**Guía2. Desarrollo Proyecto APT**

**Asignatura Capstone**

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

|  |  |
| --- | --- |
| Resumen de avance proyecto APT | Hasta la fecha, el proyecto Seguridad-SB ha avanzado de forma sostenida en su fase de desarrollo. Se han definido los roles del equipo, seleccionado la metodología ágil **Scrum**, establecido la arquitectura tecnológica y levantado los requerimientos funcionales y no funcionales.  Asimismo, se elaboraron los **mockups**, se implementó la **base de datos PostgreSQL en la nube**, se avanzó en el desarrollo de la **aplicación móvil (Ionic + Angular)** y del **sitio web (Django Templates)**, y se construyó el **Product Backlog priorizado** junto con las **historias de usuario**.  Durante esta etapa también se realizaron **ajustes en las funcionalidades** del sistema.   * En la **aplicación móvil**, se modificó la **funcionalidad del trabajador**, ajustando la lógica de registro y autenticación para permitir una mejor gestión de permisos y roles de usuario. * En la **página web**, se **agregaron nuevas funcionalidades** orientadas al control y monitoreo de las alertas, incluyendo:    + Panel de visualización con filtros dinámicos por tipo de alerta.   + Módulo para asignación de tareas a los funcionarios municipales.   + Generación automática de reportes estadísticos.   Estos ajustes surgieron tras la retroalimentación obtenida en las reuniones de revisión con la municipalidad, priorizando la usabilidad y la eficiencia del sistema. |
| Objetivos | Diseñar e implementar un sistema digital integral para el monitoreo y gestión de los Vehículos de Seguridad Ciudadana de la Municipalidad de San Bernardo, que permita optimizar la respuesta ante emergencias mediante el uso de tecnologías de geolocalización y comunicación en tiempo real.  **Objetivos específicos:**   * **Implementar un módulo de geolocalización avanzada** que permita la visualización en tiempo real de los vehículos municipales, incorporando la api de google para que permita la precisión de los datos para los reportes. * **Desarrollar un sistema de alertas bidireccional**, capaz de gestionar la recepción y envío de notificaciones entre los ciudadanos y los vehículos de seguridad, clasificando los eventos por nivel de prioridad y tipo de incidente, para mejorar la coordinación de respuestas. * **Diseñar e implementar una base de datos relacional en PostgreSQL**, estructurada para almacenar información sobre usuarios, vehículos, rutas, alertas, reportes y mantenimientos, asegurando la integridad y la escalabilidad del sistema. * **Construir una API REST utilizando Django REST Framework**, que actúe como capa de comunicación entre la base de datos, la plataforma web y las aplicaciones móviles, garantizando la seguridad y eficiencia en la transferencia de datos. * **Desarrollar una aplicación móvil híbrida mediante Ionic y Angular**, destinada tanto a los funcionarios de seguridad como a los ciudadanos, que permita generar, recibir y gestionar alertas con soporte para notificaciones push y validación de roles de usuario para los trabajadores. * **Crear una plataforma web con Django** orientada a los operadores municipales, que facilite el monitoreo centralizado de incidentes, la administración de vehículos, la asignación de tareas y la generación de reportes automáticos. * **Diseñar un módulo de análisis y reportes estadísticos**, utilizando visualizaciones dinámicas (gráficos y mapas de calor) que permitan identificar patrones de comportamiento, zonas de riesgo y tiempos promedio de respuesta. * **Ejecutar pruebas de control de calidad**, incluyendo pruebas unitarias, funcionales, de integración y de rendimiento, aplicando buenas prácticas de la industria para garantizar la estabilidad y confiabilidad del sistema en entornos reales. * **Desarrollar la documentación técnica completa del sistema**, que incluya manuales de usuario, diagramas de arquitectura, historias de usuario, procedimientos de despliegue y lineamientos para el mantenimiento futuro de la plataforma. |
| Metodología | Para este proyecto estamos utilizando la metodología ágil **Scrum**, seleccionada por su enfoque iterativo, adaptable y centrado en entregas de valor continuo (Schwaber & Sutherland, 2020). Esta metodología nos permite dividir el trabajo en **sprints** de corta duración, específicamente de **una semana**, lo que facilita la validación temprana de los avances con el cliente e incorporar retroalimentación en cada iteración. Estas reuniones de revisión se realizan directamente con la **municipalidad**, permitiendo una mejora constante del producto.  El proceso comenzó con el **levantamiento de requerimientos**, donde se definieron las funcionalidades esenciales del sistema. Posteriormente, se elaboraron **prototipos iniciales** que nos permitieron validar tanto el diseño como la experiencia de usuario.  Para el desarrollo del proyecto se están utilizando diversas herramientas, tanto para el desarrollo del sistema como para la organización de las tareas. Algunas de estas herramientas son:   * **Trello**: Utilizado para la organización de tareas (pendientes, en proceso, finalizadas o en revisión). Esta herramienta nos permite gestionar eficientemente tanto el desarrollo del sistema como la documentación del proyecto. * **PostgreSQL**: Base de datos empleada para almacenar la información necesaria de la aplicación móvil y del sitio web. Está alojada en la nube para su uso eficiente y seguro. * **Ionic con Angular**: Framework utilizado para el desarrollo de la aplicación móvil, la cual permitirá a los ciudadanos generar alertas desde sus dispositivos y que los vehículos transmitan su ubicación en tiempo real. * **Django Templates**: Se utilizan para la construcción de las vistas del sistema web, facilitando la integración entre la lógica del servidor y la interfaz destinada a los operadores de la municipalidad. * **Android Studio**: Herramienta utilizada para compilar y probar el código fuente de la aplicación móvil, permitiendo su ejecución en dispositivos reales. * **Django con Django Rest Framework**: Se emplea para el desarrollo del **Back-end** y la creación de una API REST robusta y escalable, que gestiona los procesos centrales del sistema: usuarios, vehículos, alertas y registros.   Además, se definieron los siguientes roles dentro del equipo de trabajo:   * **Product Owner**: José Villablanca * **Scrum Máster**: José Ignacio Acevedo Zamorano (responsable de la documentación) * **Team Developers**:   + Benjamín Alejandro Cerón de la Fuente (responsable de la Página Web)   + Sebastián Félix Reveco Fuentes (responsable de la Aplicación Móvil)   En la etapa final del proyecto se realizarán **pruebas funcionales y de integración**, con el objetivo de asegurar la calidad y confiabilidad del sistema. Estas pruebas incluirán validaciones específicas sobre la **geolocalización en tiempo real**, la **emisión automática de alertas** y la **generación de reportes estadísticos**.  El uso combinado de estas tecnologías permite alcanzar los objetivos establecidos, garantizando un sistema **escalable, seguro y accesible** para los usuarios finales. Asimismo, la aplicación de Scrum asegura una entrega **incremental y adaptable** a los requerimientos cambiantes, validada de forma continua con el cliente.  El uso combinado de estas tecnologías permite cumplir con los objetivos planteados, garantizando un sistema escalable, seguro y accesible para los usuarios finales. Además, la aplicación de Scrum asegurará una entrega incremental, adaptable a los requerimientos cambiantes y validada de manera constante con el cliente. |
| Evidencias de avance | Las evidencias presentadas demuestran el progreso técnico y organizativo del proyecto durante la Fase 2. Estas evidencias reflejan el cumplimiento de los objetivos y la correcta aplicación de la metodología ágil Scrum.  **Evidencias entregadas:**   * Mockups aprobados por el cliente (validación de diseño y usabilidad). * Scripts de la Base de datos utilizada en PostgreSQL con sus relaciones y llaves foráneas implementadas (Todavía en fase de pruebas).      * Aplicación móvil en Ionic + Angular con autenticación funcional y geolocalización activa        * Manejo del equipo por trello para el desarrollo de la actividades. * Documentación relacionada al proyecto manejada por drive.     **Justificación:**  Estas evidencias permiten verificar que el equipo ha avanzado de manera coherente con el plan establecido, aplicando buenas prácticas de desarrollo y control de calidad. El uso de metodologías ágiles, la documentación de cada sprint y la validación continua con el cliente aseguran el cumplimiento de los estándares de la disciplina. |

