**Informe Previo – Proyecto APT**

**1. Descripción del Proyecto APT**

El proyecto consiste en el desarrollo de una **aplicación multiplataforma (web y móvil)** orientada a la **gestión de documentos personales y vehiculares**, con funciones básicas de organización y seguridad. La app permitirá a los usuarios **subir, almacenar, consultar y clasificar documentos** digitales (PDFs, imágenes), así como establecer **recordatorios automáticos de vencimientos** de licencias, seguros y contratos.

La **relevancia** de este proyecto radica en que muchas personas pierden control sobre documentos clave, lo que genera problemas como vencimientos olvidados, multas o pérdida de beneficios. Con esta solución se busca **ordenar la información personal en un solo lugar**, mejorando la accesibilidad y reduciendo riesgos.

En el campo laboral de la Ingeniería en Informática, este proyecto es pertinente porque exige diseñar modelos de datos escalables, aplicar buenas prácticas de desarrollo de software, integrar mecanismos de seguridad y planificar proyectos con tiempos y recursos definidos, todas competencias propias del perfil de egreso.

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

**Realizar pruebas de certificación**: se diseñarán e implementarán **pruebas de validación** (unitarias, integrales y de usabilidad) para verificar el correcto funcionamiento de las funcionalidades principales (subida de documentos, almacenamiento seguro, recordatorios).

**Gestionar proyectos informáticos**: se planificará el proyecto bajo la metodología ágil Scrum, definiendo roles, entregables y control de avances en sprints. Esto permitirá evaluar alternativas y tomar decisiones frente a obstáculos.

**Construir modelos de datos**: se diseñará un modelo relacional en PostgreSQL que permita almacenar metadatos de documentos (categoría, fecha de vencimiento, institución emisora), garantizando escalabilidad en el tiempo.

**Desarrollar una solución de software**: se implementará un sistema basado en API REST que integre los distintos componentes (backend, frontend, base de datos), asegurando mantenibilidad y cumplimiento de objetivos.

**3. Relación con intereses profesionales**

Este proyecto refleja el interés del equipo en la **construcción de soluciones tecnológicas útiles y seguras**, reforzando habilidades en desarrollo web, administración de bases de datos y metodologías de gestión de proyectos. Además, brinda experiencia práctica en la entrega de un producto real que puede ser replicado y escalado en el futuro.

**4. Factibilidad del proyecto**

**Duración:** ajustado a las 18 semanas de la asignatura, con fases definidas (diseño, desarrollo, pruebas).

**Recursos:** se utilizarán tecnologías open-source y conocidas por el equipo (FastAPI/Django REST, PostgreSQL, Angular/Ionic).

**Equipo:** compuesto por 3 integrantes con roles diferenciados: Backend/Security, Base de Datos y Frontend.

**Facilitadores:** conocimiento previo en desarrollo web y móvil, modularidad del sistema.

**Posibles dificultades:** integración entre plataformas y pruebas de seguridad → se abordarán con pruebas incrementales y control de versiones.

**5. Objetivos**

**Objetivo General**

Desarrollar una aplicación multiplataforma que permita gestionar de forma segura y organizada documentos personales y vehiculares, incorporando almacenamiento cifrado y recordatorios automáticos.

**Objetivos Específicos**

Diseñar un modelo de datos escalable en PostgreSQL.

Implementar autenticación básica y almacenamiento seguro de documentos.

Construir un CRUD de documentos accesible vía web y móvil.

Desarrollar un sistema de recordatorios automáticos para vencimientos.

Validar la solución con pruebas unitarias, integrales y de usabilidad.

**6. Metodología**

El proyecto se desarrollará bajo **Scrum**, con sprints de dos semanas y reuniones de control.

**Fase 1 (Semanas 1–4):** análisis de requerimientos, diseño de arquitectura, modelo de datos.

**Fase 2 (Semanas 5–12):** desarrollo incremental de backend, base de datos y frontend.

**Fase 3 (Semanas 13–18):** integración, pruebas de certificación y presentación final.

**Roles**

**Backend/Security:** APIs, autenticación, validación de archivos.

**Base de Datos:** diseño e implementación de modelos, consultas y pruebas de rendimiento.

**Frontend/Web-Mobile:** interfaz en Angular/Ionic, formularios para CRUD y sistema de notificaciones.

**7. Plan de trabajo (resumen)**

| **Competencia** | **Actividad** | **Recursos** | **Duración** | **Responsable** |
| --- | --- | --- | --- | --- |
| Construir modelos de datos | Diseño BD y metadatos | PostgreSQL, PGAdmin | 2 sem | BD |
| Desarrollo de software | CRUD de documentos | FastAPI/Django | 4 sem | Backend |
| Seguridad informática | Autenticación básica | FastAPI, JWT | 2 sem | Backend |
| Desarrollo frontend | Interfaz CRUD | Angular/Ionic | 4 sem | Frontend |
| QA / Pruebas de certificación | Unitarias e integrales | Pytest, Postman | 2 sem | Equipo |

**8. Evidencias**

| **Tipo** | **Evidencia** | **Descripción** | **Justificación** |
| --- | --- | --- | --- |
| Avance | Informe de requerimientos y diseño | Documento con arquitectura y BD | Demuestra definición clara |
| Avance | MVP funcional (CRUD básico) | Demo con login, subida y consulta | Evidencia avance técnico |
| Final | Aplicación Web + Móvil básica | CRUD + recordatorios | Cumple objetivo general |
| Final | Informe técnico final | Desarrollo, pruebas y conclusiones | Acredita competencias |

**9. Carta Gantt (resumen)**

**Semanas 1–2:** Definición de requerimientos.

**Semanas 3–4:** Diseño arquitectónico y BD.

**Semanas 5–8:** Desarrollo Backend (APIs, autenticación).

**Semanas 9–12:** Desarrollo Frontend (CRUD, recordatorios).

**Semanas 13–14:** Integración y pruebas funcionales.

**Semanas 15–18:** Pruebas de certificación, validación final e informe técnico.