



Ingenieria de Sistemas Computacionales.

6260 Erick Lopez Gonzalez.

Cuatrimestre 2

Turno verpertino.

INGENIERIA DE REQUISITOS DE SOFTWARE.

Lic. DANIEL. GUILLERMO CONRADO MOGUEL.

FUNDAMENTOS DE INGENIERIA DE SOFTWARE

Es una disciplina fundamental en el desarrollo de sistemas confiables y de calidad. En esta clase magistral, exploramos los conceptos clave del proceso de desarrollo de software, incluyendo el análisis de requisitos, el diseño, las pruebas y el mantenimiento. Discutimos la diferencia entre los **requisitos funcionales y no funcionales**, resaltando su importancia para **la especificación precisa del software**. También analizamos la importancia del **diseño en el desarrollo de software** y los **diferentes enfoques** de diseño utilizados. Además, abordamos el tema de las pruebas de software y su papel en la detección de errores y aseguramiento de calidad.

- 1. Ciclo de vida del software:** Se refiere a las fases o etapas que atraviesa un software desde su concepción hasta su retirada o desuso. Este proceso incluye desde la planificación y desarrollo inicial hasta el mantenimiento y la eventual obsolescencia del software.
- 2. Modelos de desarrollo de software:** La interrogante principal está en conocer cuál modelo utilizar en el proceso de desarrollo de software de un proyecto. Para darle solución a este problema se define como objetivo de la investigación proponer alternativas viables a la hora de seleccionar un modelo para el desarrollo de software teniendo en cuenta las características del proyecto.
- 3. Principios de diseño de software:** Estos principios nos proporcionan unas bases para la construcción de aplicaciones mucho más sólidas y robustas. Permiten que los cambios y la evolución del software tenga un mínimo impacto en el código ya desarrollado tratando de evitar que lo que funciona deje de funcionar y por ello que el coste del mantenimiento sea mucho menor.
- 4. Gestión de proyectos:** Es el proceso de planificar, organizar, dirigir y controlar los recursos y actividades para alcanzar los objetivos de un proyecto en tiempo, coste y calidad adecuados. En el contexto de **gestión de proyectos de software**, implica coordinar equipos de trabajo, definir entregables y gestionar riesgos, plazos y presupuesto.
- 5. Calidad del software:** Es el proceso de planificar, organizar, dirigir y controlar los recursos y actividades para alcanzar los objetivos de un proyecto en tiempo, coste y calidad adecuados. En el contexto de **gestión de proyectos de software**, implica coordinar equipos de trabajo, definir entregables y gestionar riesgos, plazos y presupuesto.

- 6. Ingeniería de requisitos de software:** a disciplina para desarrollar una especificación completa, consistente y no ambigua, la cual servirá como base para acuerdos comunes entre todas las partes involucradas y en dónde se describen las funciones que realizará el sistema.
- 7. Mantenimiento y evolución:** Son actividades cruciales en el ciclo de vida de un sistema, que aseguran que el software continúe funcionando correctamente y se adapte a cambios a lo largo del tiempo. Estas actividades ocurren después de que el software ha sido entregado y puesto en producción
- 8. Herramientas y lenguajes:** Juegan un papel fundamental, ya que permiten desarrollar, mantener, gestionar y mejorar los sistemas de manera eficiente. Emite una serie de comandos que ayudan a ordenadores, teléfonos móviles, tabletas y otros dispositivos a funcionar según lo previsto y a realizar diversas tareas. Hay varios tipos de lenguajes de programación.
- 9. Ética y lenguajes:** Conocer y respetar los contextos culturales y los matices de las distintas lenguas y comunidades

Resumen del apartado.

Es la combinación de teorías, ideas, trabajos para poder realizar las practicas de manera sencilla y comprensiva para así facilitar el uso de lenguajes de cada programa o practica que se realice durante la sesión.