

|  |
| --- |
| **1. Resumen avance Proyecto APT** |
| A continuación, encontrarás distintos campos que deberás completar con la información solicitada. |

|  |  |
| --- | --- |
| Resumen de avance proyecto APT | Hasta ahora el proyecto TaskControl presenta los siguientes avances:  **Análisis y planificación**: se levantaron requerimientos funcionales y no funcionales, apoyados en diagramas BPMN (carga de evidencias, programación mensual y reportes).  **Diseño de base de datos:** se elaboró un modelo entidad-relación y se creó el script SQL con las tablas tasks, users, providers y dependencies, asegurando integridad mediante llaves foráneas.  **Backend**: se implementó el módulo de login en FastAPI y se definieron endpoints iniciales para CRUD de usuarios y tareas.  **Frontend**: se desarrolló la primera versión de la interfaz en React + JavaScript, con vistas funcionales de login.  **Mockups y prototipos**: se diseñaron en Figma las pantallas principales, validadas frente a los procesos levantados.  Se han cumplido los objetivos de análisis de requerimientos, diseño de BD y construcción de login |
| Objetivos | Se mantuvieron los objetivos definidos en la Fase 1, enfocados en el desarrollo de un software gestor de tareas para organizar y dar seguimiento a las actividades de mantenimiento.  Durante esta etapa se avanzó en la implementación técnica del sistema, cumpliendo los objetivos específicos de diseño, desarrollo, validación y documentación del proyecto. |
| Metodología | Para el desarrollo del proyecto se mantuvo la metodología en cascada, utilizada desde la Fase 1, la cual organiza el trabajo en etapas secuenciales que aseguran control y coherencia en el avance. |
| Evidencias de avance | Se presentan como evidencias los repositorios en GitHub de backend y frontend, donde se refleja la implementación del login, la organización modular de endpoints y el diseño inicial de la interfaz. Además, se incluyen el modelo entidad–relación y script SQL de la base de datos, junto con carta Gantt.  Estas evidencias demuestran el cumplimiento de los objetivos específicos y la correcta aplicación de metodologías y buenas prácticas de desarrollo*.* |

Evidencias Frontend y Backend en GitHub

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Evidencia Base de Datos y Supabase

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

Modelo de datos entidad relación.

Diagrama

El contenido generado por IA puede ser incorrecto.

Arquitectura de Software

Interfaz de usuario gráfica, Diagrama

El contenido generado por IA puede ser incorrecto.

Carta Gantt actualizada

Imagen que contiene Tabla

El contenido generado por IA puede ser incorrecto.

|  |
| --- |
| **2. Monitoreo del Plan de Trabajo** |
| Examina cuidadosamente tu plan de trabajo, enfocándote especialmente en la columna de estado de avance y ajustes. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Plan de Trabajo | | | | | | | |
| Competencia o unidades de competencias | Actividades | Recursos | Duración de la actividad | Responsable | Observaciones | Estado de avance | Ajustes |
| Análisis de problemas | Casos de uso y BPMN | Bizagi/Office | 3-4 días | Michael /Vania | Se completo sin retraso | Completo | Sin ajustes. Se realizó según lo planificado. |
| Diseño de Software | Mockups y ERD | Figma/Worbench | 6 días | Michael/Vania | Validado con el equipo | Completo | Ejecutado en paralelo con análisis para optimizar tiempos. |
| Desarrollo Backend | Login + CRUD usuarios / tareas | FastAPI Python | 10 días | Luciano | Login operativo, módulo de tareas en desarrollo | En curso | Reordenado; se priorizó frontend y base de datos antes del backend. |
| Desarrollo Frontend | Login + gestión de tareas | React/JS | 10 días | Luciano/Michael | Login funcional | Completo | Desarrollado en paralelo con base de datos. Sin cambios |
| Desarrollo BD | Tablas y relaciones | PostgreSQL/ Supabase | 4 días | Michael | Se completo sin retraso | Completo | Adelantado para habilitar integración temprana. |
| Pruebas | Validación de login y CRUD | Base de datos/ Navegador | 10 días | Vania | Depende de módulos listos | No iniciado | Reprogramadas tras la integración completa del sistema. |
| Reportes | Diseño de modulo | FastApi | 8 días | Vania/  Michael | En diseño | No iniciado | Postergados hasta finalizar backend y frontend. |

|  |
| --- |
| **3. Ajustes a partir del monitoreo** |
| Profundiza en las observaciones de tu plan de trabajo. Analiza las actividades planificadas y señala qué aspectos facilitaron u obstaculizaron la ejecución del plan. Plantea cómo abordaste y/o abordarás los obstáculos. Por último, señala los ajustes que realizaste al plan de trabajo a partir de este análisis. |

|  |
| --- |
| La planificación inicial, apoyada en diagramas BPMN y casos de uso, permitió establecer con claridad los alcances del proyecto y organizar las actividades de forma eficiente. Asimismo, el uso de tecnologías previamente dominadas por el equipo, como React, FastAPI y Supabase, redujo la curva de aprendizaje y optimizó los tiempos de implementación. Gracias a estos factores, el plan de trabajo se ha ejecutado conforme a lo previsto, sin presentar dificultades. |

|  |
| --- |
| Todas las etapas planificadas se han cumplido según lo previsto, por lo que no ha sido necesario realizar ajustes ni eliminar actividades, si hemos gestionado el trabajo en paralelo de diversas tareas dentro del proyecto. Esto se debe a la adecuada planificación inicial, al uso de metodologías apropiadas y a la correcta gestión de los recursos, lo que ha permitido que el proyecto avance conforme a lo planificado. Si bien no se han realizado ajustes formales al plan, se reordenaron las prioridades para dar continuidad al desarrollo, priorizando primero el frontend de ambos módulos y la base de datos antes del backend. |

|  |
| --- |
| La actividad que presenta retraso corresponde al backend del módulo de gestión de tareas, debido a que se priorizó la construcción del frontend y la definición de la base de datos como base del sistema. Esta decisión generó un desfase en el cronograma, manteniendo este componente en desarrollo.  Como estrategia, el equipo ha decidido concentrarse en la finalización del backend, aprovechando que el frontend ya está implementado, con el fin de asegurar la integración completa. Con esta medida se espera recuperar el tiempo estimado y cumplir con los plazos generales del proyecto. |