

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

| Resumen de avance proyecto APT | *Se completa la iteración inicial (Sprint 1) con el análisis y diseño de la base de datos, implementando modelos en Django que permiten gestionar tablas con funcionalidades CRUD (agregar, editar, eliminar registros) y filtros por tablas/campos. Se desarrollan plantillas HTML para dashboards, gestión de pausas e inicio de sesión, avanzando parcialmente en el Sprint 2 (desarrollo de módulos de registro/pausas).*  *Se cumplen objetivos específicos de los siguientes requerimientos: registro de ingresos de vehículos (patentes, rutas, choferes), gestión inicial de pausas (motivos, duración) y base para perfiles de usuario (jefe de flota, mecánico, etc.). Sin embargo, falta integrar módulos en Django, implementar notificaciones, reportes automáticos y subida/consulta de documentos, así como automatización completa y trazabilidad. Se estima un avance del 10-20%, con una base sólida para continuar integrando componentes y funcionalidades según el cronograma Gantt.* |
| --- | --- |
| Objetivos | *Sin ajustes.* |
| Metodología | *Sin ajustes.* |
| Evidencias de avance | *Se presenta como evidencia el repositorio de GitHub con la aplicación avanzada, que incluye modelos de base de datos en Django, plantillas HTML para dashboards, gestión de pausas e inicio de sesión, y funcionalidades CRUD con filtros por tablas/campos. Además, se incorporan mockups de la interfaz y un diagrama de base de datos. Estas evidencias demuestran el desarrollo inicial del proyecto, reflejando la implementación de registro y gestión de ingresos de vehículos, gestión de pausas y procesos operativos, y una base para perfiles y permisos de usuarios.*  *La calidad se resguarda mediante la aplicación de metodologías ágiles (sprints), uso de Django para estructurar modelos y vistas, control de versiones en GitHub para trazabilidad, y diseño previo con mockups y diagrama de base de datos para alinear la arquitectura con los requerimientos. Esto asegura un desarrollo ordenado, escalable y alineado con las necesidades de trazabilidad, automatización y compatibilidad responsiva.*  ***Evidencias adjuntas:***   * *Enlace al repositorio de GitHub:*   + [*https://github.com/taguilera-lab/CAPSTONE004D\_AGUILERA\_LLANOS\_MORALES*](https://github.com/taguilera-lab/CAPSTONE004D_AGUILERA_LLANOS_MORALES) * *Mockups de la interfaz:*   + [*https://drive.google.com/file/d/1yEMkCNzP2g\_ZQeZs43Nz2P\_Jof8fjS1V/view?usp=drive\_link*](https://drive.google.com/file/d/1yEMkCNzP2g_ZQeZs43Nz2P_Jof8fjS1V/view?usp=drive_link) * *Diagrama de base de datos:*   + [*https://drive.google.com/file/d/1M\_N7vqcbgOGog\_0cPaGVqK3KY\_hLN2VD/view?usp=sharing*](https://drive.google.com/file/d/1M_N7vqcbgOGog_0cPaGVqK3KY_hLN2VD/view?usp=sharing) |

