**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 | *Durante esta segunda entrega del proyecto SmartFlow – Sistema de Gestión y Reservas, se avanzó desde la fase de documentación y levantamiento de requerimientos hacia el diseño y desarrollo inicial de la aplicación web.*  *En este periodo se completaron los casos de uso extendidos, el documento ERS (Especificación de Requerimientos de Software) bajo estándar IEEE 830, y se elaboraron los mockups principales de las interfaces del sistema. Estos mockups muestran tanto el entorno del Administrador —con sus módulos de gestión de usuarios, servicios, solicitudes y reportes— como el del Usuario, que incluye su calendario de reservas, notificaciones y chat interno.*  *A nivel técnico, se consolidó la arquitectura en capas del sistema, implementada con .NET Core 8, Razor Pages y SQL Server, definiendo además las entidades principales (Usuario, Reserva, Servicio, Notificación, Solicitud, Carrera, entre otras)* |
| Objetivos | *Definir e implementar el modelo de datos completo bajo Entity Framework Core.*  *Diseñar la arquitectura del sistema basada en capas: presentación, negocio y datos.*  *Elaborar los mockups del entorno visual del administrador y del usuario final.*  *Documentar los casos de uso extendidos para garantizar coherencia entre la lógica de negocio y la interfaz.*  *Validar la conexión a la base de datos y la estructura inicial del proyecto en Visual Studio.* |
| Metodología | *El proyecto continúa aplicando la metodología ágil Scrum, organizando el trabajo en sprints de tres semanas que permiten planificar, desarrollar y validar avances de forma iterativa.*  *En esta entrega se completaron los sprints 2 y 3, centrados en el levantamiento de requerimientos, diseño de casos de uso y creación de mockups, además de la definición técnica del sistema.*  *Cada sprint se documentó con su respectivo informe de avance, donde se establecieron los objetivos, tareas realizadas y próximos pasos.*  *Las reuniones con el equipo y el cliente académico (Duoc UC) se mantienen de manera periódica para validar entregables y asegurar que el proyecto cumpla con los requerimientos definidos en el documento ERS.*  *La aplicación de esta metodología ha permitido mantener una visión clara y organizada del progreso, con tareas distribuidas y controladas, minimizando riesgos y asegurando que el desarrollo avance conforme al cronograma planificado.* |
| Evidencias de avance | *Documento ERS (Especificación de Requerimientos de Software), elaborado bajo el estándar IEEE 830, donde se detallan los requisitos funcionales y no funcionales del sistema, los tipos de usuarios, la planificación del proyecto y los costos estimados.*  *Casos de Uso Extendidos, que describen el comportamiento detallado de cada interacción dentro del sistema, asegurando la coherencia entre los requerimientos y las funcionalidades planificadas.*  *Mockups de interfaz, que muestran el diseño visual de las pantallas principales del administrador y del usuario, evidenciando la aplicación de principios de usabilidad y diseño responsivo.*  *Modelo de datos implementado en código, con las clases principales (Usuario, Rol, Reserva, Servicio, Solicitud, Notificación, Carrera, Arancel, entre otras) integradas en el contexto SmartFlowContext mediante Entity Framework Core.*  *Configuración de la base de datos y conexión funcional mediante appsettings.json, validando la correcta comunicación entre la aplicación y el servidor SQL Server.*  *Informe de avance de sprint 3, donde se registran las tareas realizadas, los resultados obtenidos y los próximos pasos del desarrollo.* |
| **2. Monitoreo del Plan de Trabajo** |
| Examina cuidadosamente tu plan de trabajo, enfocándote especialmente en la columna de estado de avance y ajustes. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Competencia o unidades de competencias** | **Actividades** | **Recursos** | **Duración de la actividad** | **Responsable** | **Observaciones** | **Estado de avance** | **Ajustes** |
| Analiza y modela requerimientos de software. | Elaboración del documento ERS según estándar IEEE 830. | Visual Studio, Word, lineamientos DUOC, repositorio GitHub. | 1 semana | Juan Castillo / Mallely Calfilaf | Se completó con validación de requisitos funcionales y no funcionales. | Completado | Se ajustaron descripciones de algunos requisitos según retroalimentación docente. |
| Aplica metodologías de ingeniería de software. | Diseño de Casos de Uso extendidos y flujos alternativos. | Draw.io, Word, documento ERS. | 1 semana | Juan Castillo | Se generaron 12 casos de uso principales y 1 extendido completo. | Completado | Se reordenó la numeración y se mejoraron las descripciones para mayor claridad. |
| Desarrolla prototipos funcionales de software. | Diseño de mockups principales (Administrador y Usuario). | Canva, guía UI/UX institucional. | 1 semana | Mallely Calfilaf | Se diseñaron vistas responsivas aplicando principios de usabilidad. | Completado | Se actualizó la paleta de colores y orden del menú principal. |
| Diseña y modela bases de datos relacionales. | Creación del modelo de datos (clases C# y contexto EF Core). | Visual Studio, SQL Server, GitHub. | 2 semanas | Juan Castillo | Se implementaron entidades: Usuario, Rol, Reserva, Servicio, Notificación, etc. | Completado | Se renombraron propiedades para mantener coherencia con el diagrama ERS. |
| Implementa arquitecturas de desarrollo web. | Configuración del proyecto en Visual Studio y conexión con base de datos (appsettings.json). | Visual Studio, SQL Server, EF Core. | 1 semana | Juan Castillo | La conexión fue validada correctamente; base de datos funcional. | Completado | Se corrigió el string de conexión por error de ruta local. |
| Aplica buenas prácticas de control de versiones. | Creación y sincronización del repositorio GitHub. | GitHub Desktop, comandos Git, repositorio CodeFixService/Capstone. | Continuo | Juan Castillo / Mallely Calfilaf | Se subieron los modelos y controladores iniciales; versionado activo. | Completado | Se agregó .gitignore para excluir archivos temporales de compilación. |
| Aplica metodologías ágiles en proyectos de software. | Desarrollo y documentación de los sprints 2 y 3. | Trello, Word, plantillas Scrum. | 3 semanas | Equipo SmartFlow | Sprints documentados con tareas, avances y próximos hitos. | Completado | Se integraron observaciones del sprint anterior y se ajustó la planificación del siguiente. |
| Diseña soluciones seguras y escalables. | Definición de arquitectura en capas y medidas de seguridad (cifrado, roles, control de sesiones). | .NET Core 8, Razor Pages, Entity Framework. | 2 semanas | Juan Castillo | Arquitectura establecida; se documentó en el código base y ERS. | En curso | Se implementará cifrado de contraseñas y validaciones de sesión en el siguiente sprint. |
| Aplica pruebas y validaciones en el desarrollo. | Revisión del funcionamiento del modelo y validación de conexión BD. | Visual Studio, SQL Server Management Studio. | 1 semana | Juan Castillo | Se probaron entidades y relaciones principales, sin errores críticos. | Completado | Se agregaron validaciones para relaciones uno a muchos (Carrera–Usuario). |
| Evalúa la calidad del desarrollo de software. | Preparación del entorno para inicio de pruebas unitarias y de integración. | Visual Studio, NUnit, Trello (control de incidencias). | 1 semana | Equipo SmartFlow | Entorno preparado para la próxima fase de desarrollo. | En curso | Se planifica ejecución de pruebas en la siguiente entrega. |

