiles de usuario con permisos diferenciados.
3. Permitir la gestión de estados y pausas en tiempo real.
4. Generar reportes automáticos de productividad y tiempos.
5. Centralizar la documentación e imágenes por vehículo.

## **6. Metodología de trabajo**

Se utilizará una metodología tradicional en cascada (Waterfall), la cual es pertinente para proyectos con requisitos bien definidos y estables desde el inicio, como es el caso de esta plataforma. Esta metodología permite una planificación detallada y un control exhaustivo de cada fase, asegurando que se cumplan todos los requerimientos disciplinares antes de avanzar a la siguiente etapa. El proceso se estructurará en las siguientes fases secuenciales:

1. **Análisis de Requisitos:** Se recopilarán y documentarán de manera exhaustiva todos los requisitos funcionales y no funcionales del sistema a partir del caso de estudio proporcionado. El resultado será un documento de Especificación de Requisitos de Software (ERS).
2. **Diseño del Sistema:** Se crearán los diseños arquitectónicos y detallados del sistema. Esto incluye el diseño de la base de datos (modelo ER), el diseño de la API RESTful y los mockups de la interfaz de usuario (UI/UX) para cada perfil de usuario (Chofer, Mecánico, Supervisor).
3. **Implementación (Desarrollo):** Se procederá a codificar el sistema según los diseños establecidos en la fase anterior. Esta fase se dividirá en dos sub-fases principales:  
   1. **Desarrollo del Backend:** Creación de la base de datos, la API y la lógica de negocio.
   2. **Desarrollo del Frontend:** Implementación de las interfaces de usuario responsivas.
4. **Pruebas (Testing):** Una vez completado el desarrollo, se ejecutará una fase de pruebas rigurosa para identificar y corregir defectos. Esto incluirá pruebas unitarias, de integración y de sistema (usabilidad y rendimiento).
5. **Despliegue (Implementación) y Mantenimiento:** Se desplegará la versión funcional de la plataforma en un entorno de producción simulado (ej: un servidor cloud como Heroku o Railway). Se elaborará un plan de mantenimiento para correcciones futuras y se entregará la documentación final del producto.

Esta metodología es la más adecuada para este proyecto, ya que los requisitos están claramente definidos en el caso de estudio, el alcance está acotado y el plazo es fijo (18 semanas), lo que permite una ejecución predecible y ordenada, asegurando el logro de los objetivos propuestos.

## **7. Plan de trabajo**

| Fase | Semanas | Actividades Clave | Entregables |
| --- | --- | --- | --- |
| 1. Análisis | 1-4 | Estudio del caso. Definición detallada de requisitos funcionales y no funcionales. | Documento de Especificación de Requisitos de Software (ERS). |
| 2. Diseño | 5-7 | Diseño del modelo de base de datos (Diagrama ER). Diseño de la arquitectura de la API. Creación de mockups de todas las interfaces de usuario. | Diagrama ER, Documento de Diseño de API, Prototipos de UI/UX. |
| 3. Implementación | 8-14 | Backend (5-7): Creación de BD, desarrollo de API, lógica de negocio. Frontend (8-9): Desarrollo de interfaces web responsivas. | Base de datos funcional, API RESTful completamente operativa, Frontend conectado al backend. |
| 4. Pruebas | 15-16 | Ejecución de pruebas unitarias, de integración y de usabilidad. Depuración y corrección de errores. | Protocolo de pruebas ejecutado, informe de bugs y su resolución. Sistema estable y listo para despliegue. |
| 5. Despliegue | 17 | Despliegue de la aplicación en un entorno cloud. Configuración del servidor y la base de datos. | URL de la aplicación desplegada y funcionando. |
| Documentación | 18 | Elaboración de la documentación técnica y del usuario final. Preparación de la presentación y video demostrativo. | Documentación completa, video de demostración, repositorio Git finalizado. |

**Recursos:** Computador personal, IDE (VSCode), Stack tecnológico (Node.js, Express, React, PostgreSQL), software de diseño (Miro para mockups).

**Facilitadores:** Requisitos bien definidos, conocimiento técnico previo en el stack elegido, amplia documentación disponible en línea.

**Obstaculizadores:** La rigidez de la metodología ante cambios imprevistos. Se mitigará mediante un análisis de requisitos extremadamente detallado en las primeras fases.

## **8. Propuesta de evidencias que darán cuenta del logro de las actividades**

Las evidencias que demostrarán el logro de las actividades serán:

* Repositorio GitHub con commits regulares y documentación.
* Prototipo funcional de la plataforma web.
* Documento de diseño de BD y arquitectura.
* Reportes generados automáticamente por el sistema.
* Informe final de proyecto con análisis de resultados.

## **9. Conclusión y reflexión en inglés**

## This APT Project represents a significant opportunity to apply theoretical knowledge to the real-world problem of operational process optimization at PepsiCo Chile. The development of a centralized web platform for vehicle intake management will not only improve efficiency and traceability but also serve as a testament to the capabilities of modern software engineering. This endeavor aligns perfectly with the core competencies of my degree profile, including software development, database management, and project governance.

The choice of a traditional Waterfall methodology is a strategic decision, justified by the project's stable and well-defined requirements. This structured approach ensures a disciplined progression through sequential phases—analysis, design, implementation, testing, and deployment—providing clear milestones and reducing uncertainty. While I am aware of this methodology's inherent rigidity, this risk will be mitigated through meticulous initial planning and design. The defined objectives are clear and achievable within the academic timeframe, and the proposed plan ensures a controlled path to delivery.

This project is a valuable capstone experience that will allow me to apply core engineering principles, manage a linear project lifecycle, and deliver a robust product. Ultimately, the evidence collected throughout this process will demonstrate the successful integration of the technical and professional skills I have acquired during my program.