**2. Monitoreo del Plan de Trabajo**

| Plan de Trabajo | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Competencia o unidades de competencias | Actividades | Recursos | Duración de la actividad | Responsable[[1]](#footnote-0) | Observaciones | Estado de avance | Ajustes |
| *Gestionar proyectos informáticos (2.1, 2.2),*  *Desarrollar solución software (4.1-4.3)* | *Sprint 1: Iteración inicial* | *Documentos PDF, reuniones, herramientas como dbdiagram.io y Canva* | *1 Semana* | *Tomás Aguilera,*  *Nicolás Llanos* | *Facilita diseño con dbdiagram.io y Canva; dificulta retraso en elicitación de requerimientos, se mitiga con reuniones adicionales.* | *Completado* | *Se ajustaron los responsa-bles.*  *Tomás está a cargo del backend.*  *Nicolás está a cargo del mockup.* |
| *Gestionar proyectos informáticos (2.1, 2.2),*  *Desarrollar solución software (4.1-4.3)* | *Sprint 2: Desarrollo de módulos* | *IDE como VS Code, frameworks web (Django)* | *1 Semana* | *Todos* | *Facilita uso de Django para CRUD y plantillas HTML; dificulta falta de conocimientos integrales y retraso en elicitación, se aborda con capacitación interna y Zoom.* | *En curso* | *Se ajustaron los responsa-bles.*  *Nicolás y Michelle están a cargo de las plantillas HTML.* |
| *Desarrollar solución software (4.1-4.3)* | *Sprint 3-4: Desarrollo de reportes* | *IDE* | *2 semanas* | *Todos* | *Facilita automatización de reportes con Django; dificulta cobertura completa y retraso en elicitación, se priorizan reportes críticos (productividad, repuestos).* | *Con retraso* | *Se ajustaron los responsa-bles.* |
| *Desarrollar solución software (4.1-4.3)* | *Sprint 5-6: Integración* | *IDE, herramientas de testing.* | *2 Semanas* | *Tomás Aguilera* | *Facilita integración con Django y mockups previos; dificulta validación completa por retrasos previos, se revisa en equipo.* | *Con retraso* | *Se ajustaron los responsa-bles.* |
| *Realizar pruebas (1.1-1.3)* | *Sprint 7: Ajustes* | *IDE, herramientas de testing.* | *1 Semana* | *Tomás Aguilera* | *Facilita optimización con herramientas de testing; dificulta cobertura total, se priorizan casos críticos.* | *No iniciado* | *Se ajustaron los responsa-bles.* |
| *Realizar pruebas (1.1-1.3)* | *Sprint 8-9: Pruebas finales* | *Herramientas de testing.* | *2 semanas* | *Tomás Aguilera* | *Facilita calidad con pruebas automatizadas; dificulta cobertura total, se priorizan casos críticos y revisión grupal.* | *No iniciado* | *Se ajustaron los responsa-bles.* |
| *Todas* | *Sprint 10: Deployment* | *Hosting web, herramientas de documentación* | *1 Semana* | *Todos* | *Facilita finalización con hosting y documentación; dificulta integración final por retrasos previos, se mitiga con revisión grupal.* | *No iniciado* | *Sin ajustes.* |

**2. Ajustes a partir del monitoreo**

| **Factores que han facilitado y/o dificultado el desarrollo de mi plan de trabajo:**  ***Factores facilitadores****:*   * ***Uso de herramientas especializadas****: Se emplean Django para el desarrollo de modelos y funcionalidades CRUD, dbdiagram.io para el diseño de la base de datos y Canva para mockups, lo que agiliza la estructuración y visualización del proyecto.* * ***Metodología ágil****: La organización en sprints permite un enfoque iterativo, facilitando la planificación y seguimiento del avance, como en la finalización del Sprint 1 (análisis y diseño de base de datos).* * ***Colaboración en equipo****: La asignación clara de roles (Tomás en backend, Nicolás y Michelle en mockups/HTML) optimiza el desarrollo inicial, especialmente en plantillas y diseño.* * ***Control de versiones****: GitHub asegura trazabilidad y colaboración, permitiendo un desarrollo ordenado y recuperación de cambios.*   ***Factores dificultadores y acciones****:*   * ***Retraso en elicitación de requerimientos****: La falta de claridad o demora del cliente en definir requerimientos impacta los Sprints 1 y 2.*    + ***Acción tomada****: Se realizaron reuniones adicionales vía Teams para aclarar necesidades, ajustando responsabilidades (Tomás en backend, Nicolás en mockups).*   + ***Acción futura****: Se programarán sesiones semanales con el cliente para validar avances y evitar retrasos en Sprints 3-4 (reportes).* * ***Falta de conocimientos integrales****: La carencia de experiencia en integración de módulos en Django retrasa el Sprint 2.*    + ***Acción tomada****: Se implementó capacitación interna en Django y frameworks web.*   + ***Acción futura****: Tomás tomará cursos en Udemy para capacitarse en las tecnologías necesarias para los Sprints 3-6, enfocándose en reportes e integración.* * ***Cobertura limitada en desarrollo y pruebas****: La dificultad para abarcar todas las funcionalidades (reportes, notificaciones, documentos) retrasa los Sprints 3-4.*    + ***Acción futura****: Se priorizarán funcionalidades críticas (registro de ingresos, pausas, reportes de productividad) y se emplearán herramientas de testing automatizado en los Sprints 7-9 para garantizar calidad.*   *Se mantiene un avance del 10-20% (Sprint 1 completado, Sprint 2 en curso), con acciones correctivas que aseguren alineación con los requerimientos de registro de ingresos, gestión de pausas, perfiles de usuario y trazabilidad, mitigando retrasos para cumplir el cronograma Gantt.* |
| --- |

