

Programa del Curso

Conocimientos Previos

- Dominio completo de por lo menos un lenguaje de programación de alto nivel
- Conocimientos de técnicas de especificación de tipos de datos y de encapsulamiento de procedimientos.
- Conocimientos de base de datos y análisis de sistemas.

Objetivo General

- Que el estudiante se familiarice con las técnicas para el diseño, construcción e implementación de productos de software.

Objetivos Específicos

- Aplicar en un sentido eminentemente práctico los conocimientos adquiridos
- Dominar técnicas para modularización de sistemas
- Habituarse a los ambientes de programación en los que necesiten labores de planificación, control y documentación.
- Desarrollar un sistema de mediana complejidad, mediante la aplicación de herramientas sistemáticas para la construcción de software

Contenido

- 1-) *La problemática propia del desarrollo de software*
 - Costos y tiempos: las dos variables críticas
 - Mitos y realidades: Un análisis de las posibilidades reales de proyectos de software y de los factores críticos de éxito.
- 2-) *Planificación de Productos de software*
 - Herramientas básicas de planificación: PERT, GANTT, CPM
 - Análisis y uso de algún producto típico para esto, por ejemplo MANAGER, Mc Poyect, etc.
 - Planificación a nivel de Sistema
 - Planificación a nivel de Programa
 - Estudio de un modelo de estimación de costos
- 3-) *Técnicas de Diseño*
 - Conceptos fundamentales de diseño. Aquí se usa fuertemente todo el contenido de programación II
 - Incluye técnicas para la derivación de la estructura de un producto de software. Se deben abarcar conceptos tales como acople, cohesión, rendimiento, flexibilidad, etc.
 - Herramientas y metodología de diseño
 - Uso de planos estructurados
 - Diseño por prototipos
 - Técnicas de desarrollo con orientación a objetos
- 4-) *Técnicas de implementación*
 - Verificación y conducción de equipos de programación
 - Técnicas de documentación de software
 - Conducción de pruebas de aceptación
 - Técnicas para pruebas y depuración: a nivel de módulo y a nivel de programa (top-down u bottom-up)

Evaluación

- 2 exámenes parciales (20% c/u)	40%
- Proyecto	35%
- Tareas y exámenes cortos	10%
- Exposiciones	15%
	100%

Notas

- La presentación del proyecto es requisito obligatorio para aprobar el curso
- En la entrega final del proyecto deberá adjuntar un acta de participación firmada por los integrantes el grupo

Bibliografía

- Ingeniería de software, Fairley Richard
- Software Engineering, Shooman
- Software Engineering a practitioner approach, Pressman Roger
- The art of software Testing Myers
- Software Engineering Economics, Boehm Bary
- Análisis y diseño de sistemas de Información, Senn James