**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 el momento de rellenar este documento, hemos completado de manera satisfactoria los tres primeros sprints del proyecto, correspondiente al desarrollo inicial de la plataforma de gestión de agresiones en establecimientos públicos de salud. Este avance representa un cumplimiento aproximado del 40% de los objetivos propuestos, centrados en la configuración del sistema y en la implementación de funcionalidades clave de gestión de usuarios y reportes de incidentes.  Actividades realizadas:   1. Configuración inicial y arquitectura del proyecto    * Durante este sprint, se estableció la estructura base del sistema. Configuramos Next.js en el front-end y .NET Core 8 para el back-end en una arquitectura moderna modular y escalable. (Arquitectura Limpia)    * Objetivos específicos logrados:      + Diseño e implementación de los usuarios con roles y datos de los establecimientos      + Configuración de la autenticación y estructura de carpetas en el sistema para asegurar una arquitectura sólida. 2. Gestión de usuarios y establecimientos    * En el segundo sprint, centramos los esfuerzos en la implementación de la funcionalidad de gestión de usuarios y en la asociación de estos a los establecimientos.    * Objetivos específicos logrados:      + Implementación de la API de creación de usuarios con roles jefatura y personal asociado      + Desarrollo de la interfaz en Next.js para la gestión de usuarios y establecimiento, asegurando un flujo claro y sencillo para el administrador. 3. Registros de reportes de incidentes de agresión    * En este sprint, se introdujeron las funcionalidades que permiten a la jefatura crear reportes de agresión y asignar un afectado.    * Objetivos específicos cumplidos:      + Desarrollar la API de registro de agresión      + Realización de pruebas unitarias y validaciones de estos módulos. |
| Objetivos | No se han realizado cambios en los objetivos generales, dado que el plan de trabajo ha resultado adecuado hasta la fecha. Sin embargo, con el progreso del proyecto, hemos identificado la necesidad de documentación adicional. |
| Metodología | No se han realizado cambios en la metodología de desarrollo basada en Scrum. |
| Evidencias de avance | Objetivos específicos:   1. Configurar la arquitectura del sistema para soportar el manejo de usuarios, roles y registros de agresiones en un entorno seguro y escalable. 2. Desarrollar funcionalidades de gestión de usuarios, estableciendo el nivel de acceso y control por roles. 3. Implementar la creación, notificación y validación de reportes de agresión en tiempo real. |

|  |
| --- |
| **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 |
| Dominio de tecnologías de la información y comunicación (TIC) | Configurar arquitectura del sistema en Azure y crear base de datos en Azure SQL. | Azure, Azure SQL, .NET Core 8 y Next.js | 2 Semanas | Valeria | La conexión a Azure fue sencilla y estable; se necesitó configurar permisos para Azure SQL | Completado | No se requirieron ajustes |
| Desarrollo y mantenimiento de software | Desarrollar interfaz de usuario para la gestión de establecimientos | Next.js, React y VSCode | 2 Semanas | Fernanda | Diseño intuitivo; algunas validaciones en el formulario requirieron ajustes | Completado | Se ajustaron validaciones para prevenir datos duplicados en la creación de establecimientos |
| Gestión de proyectos informáticos | Desarrollar API para agregar establecimientos y asociar usuarios | .NET Core, C#, Azure DevOps | 2 semanas | Juan | API desarrollada con éxito; pruebas de conexión y acceso satisfactorias. | Completado | No se requirieron ajustes. |
| Seguridad informática | Configurar autenticación y roles de acceso | Azure AD, JWT, .NET Core, documentación | 2 semanas | Valeria | La configuración de roles de acceso fue compleja; se implementaron pruebas adicionales. | Completado | Se agregó documentación extra para roles y accesos de seguridad. |
| Trabajo en equipo y comunicación | Configurar envío de correos de notificación | Servicios de correo electrónico (SMTP) | 1 semana | Juan | La configuración del servidor SMTP tiene retrasos. | Retrasado | No se realizaron ajustes |
| Análisis y diseño de sistemas | Implementar API para la gestión de usuarios y asociación con establecimientos | .NET Core, SQL Server | 2 semanas | Juan | La estructura de la base de datos fue adecuada; se hicieron pequeños ajustes en el esquema. | Completado | Se ajustaron los esquemas de tablas para mejorar la relación entre establecimientos y usuarios. |
| Desarrollo y mantenimiento de software | Implementar funcionalidad para crear reportes de agresión desde jefatura | .NET Core, SQL Server, Next.js | 2 semanas | Juan - Fernanda | Requiere validación de campos; problemas menores con el manejo de fechas. | En curso | Validación adicional para prevenir errores en el ingreso de fechas incorrectas. |
| Adaptabilidad y aprendizaje continúo | Realizar pruebas de funcionalidad y usabilidad | Usuarios de prueba, QA tools | 1 semanas | Valeria | Las pruebas de funcionalidad son satisfactorias; se necesitan ajustes menores. | En curso | Ajustes menores en la interfaz para mejorar la usabilidad. |
| Análisis y diseño de sistemas | Diseñar y desarrollar dashboard con gráficos y mapa georreferenciado | React Leaflet, librerías de gráficos (Chart.js) | 2 Semanas | Fernanda | Integración de mapa requiere ajustes para una mejor visualización de datos. | No iniciado | Se plantearán ajustes en la visualización según feedback de pruebas |
| Trabajo en equipo y comunicación | Revisar feedback del equipo y realizar retrospectivas | Herramientas de feedback | Al final de cada sprint | Equipo | Las reuniones de retrospectiva han sido útiles para mejorar la comunicación en el equipo. | En curso | No se han realizado ajustes específicos hasta el momento, pero se implementaron mejoras en la planificación de tareas |