|  |
| --- |
| **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 | Actividades | Recursos | Duración | Responsable[[1]](#footnote-1) | Observaciones | Estado de avance | Ajustes |
| Análisis y gestión de requerimientos | **Toma de requerimientos** | Entrevistas con cliente, Reuniones, Trello | 3 | José Acevedo y Benjamín Cerón | Se levantaron requerimientos base con la municipalidad. | Completado | — |
| Análisis y gestión de requerimientos | **Análisis de requerimientos** | Documentación | 3 | Equipo completo | Se priorizaron requerimientos y se definieron funcionalidades principales. | Completado | — |
| Diseño de interfaz y experiencia de usuario | **Diseño de mockups** | Figma, Reuniones de revisión | 4 | Sebastián Reveco y Benjamin Ceron | Mockups aprobados por el cliente. | Completado | — |
| Análisis funcional | **Desarrollo de historias de usuario** | Trello, Documento de historias | 2 | José Ignacio Acevedo | Historias validadas y priorizadas. | Completado | — |
| Planificación ágil | **Elaboración del Product Backlog** | Trello | 2 | Equipo completo | Backlog priorizado y validado. | Completado | — |
| Diseño técnico | **Arquitectura del sistema** | Diagramas, Herramientas de modelado | 3 | Equipo completo | Arquitectura definida. | Completado | — |
| Diseño técnico | **Selección de base de datos** | Análisis comparativo, PostgreSQL | 2 | Equipo completo | Se seleccionó PostgreSQL en la nube. | Completado | — |
| Desarrollo frontend web | **Levantamiento de página web** | Django Templates, HTML/CSS | 5 | Benjamín Cerón | Estructura inicial lista. | En curso | — |
| Desarrollo frontend móvil | **Levantamiento de aplicación móvil** | Ionic + Angular | 5 | Sebastián Reveco | Se configuró la estructura base. | En curso | — |
| Desarrollo backend | **Levantamiento de base de datos** | PostgreSQL | 4 | Benjamin Ceron | Tablas iniciales creadas. | Completado | — |
| Desarrollo backend | **Desarrollo de base de datos** | PostgreSQL, SQL Server | 6 | José Ignacio Acevedo | Relaciones y llaves foráneas implementadas. | En curso | Analizando si conectar con BD directo o con una api. |
| Desarrollo web | **Desarrollo de página web** | Django + DRF | 10 | Benjamín Cerón | Se agregaron módulos de control y gestión de alertas. | En curso | Ajuste por nuevas funcionalidades |
| Desarrollo móvil | **Desarrollo de aplicación móvil** | Ionic, Android Studio | 10 | Sebastián Reveco | Flujo de usuario actualizado con roles. | En curso | Ajuste de funcionalidad de trabajador |
| Integración de sistemas | **Integración base de datos y página web** | Django ORM, API REST | 5 | José Ignacio Acevedo | Integración funcional. | En curso | — |
| Integración de sistemas | **Integración aplicación móvil y página web** | API REST, Postman | 5 | Sebastián Reveco | Conexión establecida parcialmente. | En curso | Ajuste por errores de conexión |
| Control de calidad | **Pruebas de calidad** | Postman, Android Studio | 7 | Equipo completo | Pendiente tras la integración final. | No iniciado | — |
| Documentación técnica | **Documentación final** | Word, GitHub | 5 | José Ignacio Acevedo | En redacción. | En curso | Se añadieron nuevas secciones y capturas. |

