**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 | Hemos completado con éxito la fase de Análisis de Requisitos y la fase de Diseño Funcional y Arquitectónico del proyecto "Sistema de Monitoreo de Niveles de Agua en Pozos". El objetivo principal de esta fase era sentar las bases sólidas para el desarrollo del sistema. Se han cumplido los siguientes objetivos específicos: Diseñar la estructura funcional (Épicas y Historias de Usuario), y Definir la lógica de interacción de datos (Matriz CRUD y Diagramas de Flujo para procesos clave). Actualmente, el proyecto se encuentra en la etapa inicial de Desarrollo de la API RESTful. |
| Objetivos | No se requieren ajustes mayores a los objetivos iniciales. El objetivo principal de desarrollar una aplicación web para el monitoreo en tiempo real de niveles de agua se mantiene. |
| Metodología | Mantenemos la metodología Ágil (Scrum/Kanban) con un enfoque iterativo. Se refuerza el uso de GitHub para el control de versiones y trazabilidad de las tareas y Trello para la gestión del backlog y el sprint. |
| Evidencias de avance | La evidencia presentada incluye la documentación completa del diseño funcional: Épicas, Historias de Usuario, Matriz CRUD y Diagramas de Flujo. Esta evidencia da cuenta del desarrollo de las siguientes fases: Definición del Alcance y Diseño de Interacciones. Específicamente, los Diagramas de Flujo y la Matriz CRUD demuestran cómo la lógica del sistema gestionará las Alertas Críticas y los Reportes. Para resguardar la calidad, se aplicó la correcta metodología de Ingeniería de Requisitos, asegurando que las Historias de Usuario contengan Criterios de Aceptación claros para las pruebas futuras. |

|  |
| --- |
| **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[[1]](#footnote-1) | Observaciones | Estado de avance | Ajustes |
| Fase 1: Análisis y Requisitos |  |  |  |  |  |  |  |
|  | 1.1 Definición de Alcance, Objetivos, y Análisis de Actores. | Documentos de la Guía, Reuniones de Equipo. | 1 Semana | Felipe González | No hubo dificultades, el caso de negocio fue claro. | Completado | Ninguno. |
|  | 1.2 Creación de Épicas, Historias de Usuario y Priorización del Backlog. | Plantillas Ágiles, Product Backlog. | 1 Semana | Martin Nenen | Las Historias de Usuario y sus Criterios de Aceptación fueron bien definidos. | Completado | Ninguno. |
| Fase 2: Diseño Funcional y Arquitectónico |  |  |  |  |  |  |  |
|  | 2.1 Diseño de Flujos de Proceso (Diagramas) y Matriz CRUD. | Herramientas de Modelado (Mermaid), Matriz CRUD. | 1 Semana | Felipe González | Se logró definir la lógica de Alerta Crítica de manera detallada. | Completado | Ninguno. |
|  | 2.2 Diseño de Mockups de Interfaz y Prototipado. | Herramienta de Diseño UI/UX. | 1 Semana | Martin Nenen | El diseño es simple y enfocado en la usabilidad del monitoreo. | Completado | Ninguno. |
| Fase 3: Desarrollo de la Plataforma |  |  |  |  |  |  |  |
|  | 3.1 Implementación de la API RESTful (Recepción de Datos). | Python, Base de Datos, GitHub. | 2 Semana | Maximiliano Hormaeche | Tarea crítica. El equipo comenzó con la configuración del entorno. | En curso | Ajustada (Ver sección 3). |
|  | 3.2 Desarrollo del Módulo de Autentificación y Gestión de Usuarios. | Python (Framework Web), Base de Datos. | 1 Semana | Maximiliano Hormaeche | Depende directamente de la API. Requiere roles claros (Admin, Operador). | En curso | Ninguno. |
|  | 3.3 Desarrollo del Dashboard de Visualización (Gráficos). | HTML/CSS, Librería de Gráficos (JS/Python). | 2 Semana | Felipe González | Requiere que la API de datos (3.1) esté estable. | No iniciado | Ninguno. |

|  |
| --- |
| **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. |

|  |
| --- |
|  |

|  |
| --- |
| Actividades ajustadas o eliminadas: *Señalar los ajustes que realizaste a tu plan de trabajo o actividades que eliminaste y, justifica por qué lo hiciste.*  *En el caso de que tu plan de trabajo no haya requerido ni requiera ajustes, justifica esta decisión a partir de los facilitadores que te han permitido desarrollarlo como fue planeado.*   * **Actividad Ajustada:** La duración de la actividad **3.1 Implementación de la API RESTful (Recepción de Datos)** se ajusta de 2 semanas a **2.5 semanas**. * **Justificación:** El análisis de la **Épica de Recepción y Procesamiento de Datos de Sensores** reveló que la validación y el procesamiento inicial de los datos (antes de la evaluación del umbral) son más complejos de lo inicialmente previsto. Se necesita más tiempo para garantizar la robustez de este componente crítico. |

|  |
| --- |
| Actividades que no has iniciado o están retrasadas: *En caso de que* ***no hayas iniciado actividades o estén retrasadas*** *de acuerdo a tu planificación, señala los motivos por los que no has podido cumplir dichos plazos y qué estrategias utilizarás para avanzar en dichas actividades y no afectar tu proyecto APT.*   * **Actividad Retrasada:** **3.3 Desarrollo del Dashboards de Visualización (Gráficos)** (Estado: No iniciado). * **Motivos del Retraso:** Esta actividad es totalmente dependiente de la finalización de la API de datos (3.1). No es posible iniciar la conexión a los gráficos sin una fuente de datos funcional. * **Estrategias para avanzar:**  1. **Paralelización:** El responsable comenzará la implementación de la interfaz de usuario estática y los contenedores de los gráficos (sin datos en tiempo real) utilizando los Mockups como referencia. 2. **Minimizar Alcance Inicial:** Priorizar en el sprint de desarrollo solo la visualización en tiempo real (HU 001-HU) y dejar la visualización histórica para una iteración posterior si el tiempo es limitado. |

1. [↑](#footnote-ref-1)