

UNIVERSIDAD CATOLICA DE CORDOBA

FACULTAD DE INGENIERÍA

Asignatura: **INGENIERIA DE SOFTWARE / INGENIERIA DE SOFTWARE I**

Año lectivo: **2021**

Docentes:

- MARIA ALEJANDRA BOSIO
- JULIO CASTILLO

OBJETIVOS GENERALES:

Introducir al alumno en los conceptos fundamentales de la ingeniería del software, mostrando la necesidad de establecer un método disciplinado de desarrollo de software.

OBJETIVOS ESPECÍFICOS

- Presentar los modelos de proceso del desarrollo de software.
- Describir las características de las metodologías dirigidas por plan y las de las metodologías ágiles reconociendo su adecuada aplicación.
- Introducir el concepto de requerimiento y los métodos para su obtención y especificación.
- Definir parámetros destinados a representar características, comportamiento y rendimiento de los sistemas informáticos.
- Describir las técnicas de prueba en sistemas informáticos.
- Introducir el concepto de la Calidad en Sistemas Informáticos
- Introducir los conceptos fundamentales de la gestión de proyectos de software.

SÍNTESIS CONCEPTUAL DE LA ASIGNATURA (ABSTRACT)

El objetivo principal de la materia es presentar al estudiante los métodos fundamentales para el desarrollo de software en forma disciplinada y precisa. Para ello se inicia con la descripción general del proceso de desarrollo de software analizando los diferentes modelos y sus casos típicos de aplicación.

Se analiza todo lo relativo a la gestión de procesos en general y a la gestión particular de procesos de desarrollo de software poniendo énfasis en los estándares de calidad. Se introduce al estudiante en la gestión y liderazgo de proyectos de desarrollo de software.

CONTENIDOS

Unidad 1: Los procesos de Software

Introducción a la Ingeniería de software. Concepto de software y de ingeniería de software. Desarrollo de software profesional.

Modelos de proceso de software. Modelo en cascada. Ciclo de Vida del desarrollo de software. Desarrollo incremental. Ingeniería de software orientada a la reutilización.

Especificación del software. Diseño e implementación. Validación. Evolución. Prototipos. Modelo en espiral. Proceso unificado. Ejemplos y Estudio de casos.

Unidad 2: Desarrollo ágil de software

El desarrollo ágil de software. Introducción a las metodologías ágiles Concepto de agilidad. El manifiesto ágil. Los principios del manifiesto ágil. Historias de usuario. Casos en los que es conveniente aplicar metodologías ágiles. Scrum. El equipo. Sprints. Etapas. Valoración. Selección. Desarrollo. Revisión. Programación extrema (XP). Bases. Prácticas. Desarrollo dirigido por pruebas (TDD). Desarrollo dirigido por un plan y desarrollo ágil. Comparativa entre metodologías ágiles y tradicionales. Estudio de casos.

Unidad 3: Especificación de requerimientos

Introducción. Concepto de Requerimiento. Requerimientos de usuario y de sistema. Requerimientos funcionales. Requerimientos no funcionales. Obtención de requerimientos. Especificación de requerimientos. Especificación en lenguaje natural. Especificaciones estructuradas. Adquisición y análisis. Validación. Administración. Ejemplos y Estudio de casos.

Unidad 4: Usabilidad

Usabilidad. Antecedentes. Características principales de la usabilidad. Definiciones y conceptos. La Usabilidad en el Proceso Desarrollo de Software. La Interfaz y su importancia. Interacción Humano Computadora. El Factor Humano. Tipos de usuarios. Métodos de evaluación de la usabilidad. Usabilidad y Accesibilidad. Diseño Centrado en el Usuario. Análisis de Requerimientos en el Diseño Centrado en el Usuario. El proceso de ingeniería de usabilidad. Conceptos de Diseño. Heurística de Diseño.

Unidad 5: Modelado de sistemas

Modelos de contexto. Modelos de interacción. Modelos estructurales. Diagramas de Clases. Generalización y agregación. Modelos de comportamiento. Orientación a eventos. Ejemplos. Estudio de casos.

Unidad 6: Diseño arquitectónico

Definición. Decisiones de diseño arquitectónico. Vistas arquitectónicas. Patrones. Modelo-Vista-Controlador. Arquitectura de capas. Arquitectura de repositorio, Arquitectura cliente servidor. Arquitectura de tubería y filtro. Arquitectura de micro servicios. Estudio de casos. Ejemplos.

Unidad 7: Diseño e implementación

Diseño orientado a objetos. Patrones de diseño. Implementación. Conflictos de implementación. Administración. Herramientas de Implementación. Estándares y codificación. Integración. Selección de la plataforma. Selección del lenguaje. Herramientas. Reusabilidad. Ejemplos.

Unidad 8: Pruebas de software

Proceso de Validación y Verificación. Pruebas. Inspecciones. El proceso de pruebas de software. Pruebas de desarrollo. Pruebas de unidad. Pruebas de componentes. Elección de casos de pruebas. Pruebas de sistema. Desarrollo dirigido por pruebas. Automatización de pruebas. Pruebas de versión. Pruebas basadas en requerimientos. Pruebas de escenario. Pruebas de rendimiento. Pruebas de usuario. Pruebas de aceptación. Ejemplos. Estudio de casos.

Unidad 9: Mantenimiento y evolución.

Concepto de mantenimiento y evolución. Tipos de mantenimiento. Correctivo. Perfectivo. Adaptativo. Preventivo. Técnicas de Mantenimiento. Métricas. Costos y beneficios. Ejemplos.

Unidad 10: Calidad del software.

Calidad del software. Estándares de calidad. Revisiones e inspecciones. Medición y métricas del software. Mejora de procesos. Análisis del proceso. Cambios y mejoras. El marco de trabajo para la mejora de procesos CMMI. Ejemplos.

Unidad 11: Gestión de proyectos.

Concepto de proyecto. Características. Ciclo de vida. Fases. Actores. Planeación y control. Definición y tipos de proyectos. Objetivos, alcance y formulación de proyectos. El director de proyecto y los componentes. Gestión de riesgos. Análisis y gestión de tiempos y costos. Seguimiento y control del proyecto. Comunicación e informes de proyecto. Finalización, evaluación y cierre. Estudio de casos.

Unidad 12: Software embebido.

Diseño de sistemas embebidos. Patrones. Sensores y actuadores. Modelado y programación en tiempo real. Análisis de temporización. Sistemas operativos de tiempo real. Ejemplos.