

COMPETENCIAS: TERCERA ENTREGA.

Categoría de Competencia	Competencia	Actividades Específicas del Proyecto	Argumentación
GENÉRICAS	Se comunica en español en forma oral y escrita en sus intervenciones profesionales y en su vida personal, utilizando correctamente el idioma.	Redacción de la lista de Requisitos Funcionales y No Funcionales, y las Historias de Usuario con Criterios de Aceptación.	La documentación de todos los artefactos, como lo exige la Lista de Cotejo, demuestra nuestra capacidad para expresar ideas técnicas complejas (ej. el requisito FR-02 de Mapa Interactivo) de forma clara y precisa en el idioma español.
	Usa las TIC en sus intervenciones profesionales y en su vida personal de manera pertinente y responsable.	Creación del Boceto Interactivo o Sketch, elaboración del Diagrama de Casos de Uso y la presentación de los avances del proyecto.	Utilizamos herramientas de diseño y modelado para construir los artefactos requeridos (ej. Diagrama de Casos de Uso y Boceto), lo cual es la aplicación práctica y pertinente de la tecnología para la fase de requisitos.
	Gestiona el conocimiento en sus intervenciones profesionales y en su vida personal, de manera pertinente.	Análisis de Requisitos y formulación de la Propuesta de Valor.	Investigamos el ecosistema digital y las problemáticas del usuario (ej. Daniel, el estudiante nuevo) para asegurar que UADY ASSISTANT fuera una solución única y viable. Esta investigación y análisis es la base para gestionar el conocimiento y tomar decisiones de diseño.
	Trabaja con otros en ambientes multi, inter y transdisciplinarios de manera cooperativa.	División de responsabilidades dentro del equipo (20% de contribución para cada miembro) y uso del método SCRUM.	El trabajo en equipo se evidencia en la división organizada de tareas y en la adopción del marco SCRUM, el cual es un método cooperativo que requirió comunicación constante para cumplir con la planificación de la entrega.
	Toma decisiones en su práctica profesional y personal, de manera responsable.	Priorización de Requisitos funcionales y no funcionales.	Se cumple al decidir qué características son esenciales para la primera versión del producto. Priorizamos las funcionalidades clave como el Mapa Interactivo (FR-02) y la Usabilidad (NFR-01) para asegurar la solución directa a los principales problemas de los usuarios.
ESPECÍFICAS	Analiza las características principales de los métodos, técnicas, procedimientos y buenas prácticas, utilizados en los procesos de desarrollo y gestión del software, de acuerdo con el cuerpo de conocimientos reconocido por la disciplina.	Uso del marco de trabajo SCRUM y aplicación de la técnica de Historias de Usuario para el desarrollo.	Se cumple al aplicar SCRUM (un método de gestión) y las técnicas de elicitación de requisitos (un proceso de desarrollo) reconocidos en el SWEBOK para estructurar la primera entrega.
	Interpreta correctamente tablas, gráficas, diagramas y textos expresados con lenguaje matemático y científico que se utilizan en las matemáticas de nivel superior.	Elaboración del Diagrama de Casos de Uso y la matriz de requisitos del sistema.	Cumplimos al modelar el sistema gráficamente con el Diagrama de Casos de Uso (lenguaje UML) e interpretar los requerimientos en formato estructurado (matriz de requisitos), lo cual demuestra el uso de lenguaje científico para la Ingeniería de Software.
	Analiza la evolución disciplinar de la Ingeniería de Software, así como las características del profesionista vinculado con la misma, de acuerdo con el marco teórico y los modelos curriculares de la disciplina.	Justificación de la Propuesta de Valor y enfoque del proyecto en resolver una problemática social/académica.	Nuestro proyecto se justifica como una solución ingenieril a un problema real de la UADY, aplicando el cuerpo de conocimientos (ej. modelado y especificación de requisitos) de la disciplina para lograr la adaptación del estudiante (perfil del profesionista).
	Analiza los principales métodos, técnicas, procedimientos y buenas prácticas utilizados en las fases de requisitos, diseño, codificación, pruebas y mantenimiento del software, de acuerdo con el cuerpo de conocimientos reconocido por la disciplina.	Fase de Requisitos: Definición de Historias de Usuario y Requisitos Funcionales/No Funcionales. Fase de Diseño: Creación del Boceto Interactivo.	El proyecto cubre las dos primeras fases: la fase de requisitos se abordó con la especificación detallada de Historias de Usuario, y la fase de diseño se inició con el Boceto Interactivo para la interfaz.
	Selecciona el Modelo de Ciclo de Vida del Software en el inicio de un proyecto, con base en las características del equipo de desarrollo, del cliente y de la problemática.	Selección del Modelo SCRUM como ciclo de vida del software.	Elegimos el modelo SCRUM ya que es el más adecuado para un proyecto de este tipo: los requisitos podrían cambiar o evolucionar con el tiempo, y el desarrollo en entregas nos permite adaptarnos a ese feedback.
	Analiza los principales métodos, técnicas, procedimientos, y buenas prácticas utilizados en los procesos de estimación, planificación, seguimiento, control, calidad y configuración del software, de acuerdo con el cuerpo de conocimientos reconocido por la disciplina.	Planificación de las entregas usando SCRUM, estableciendo roles y el cronograma de trabajo.	Se cumple al aplicar métodos de planificación y control del proyecto, usando SCRUM para establecer la estructura de trabajo, dividir las tareas y garantizar el seguimiento de los artefactos para asegurar la calidad de la entrega.
	Identifica los principales factores humanos que inciden en el éxito o fracaso de los procesos de software, de acuerdo con el marco teórico de la disciplina.	Identificación de los usuarios primarios y sus historias de usuario, y definición del requisito de Usabilidad.	El proyecto se centra en el factor humano al identificar los problemas específicos de los estudiantes. El requisito de Usabilidad asegura que la aplicación sea fácil de usar, garantizando que el diseño esté alineado con las necesidades reales y minimizando el riesgo de fracaso por rechazo del usuario.

Crítica constructiva al curso:

El equipo llegó a la conclusión de que, a través de las actividades designadas por la maestra, los integrantes pudimos lograr desarrollar las capacidades y habilidades esperadas en el diseño del curso, aprendimos, aunque superficialmente, a usar GitHub, practicamos las metodologías ágiles y pusimos en práctica la teoría vista en clase. Como sugerencia para poder disfrutar mejor la clase en el futuro, y lograr un mayor aprendizaje, sugerimos menos dependencia de las diapositivas que ya habían sido diseñadas para el curso, y un poco de mayor espontaneidad. Nos gustaría ver más historias reales acerca del uso de las metodologías y de su funcionamiento en el entorno laboral real.