

1ra Reflexión. Habilidades para Requerimientos

Reflexiona sobre 3 *habilidades técnicas* asociadas a la fase de Requerimientos. Indica su importancia dentro del ciclo de Ingeniería de Requerimientos. Incluye argumentos de tu selección.

1. Obtención de Requerimientos

La obtención de requerimientos es la parte mas importante de toda esta fase de los requerimientos ya que todo lo que salga de aquí marcara las bases para la construcción del sistema

2. Análisis de Requerimientos

El análisis de Requerimiento es esencial para la determinación de los requerimientos ya que en esta fase es donde reduciremos los errores y plantearemos un modelo que nos permita establecer un vínculo con las especificaciones además de que fijaremos las reuniones con el cliente para discutir acerca de los requerimientos.

3. Verificación de Requerimientos

La verificación de requerimientos es importante por que es la parte que nos dictara que requerimiento son los que funcionaran haciendo pruebas de escritorio, revisando los documentos, etc....

Rodríguez, G. J. (2022, 7 octubre). *Técnicas efectivas para la toma de requerimientos*.

Northware. <https://www.northware.mx/blog/tecnicas-efectivas-para-la-toma-de-requerimientos/>

2da Reflexión. CU vs US

Explica de forma clara la diferencia entre Especificación de Casos de Uso vs Historias de Usuario. Adicionalmente indica en qué casos se recomiendan el uso de cada una de estas herramientas.

La diferencia entre la especificación de Casos de Uso contra la de las Historias de usuario son que aunque ambas son descripciones textuales, las historias de usuario son descripciones generales e informales desde el punto de vista del usuario o el cliente

Atlassian. (s. f.). Historias de usuario | Ejemplos y plantilla. Recuperado 30 de octubre de 2022, de <https://www.atlassian.com/es/agile/project-management/user-stories>

Descripción de especificación de caso de uso de gestión de requisitos. (s. f.). © Copyright IBM Corp. 2016. Recuperado 30 de octubre de 2022, de <https://www.ibm.com/docs/es/elm/6.0.2?topic=cases-use-case-specification-outline>

3ra Reflexión. Métodos de Diseño

Selecciona un producto resultante de la etapa de diseño (Arquitectura, Base de Datos, Interfaz de Usuario, Procedimientos). Explica de forma clara y sintética algún método asociado al producto resultante.

Diseño UX

El diseño UX es el diseño de interfaces de usuario que se basa únicamente en la experiencia del usuario y la optimización de las interacciones de las personas con el producto para eso usa fundamentos como la usabilidad y la accesibilidad

4ta Reflexión. Políglotas

Dada la gran cantidad de lenguajes de programación que se generan constantemente, explica cuál sería una estrategia que te permita generar una cultura políglota (transición rápida entre diferentes lenguajes).

Un buen ejercicio para poder transicionarte mejor a otros lenguajes es reafirmar los fundamentos de programación ya que todos estos son necesarios para que en un principio puedas empezar a programar y generar un buen conocimiento acerca de la estructura de los lenguajes de la programación

¿Qué es y qué hace un UX Designer? | Ironhack Blog. (2022, 15 septiembre). <https://www.ironhack.com/es/es/blog/que-es-y-que-hace-un-disenador-ux-ui>

5ta Reflexión. Métodos y Técnicas de Pruebas

Investiga sobre una método o técnica que permita incluir pruebas de forma efectiva en el proceso de desarrollo. ¿A partir de qué etapa pueden ser consideradas las pruebas?

Pruebas de integración.

Las pruebas de integración son aquellas que verifican que los distintos módulos o servicios utilizados funcionan bien en conjunto. Por ejemplo, se puede probar la interacción con la base de datos o asegurarse de que los microservicios funcionan bien en conjunto y según lo esperado.

Estas pruebas pueden empezar a ser consideradas a partir de la verificación de los requerimientos ya que necesitamos que nuestra idea ya tenga una base sólida y varios módulos que necesitemos comprobar que funcionan bien con los demás.

Atlassian. (s. f.). Los distintos tipos de pruebas en software. Recuperado 30 de octubre de 2022, de <https://www.atlassian.com/es/continuous-delivery/software-testing/types-of-software-testing>

6ta Reflexión. Mantenimiento de Software como Competencia

El programa de estudios de LIS-FMAT incluye una Competencia del PROGRAMA de estudios como "Mantenimiento de Software". Consideras correcto que sea considerada de esa manera. Justifica tu respuesta con elementos objetivos del programa de estudios.

Considero correcto que se considere una competencia dentro del programa de estudio, esto porque las competencias específicas de esta son esenciales para el desarrollo de nuestro perfil laboral y educativo en la propia escuela, ya que "Mantenimiento de Software" también está ligada a otras competencias en el mismo programa de estudios que pierden sentido sin la misma