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

|  |
| --- |
| Factores que han facilitado y/o dificultado el desarrollo de mi plan de trabajo: *Describe los factores que han facilitado y/o dificultado el desarrollo de tu Proyecto APT hasta ahora. En el caso de las dificultades debes describir qué acciones tomaste y/o tomarás para solucionarlas.*   * **Dominio de herramientas y tecnologías:** La experiencia que tenemos como equipo en tecnologías como **.NET Core**, **Next.js** y la plataforma de **Azure** ha facilitado la implementación de las primeras fases del proyecto. La experiencia con estos entornos ha permitido que la configuración inicial de la arquitectura y la base de datos se completaran sin inconvenientes. * **Colaboración eficiente y comunicación en el equipo:** Las reuniones de **retroalimentación y retrospectivas** al final de cada sprint han sido clave para identificar problemas a tiempo y optimizar el flujo de trabajo. La comunicación constante entre nosotros como integrantes (usando herramientas de trabajo colaborativo) ha permitido que se resuelvan dudas y se eviten posibles bloqueos en el desarrollo. * **Metodología Ágil (Scrum):** La división del proyecto en sprints de dos semanas nos ha ayudado a enfocar los esfuerzos en entregas específicas y a evaluar el progreso de forma regular. Esto nos ha permitido tener una visión clara de los objetivos y ajustes en cada sprint, manteniendo el proyecto en una dirección controlada. |

|  |
| --- |
| 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.*   * **Problemas de configuración de correos de notificación**: Durante la configuración del sistema de notificaciones por correo, enfrentamos problemas con la autenticación del servidor SMTP inicial. Esto provocó retrasos en el envío de notificaciones a los usuarios, ya que el sistema no podía completar la autenticación necesaria para el envío automático de correos.   + **Acción tomada**: Para superar este obstáculo, optamos por cambiar la prioridad y esto es algo que se desarrollará al final del desarrollo, ya que, no afecta a la funcionalidad principal y es solo una especie de alerta para notificar que el usuario tiene un registro de agresión para revisar. Pero el usuario ingresando a la plataforma tendrá el registro visible. * **Validaciones y pruebas de interfaz**: Algunos formularios de la interfaz de usuario requerían validaciones adicionales para prevenir errores (por ejemplo, evitar datos duplicados en la creación de establecimientos y validar fechas en los reportes de agresión). Esto aumentó ligeramente el tiempo de desarrollo.   + **Acción tomada**: Implementamos una ronda adicional de pruebas de calidad y ajustes de validación para asegurar la integridad de los datos ingresados. Se aplicaron validaciones tanto en front-end como en back-end para prevenir errores en tiempo real y garantizar una mejor experiencia de usuario. |

|  |
| --- |
| 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.*   * **Actividades retrasadas:**   + Implementación de la Funcionalidad de Notificaciones por Correo Electrónico: La implementación de esta funcionalidad se retrasó debido a los problemas de autenticación mencionados en el servicio de correo SMTP. * **Motivos del retraso:**   + Los problemas técnicos de autenticación en el servicio de correo SMTP fueron el principal motivo del retraso en esta actividad. Estos problemas no se habían anticipado en la planificación inicial, ya que se esperaba que la configuración del servicio de correo fuera más directa. * **Estrategias para avanzar y no afectar el proyecto:**   + Cambio en la prioridad de tareas: El cambio de prioridad se realizó de manera ágil, permitiendo que la funcionalidad de notificaciones avanzara se desplazará sin comprometer otras tareas.   + Redistribución de Tareas para Optimizar Tiempo: Mientras se resolvían los problemas del servicio de correo, los integrantes del equipo avanzaron en otras tareas en paralelo, como la implementación de la API de reportes de agresión y la configuración de roles de acceso. Esta estrategia permitió que el impacto en el cronograma fuera mínimo. |

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)