



# TEMARIO DE CURSO INGENIERÍA DE SOFTWARE I OTOÑO 2011 V1.2 (15-Ago-2011)

## INSTRUCTORA

***Dra. María del Pilar Gómez Gil***

*Oficina 8-228 Edificio de Computación*

*Teléfono: (222) 266-3100 ext. 8228*

[\*pgomez@ccc.inaoep.mx\*](mailto:pgomez@ccc.inaoep.mx)

[\*http://ccc.inaoep.mx/~pgomez/\*](http://ccc.inaoep.mx/~pgomez/)

## PAGINA DEL CURSO

[\*http://ccc.inaoep.mx/~pgomez/cursos/ingsw/\*](http://ccc.inaoep.mx/~pgomez/cursos/ingsw/)

## OBJETIVO DEL CURSO

Este curso tiene como objetivo desarrollar conocimiento y habilidades para la aplicación sistemática y eficiente de técnicas integrales para el desarrollo de productos software, con un enfoque de aseguramiento de calidad.

## OBJETIVOS ESPECÍFICOS

- Conocer metodologías actuales para desarrollo de software, y desarrollar habilidades para su uso cotidiano
- Desarrollar capacidades de administración de proyectos de software, así como hábitos de uso de herramientas de administración y control de proyectos
- Conocer técnicas para mejora de procesos de desarrollo de software
- Desarrollar habilidades de expresión escrita y oral para documentación el desarrollo de proyectos de software
- Dominar técnicas básicas de aseguramiento de calidad en desarrollo de software

## **MÉTODO DE TRABAJO**

Los estudiantes revisarán el material sugerido antes de las sesiones de clase. Durante las sesiones, la instructora y/o los estudiantes expondrán las ideas principales del tema, se discutirán las ideas exploradas y se discutirán las estrategias para sus aplicaciones.

Los estudiantes desarrollarán en equipo un proyecto de software pequeño, siguiendo todos los componentes metodológicos vistos en clase, el cual será calificado en cada una de sus partes. Al final del curso, los estudiantes expondrán los resultados obtenidos en sus proyectos.

## **CONTENIDO DEL CURSO.**

1. El proceso de desarrollo de software
  - 1.1. Proceso y ciclo de vida de software
  - 1.2. Modelos de desarrollo
  - 1.3. Desarrollo ágil
  
2. La administración de proyectos de software
  - 2.1. Conceptos básicos
  - 2.2. Métricas
  - 2.3. Estimación
  - 2.4. Calendarización
  - 2.5. Administración del riesgo
  - 2.6. Administración de la calidad
  - 2.7. Administración del cambio
  
3. La ingeniería de software
  - 3.1. Componentes fundamentales
  - 3.2. Ingeniería de Requerimientos
  - 3.3. Modelado del análisis
  - 3.4. Ingeniería de diseño
  - 3.5. Diseño de arquitecturas
  - 3.6. Diseño de interfaz de usuario
  - 3.7. Estrategias y técnicas de pruebas
  - 3.8. Ingeniería de software basada en componentes

## **BIBLIOGRAFÍA.**

### **LIBROS RECOMENDADOS DISPONIBLES EN BIBLIOTECA DEL INAOE:**

<b>AÑO</b>	<b>AUTOR</b>	<b>TITULO</b>
2008	Schach, Stephen	Object Oriented Software Engineering (5 <sup>th</sup> . Edition) New York, NY: McGraw-Hill Companies. QA76.758 S3185 2008 BLE16321 ( <i>en reserva</i> )
2006	Käkölä, Timo and Juan Carlos Dueñas (eds.)	Software Product Lines. Research Issues in Engineering and Management, Springer. 2006. QA76.76 D47 S66395 2006
2006	Larman, Craig	Applying UML and patterns: an introduction to object-oriented analysis and design and iterative development (3 <sup>o</sup> . Edition) Pearson Education, 2006. QA76.9 035 L37 2006 ( <i>en 3eserve</i> )
2005	Pressman, Roger S.	<b>Libro de texto:</b> Software Engineering, A practitioner's Approach (6 <sup>th</sup> . Edition. McGraw Hill. 2005. QA 76.758 P75 2005 BLE 16317 ( <i>en reserva</i> )
2003	Kan, S	Metrics and models in software quality engineering (2 <sup>nd</sup> . Edition) Boston, MA: Pearson Education, Inc. QA76.96 Q35 K35 2003 BLE16320
2003	Wieggers, Karl E.	Software Requirements: practical techniques for gathering and managing requirements throughout the product development cycle (2 <sup>nd</sup> . Ed) . Redmond, WA: Microsoft Press, 1999. QA76.76 D47 W517 2003 BLE16323 ( <i>en reserva</i> )
2001	D´Souza, Desmond F.	Objects Components and Frameworks with UML. The Catalysis Approach. Alan Cameron Wills QA 76.9 O35 D76 2001
2000	Hunt, John.	The Unified Process for Practitioners. Springer QA76.9 O35 H86 2000
2000	Jacobson, Iva, Grady Booch and James Rumbaugh	The Unified Software Development Process. Addison Wesley, Boston, 1999. QA76.76 D47 J35 2000
2000	Sommerville, Ian and Pete Sawyer.	Requirements Engineering. Willey 2000. QA76.76 D47 S66 2000
2003	Bass, Len, Paul Clements and Riuck Kazman	Software Architecture in Practice. Second edition. Boston: Addison Wesley. QA76.754 B37 2003 BLE16324

## **OTROS LIBROS RECOMENDADOS:**

<b>AÑO</b>	<b>AUTOR</b>	<b>TÍTULO</b>
1998	Booch, Grady	Análisis y Diseño Orientado a Objetos con aplicaciones. Segunda Edición Pearson Addison Wesley Longman.
2004	IEEE Computer Society	Guide to the Software Engineering Body of Knowledge (SWEBOK) 2004, disponible en: <a href="http://www.swebok.org/">http://www.swebok.org/</a>
1999	S. Robertson & J. Robertson.	Mastering the Requirement Process. ACM Press, 1999

Durante el curso se revisarán varios artículos y otros documentos.

## **EVALUACIÓN DEL CURSO**

Dos exámenes (medio curso y final)	50%
Exámenes en clase, tareas, exposiciones y participación	20%
Proyecto del curso	30%

Los exámenes incluyen todo el material comentado en clase ya sea por la instructora o por los expositores.

## **FECHAS IMPORTANTES**

Primer día de clases	15 de Agosto 2011
Último día de clases	2 de Diciembre 2011
Entregas del proyecto	ver <a href="#">calendario del proyecto</a>
Primer examen	6 de Octubre 2011
Segundo examen	22 de Noviembre 2011
Presentaciones del proyecto	por asignar, entre el 5 y 7 de Diciembre 2011

## **CODIGO DE ETICA**

Se espera que el/la estudiante observen en todas las actividades del curso, el comportamiento ético adecuado a cualquier profesional de ciencias computacionales. Como un ejemplo de Código de Ética consultar el de la IEEE ([www.ieee.org](http://www.ieee.org)) o de la ACM ([www.acm.org](http://www.acm.org)).