|  |
| --- |
| **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 abordar los obstáculos. Por último, señala los ajustes que realizaste al plan de trabajo a partir de este análisis. |

|  |
| --- |
| **Factores que han facilitado y/o dificultado el desarrollo de mi plan de trabajo**  Durante el proyecto hemos realizado un seguimiento constante al plan de trabajo establecido desde el inicio, lo que nos permitió identificar los avances y también algunos aspectos que fue necesario ajustar. Entre los factores que facilitaron el desarrollo, destacan las reuniones semanales con la Municipalidad de San Bernardo, las cuales han sido fundamentales para aclarar dudas técnicas y validar los avances del sistema. Además, el uso de herramientas de desarrollo y gestión como Trello, GitHub y Visual Studio Code. Cuales han permitido mantener una organización eficiente del trabajo, el control de versiones y la colaboración entre los integrantes. También ha sido beneficioso que los lenguajes y tecnologías utilizados sean conocidos por el equipo, lo que agiliza las tareas de programación y pruebas.  Por otro lado, las principales dificultades se relacionaron con la incorporación de nuevas funcionalidades solicitadas por el cliente, lo que implicó reestructurar parte del desarrollo. También enfrentamos desafíos en la organización del equipo debido a las diferencias en los horarios laborales de los integrantes, y en la elaboración de la documentación técnica, que ha requerido más tiempo del esperado |

|  |
| --- |
| **Actividades ajustadas o eliminadas**  Durante el desarrollo del proyecto fue necesario realizar algunos ajustes importantes en el plan de trabajo. El cambio más relevante fue la eliminación de una de las aplicaciones móviles que inicialmente se habían planificado: una destinada a los funcionarios de seguridad y otra para los ciudadanos. Tras analizar la arquitectura y las necesidades del cliente, se decidió unificar ambas en una sola aplicación móvil que integra las funcionalidades para los dos tipos de usuario, permitiendo un mantenimiento más simple y una experiencia de uso más factible.  Además, se realizaron ajustes en la integración entre la aplicación móvil y la plataforma web, debido a dificultades en la conexión y autenticación. Para resolver estos inconvenientes, se realizaron nuevas pruebas con Postman y se aplicaron configuraciones adicionales en la API. También se amplió el tiempo destinado al desarrollo de la página web, ya que se agregaron nuevas funcionalidades solicitadas por la municipalidad, como la asignación de tareas y la generación automática de reportes.  Estos cambios permitieron optimizar el desarrollo, mejorar la eficiencia del sistema y mantener la coherencia del proyecto con los requerimientos establecidos por el cliente. |

|  |
| --- |
| **Actividades que no has iniciado o están retrasadas**  Actualmente, algunas actividades del proyecto presentan un leve retraso respecto a la planificación inicial. Entre ellas se encuentran las pruebas de calidad e integración, las cuales se programaron para realizarse una vez finalizada la conexión completa entre la aplicación móvil y la plataforma web. Debido a los ajustes realizados en la API.  Asimismo, la documentación técnica final se encuentra en desarrollo y otra que se está realizando para la validación del proyecto, ya que se ha priorizado la implementación de nuevas funcionalidades y la resolución de errores detectados durante las pruebas . Para evitar un mayor retraso, el equipo acordó destinar sesiones adicionales de trabajo y utilizar GitHub como repositorio central para registrar avances mientras que la documentación se va actualizando a través de un almacenamiento en la nube compartido que sería google drive. |

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-1)