|  |
| --- |
| **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:  *Entre los factores que han facilitado el avance, destaca la organización del equipo y el uso de herramientas de gestión como Trello y GitHub, que han permitido mantener un control claro de las tareas, versiones y fechas de entrega.*  *La distribución por sprints también ha sido un elemento clave, ya que permitió trabajar con objetivos específicos en cada fase, garantizando entregas parciales revisadas y aprobadas antes de pasar a la siguiente etapa.*  *Además, la claridad de los requerimientos definidos en el documento ERS y la aplicación de una arquitectura en capas bien estructurada facilitaron la implementación técnica y redujeron los errores durante la codificación inicial.*  *Por otro lado, entre los factores que han dificultado el desarrollo, uno de los principales fue el tiempo requerido para validar las relaciones entre las entidades y la base de datos, ya que se debieron realizar varias pruebas para garantizar la integridad de los datos antes de avanzar con los CRUD.*  *También se presentaron dificultades en la sincronización del entorno local con GitHub, lo que ocasionó pequeñas demoras en la subida de versiones del código y obligó a reforzar las buenas prácticas de control de versiones.*  *Para solucionar estos inconvenientes, se estableció un proceso de validación técnica semanal, en el cual se revisa el estado del código, las conexiones y los modelos antes de continuar con nuevos módulos.*  *Además, se optimizó la planificación para dejar tiempos de revisión entre fases, asegurando que los ajustes menores no afecten la entrega final.* |

|  |
| --- |
| Actividades ajustadas o eliminadas:  ,*Durante el desarrollo de esta etapa, el plan de trabajo del proyecto SmartFlow se mantuvo en gran medida conforme a lo planificado. Sin embargo, fue necesario realizar algunos ajustes en la secuencia de actividades, principalmente para optimizar el tiempo y garantizar la correcta validación técnica antes de avanzar con el desarrollo visual.*  *Se ajustó la actividad relacionada con la validación del modelo de datos, ya que inicialmente estaba prevista dentro del sprint de diseño, pero se extendió al inicio del sprint de desarrollo para permitir una revisión más exhaustiva de las relaciones entre las entidades. Esta decisión evitó errores futuros en la fase de codificación y fortaleció la estabilidad del sistema.* |

|  |
| --- |
| Actividades que no has iniciado o están retrasadas: *Dentro del plan de trabajo del proyecto SmartFlow, la mayoría de las actividades contempladas en la Entrega 2 fueron cumplidas en los plazos establecidos. Sin embargo, existen algunas tareas que se encuentran en curso o levemente retrasadas, principalmente por motivos técnicos relacionados con validaciones y ajustes de desarrollo.*  *La principal actividad pendiente corresponde a la implementación del módulo de autenticación con contraseñas cifradas, que se pospuso debido a que fue necesario priorizar la revisión del modelo de datos y las relaciones entre entidades para evitar inconsistencias en la base de datos. Este ajuste técnico fue fundamental para garantizar la estabilidad del sistema antes de integrar la seguridad de acceso.*  *Otra tarea que se encuentra parcialmente retrasada es la creación del calendario dinámico de reservas, ya que requiere integrar la lógica del backend con la interfaz visual y validar la disponibilidad en tiempo real. Por su complejidad técnica, se decidió trasladar esta funcionalidad al próximo sprint, donde se abordará con pruebas específicas.* |