

UNIVERSIDAD INTERAMERICANA DE PUERTO RICO
RECINTO DE AGUADILLA
DEPARTAMENTO DE CIENCIAS Y TECNOLOGIA

P R O N T U A R I O

COMP 3400 – INGENIERÍA DE SOFTWARE

TITULO DEL CURSO	: INGENIERÍA DE SOFTWARE
CÓDIGO	: COMP 3400
SECCIÓN	: 16772
CREDITAJE	: 3 CRDS
PRE-REQUISITO	: COMP 2900
HORAS DE CLASE	: 3 HRS
LAB	: Abierto
DÍA Y HORA	: TR 7:00 – 8:30PM
SALÓN	: CIT 104
TEXTO	: P. Jalote, “An Integrated Approach to Software Engineering”, 2da. Ed., Springer, 1997, ISBN: 0-387-94899-6
PROFESOR	: José Navarro
E-MAIL	: jnavarro@aguadilla.inter.edu
WEB PAGE	: www.geocities.com/navarro_inter
HORAS OFICINA	: (En la oficina) Martes & Jueves 8:30 – 9:30 PM Martes 9:30 – 10:30 PM (En el web) Lunes y Miércoles 8:00 – 9:30PM

DESCRIPCION DEL CURSO:

Técnicas en el ciclo de desarrollo e implantación de software: análisis, especificaciones, diseño, verificación, validación, documentación y mantenimiento. Medidas de eficiencia. Requiere horas adicionales en un laboratorio abierto.

OBJETIVOS TERMINALES

Conocer el ciclo de desarrollo e implantación de software

Utilizar las herramientas de ingeniería de programación en del desarrollo e implantación de software

OBJETIVOS CAPACITANTES

Determinar los requisitos para el desarrollo de un software para una aplicación específica

Documentar los requisitos para el desarrollo de un software para una aplicación determinada

Utilizar las especificaciones de los requisitos de un problema en el diseño de un software

Aplicar técnicas de verificación y validación de especificaciones en el diseño de un software

Documentar apropiadamente los componentes de un software

Conocer las técnicas para facilitar y llevar a cabo el mantenimiento de un software

CONTENIDO DEL CURSO

Introducción al curso

Actividades:

- Discusión de los temas y destrezas que se cubrirán en el curso
- Discusión en grupo sobre los conocimientos que tienen los estudiantes sobre la ingeniería de programación

Introducción a la ingeniería de programación

Actividades

- Discusión del problema del software (1.1)
- Discusión del problema del software desde el punto de vista de la ingeniería de programación (1.2-1.4)

Introducción al proceso de desarrollo de software

Actividades

- Estudio de las características del proceso de desarrollo de software (2.1-2.2)
- Estudio de los diferentes modelos para el desarrollo de software (2.3)
- Estudio de la gerencia para el manejo y desarrollo del proyecto (2.4-2.6)

Análisis de requisitos y especificaciones del software

Actividades

- Se estudiarán los métodos utilizados para especificar los requisitos de un software para resolver un problema (capítulo 3)
- Luego de seleccionar un proyecto para el desarrollo de un software, con el cual trabajarán a lo largo del semestre; los estudiantes desarrollarán un documento de requisitos para el mismo.
- **Parcial #1**

Proceso de planificación para un proyecto de desarrollo de un software

Actividades

- Se discutirán diferentes métodos para el estimado de costos, la programación de actividades y la selección de personal para el desarrollo del software (4.1-4.3)
- Se discutirán y analizarán los métodos para monitorear el progreso de un proyecto de software (4.4-4.6)
- Se discutirán mecanismos para el manejo de riesgos durante el desarrollo de un proyecto (4.7)
- Los estudiantes desarrollarán un documento presentando su plan para el desarrollo del proyecto del curso

Diseño de software orientado a funciones

Actividades

- Se discutirán y analizarán las estrategias para el diseño de software en forma orientada a funciones (5.1-5.4)
- Se estudiarán diferentes técnicas para la verificación del software (5.5-5.6)
- Los estudiantes crearán el diseño orientado a funciones para su proyecto
- *Opcional: Los estudiantes que lo prefieran pueden realizar un diseño orientado a objetos en lugar de orientado a funciones.

Diseño detallado

Actividades

- Discusión de métodos utilizados para establecer las especificaciones de los componentes del diseño del software (Capítulo 7)
- Los estudiantes desarrollaran un documento estableciendo los detalles de los componentes de su diseño según los métodos estudiados
- **Parcial #2**

Fase de codificación

Actividades

- Se estudiarán las consideraciones que se deben tomar para la codificación del software (8.1-8.2)
- Los estudiantes realizarán la codificación de su programa

Pruebas del software

Actividades

- Se discutirán diferentes métodos para probar el funcionamiento del programa (9.1-9.3, 9-5)
- Los estudiantes realizarán la pruebas de funcionamiento de su proyecto y someterán un reporte sobre el proceso y sus hallazgos.

EVALUACION:

2 exámenes parciales	34%
1 final	16%
proyectos	50%

REGLAS DEL CURSO:

1. Los exámenes de reposición se ofrecerán durante el período de exámenes finales en una fecha fijada por el profesor. Sólo se ofrecerán exámenes de reposición a aquellos estudiantes que por alguna razón se ausenten el día del examen. Es responsabilidad del estudiante que necesita tomar un examen de reposición recordarle al profesor su situación de forma tal que el profesor pueda preparar el mismo.
2. Cada estudiante debe poseer una dirección de correo electrónico y proveer la misma al profesor durante las primeras dos semanas de clase. Es responsabilidad del estudiante revisar su correo electrónico a diario para cotejar si hay alguna correspondencia relacionada con el curso (provista por el profesor o alguna otra fuente).
3. Cada estudiante debe ser capaz de realizar búsquedas de información utilizando diferentes motores de búsqueda para el WEB.
4. Durante los exámenes, en caso de que el profesor permita utilizar algún material de referencia, no se permitirá el que un estudiante le facilite material a otro (una vez comenzado el examen) aún en los casos en que el material pertenezca al segundo estudiante.
5. No se permitirá el uso en el salón de material fotocopiado ni ningún otro material que viole leyes de derechos de autor.
6. En caso de que el profesor provea una página en el internet con información sobre el curso, es responsabilidad del estudiante verificar su contenido frecuentemente (al menos dos veces por semana).

REFERENCIAS:

Carnegie Mellon, Software Engineering Institute, “*SEI Headlines*”, url: <http://www.sei.cmu.edu/>, Carnegie Mellon University, 2002.

IEEE Computer Society, “*TCSE - IEEE Technical Council on Software Engineering Software Engineering Information Center*”, url: <http://www.tcse.org/>, 2003.

Jonathan Bowen, “*Formal Methods*”, url: <http://www.afm.sbu.ac.uk/>, South Bank University, London, UK, 25 February 2003

David Alex Lamb, “*Software Engineering Archives*”, url: <http://www.qucis.queensu.ca/Software-Engineering/>, Queen's University, Ontario, 2003.

Philip Johnson, “*The WWW Formal Technical Review Archive*”, url: <http://zero.ics.hawaii.edu/%7Ejohanson/FTR/>, Department of Information and Computer Sciences, University of Hawaii, 2565 The Mall Honolulu, HI, 1999.

R.S. Presuman and Associates, “*Software Engineering Resources*”, url: <http://www.rspa.com/spi/index.html>, Boca Ratón, Florida, 2003.

IIT Software Engineering Group, “*A Software Engineering Resource List*”, url: <http://wwwsel.iit.nrc.ca/favs/>, National Research Council, Canada, 1998.

ver 2006-01-21