| **Actividades ajustadas o eliminadas:**  *Se realizaron ajustes en el plan de trabajo, específicamente en la asignación de responsables para las actividades, sin eliminar ninguna actividad. Los ajustes son:*   1. ***Sprint 1: Iteración inicial***    * ***Ajuste****: Se reasignaron responsabilidades. Tomás Aguilera se enfocó en el backend (modelos Django y base de datos), mientras Nicolás Llanos asumió el diseño de mockups.*    * ***Justificación****: El retraso en la elicitación de requerimientos por parte del cliente dificulta la planificación inicial. Dividir tareas según fortalezas (Tomás en desarrollo técnico, Nicolás en diseño visual) optimizó el avance, completando el diseño de la base de datos y mockups con herramientas como dbdiagram.io y Canva.* 2. ***Sprint 2: Desarrollo de módulos***    * ***Ajuste****: Se reasignaron responsables. Nicolás Llanos y Michelle Morales se encargaron de las plantillas HTML, mientras Tomás realiza el desarrollo de módulos en Django.*    * ***Justificación****: La falta de conocimientos integrales en Django retrasó la implementación de módulos (CRUD). Asignar a Nicolás y Michelle a las plantillas HTML permitió avanzar en la interfaz, mientras el equipo se capacita en Django para integrar módulos, asegurando progreso en un sprint crítico.*   ***Justificación de no eliminar actividades****:*  *No se eliminaron actividades porque todas (iteración inicial, desarrollo de módulos, reportes, integración, pruebas y deployment) son esenciales para cumplir los requerimientos del proyecto APT: registro de ingresos de vehículos, gestión de pausas, perfiles de usuario, notificaciones, reportes y trazabilidad. La metodología ágil (sprints) y herramientas como Django, GitHub y Discord facilitan la adaptación a retrasos, permitiendo mantener el plan original con ajustes en responsabilidades para optimizar recursos y habilidades del equipo.*  ***Facilitadores para mantener el plan****:*   * ***Metodología ágil****: La estructura en sprints permite iterar y ajustar sin eliminar tareas, manteniendo el enfoque en los objetivos.* * ***Herramientas eficientes****: Django, dbdiagram.io y Canva agilizan el desarrollo y diseño, mientras GitHub asegura trazabilidad.* * ***Colaboración estructurada****: Reuniones vía Discord y reasignación de roles según fortalezas del equipo (backend, diseño, HTML) mitigan dificultades como retrasos en elicitación o carencias técnicas, permitiendo avanzar sin reformular el cronograma.*   *Se asegura así la continuidad del plan, con un avance del 10-20% (Sprint 1 completado, Sprint 2 en curso), alineado al cronograma Gantt y los requerimientos del proyecto.* |
| --- |

| Actividades que no has iniciado o están retrasadas:   | ***Actividad (Sprint)*** | ***Estado*** | ***Motivo retraso*** | ***Acciones correctivas*** | | --- | --- | --- | --- | | Sprint 3-4: Desarrollo de reportes | Con retraso | Falta de experiencia en Django y priorización de otros módulos. | Capacitación interna y enfoque en reportes críticos. | | Sprint 5-6: Integración del sistema | Con retraso | Dificultad en validación de módulos y comunicación entre componentes. | Revisión grupal y pruebas unitarias. | | Sprint 7-9: Pruebas y ajustes | No iniciado | Depende del avance de integración y módulos. | Adelantar parte del testing y coordinar revisiones. | | Sprint 10: Deployment y documentación | No iniciado | Retrasos acumulados de fases anteriores. | Dedicación extra semanal y control del cronograma. | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

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)