



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD CIENCIAS DE LA COMPUTACION

PROGRAMA DE LA MATERIA CORRESPONDIENTE A LA LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN.

Coordinación: Área de Bases de Datos

NOMBRE DE LA MATERIA:	Ingeniería de Software Avanzada
------------------------------	---------------------------------

Clave: CCO 490

Créditos: 10

Modalidad: Escolarizada

Nivel de Ubicación: Formativo

Tipo de Materia: Optativa

PRE-REQUISITOS:	CCO 207 Ingeniería de Software
------------------------	--------------------------------

MATERIA CONSECUENTE:	Ninguna
-----------------------------	---------

TIEMPO TOTAL ASIGNADO:	80
-------------------------------	----

PRIMAVERA – OTOÑO

HRS. TEÓRICAS/SEM:	5	HRS. PRÁCTICAS/SEM:	0
---------------------------	---	----------------------------	---

VERANO

HRS. TEÓRICAS/SEM:	10	HRS. PRÁCTICAS/SEM:	0
---------------------------	----	----------------------------	---

AUTOR(ES) DEL PROGRAMA CON FECHA: ABRIL 2000	
ETELVINA ARCHUNDIA SIERRA	
BEATRIZ BERNABÉ LORANCA	
MA. DEL ROCÍO BOONE ROJAS	
MARCO ANTONIO SORIANO ULLOA	
RAMÓN AGUIRRE VARA	

REVISADO POR:	CARMEN CERON GARNICA, MARÍA JOSEFA SOMODEVILLA GARCÍA, BEATRIZ BERNÁBE LORANCA
APROBADO POR:	F.C.C
AUTORIZADO POR:	VICERECTORIA DE DOCENCIA

FECHA DE ELABORACIÓN/REVISIÓN:	Julio 2000 / Mayo 2003
VIGENCIA:	A partir del Periodo de Otoño del 2000

JUSTIFICACIÓN:

Las demandas tecnológicas, científicas y económicas de diversas organizaciones requieren que los proyectos de sistemas cumplan con los requisitos estándares de calidad que permitan la confiabilidad en el manejo y proceso de la información.. Por ello, el conocimiento teórico y práctico de la Ingeniería de Software Avanzada es fundamental para el estudio de Licenciaturas e Ingenierías en Computación.

OBJETIVO GENERAL DE LA MATERIA:

El estudiante identificará y aplicará los conceptos y técnicas relacionadas con la Planificación, Gestión y Control de Calidad de los Proyectos de Software. Así como que adquiera un panorama de las tendencias de desarrollo, abordando como caso de estudio la Ingeniería de Software Orientada a Objetos.

CONTRIBUCIÓN DE LA SIGNATURA AL PERFIL DE EGRESO:

En el perfil del egresado se establece que éste debe tener una visión y creatividad para resolver problemas de las Ciencias de la Computación. Un área importante de la computación es la Ingeniería de Software. Esta disciplina proporcionará conocimientos sólidos para el desarrollo de proyectos de software con técnicas de calidad y nuevas tendencias para la construcción de aplicaciones computacionales. La asignatura de Ingeniería de Software Avanzada contribuye a la formación del estudiante de Licenciatura en Ciencias de la Computación.

CONTENIDO TEMÁTICO

UNIDAD: 1				TÍTULO: GESTIÓN DEL PROYECTO DE SOFTWARE		
OBJETIVO ESPECÍFICO: El estudiante identificará los conceptos y técnicas convencionales de gestión de un proyecto Bibliografía [1,2,3 y4]						
CONTENIDO DE LA UNIDAD		Tiempo de impartición (hrs.).		Actividades de Aprendizaje	Técnicas	Recursos Necesarios
		HT	HP			
1.1	El contexto de la gestión de proyecto	3		Introducción y Motivación; Comprensión de Ideas. La importancia de la gestión del proyecto.	Exposición del Profesor; Discusión grupal y ejemplos	Salón, pizarrón, plumones, proyector de acetatos o de video.
1.2	Cuantificación de la demanda de servicio	2		Análisis de la demanda de servicio para el desarrollo del proyecto.	Exposición del Profesor; Discusión grupal y ejemplos	Ídem.
1.3	Cuantificación del proyecto	2		Comprensión de Ideas. La importancia de tiempo, recursos humanos y dinero dentro del desarrollo de un proyecto.	Exposición del Profesor; Discusión grupal y ejemplos	Ídem.
1.4	Análisis de costo-beneficio	2		Comprender y analizar ideas. Identificar tareas requeridas para valorar riesgos técnicos y de costo-beneficio del proyecto.	Exposición del Profesor; Discusión grupal y ejemplos	Ídem.

1.5	Mecanismo de adopción del nuevo proyecto	2		Comprensión de Ideas. Actividades de Control para el desarrollo del proyecto	Exposición del Profesor; Discusión grupal y ejemplos	Ídem.
1.6	Proyecto	3		Comprensión de Ideas y análisis. Identificar modelos y etapas para el desarrollo de un proyecto	Exposición del Profesor; Discusión grupal y ejemplos	Ídem.
HORAS TOTALES:		14				

UNIDAD: 2				TÍTULO: PLANIFICACIÓN DE PROYECTOS DE SOFTWARE		
<p>OBJETIVO ESPECÍFICO: El estudiante será capaz de realizar la planificación de un proyecto a través de las técnicas y modelos que han sido suficientemente evaluados. Bibliografía [1,2,3 y4]</p>						
CONTENIDO DE LA UNIDAD		Tiempo de impartición (hrs.).		Actividades de Aprendizaje	Técnicas	Recursos Necesarios
		HT	HP			
2.1	Planificación	4		Comprensión de Ideas. Identificar los objetivos, ámbito, recursos y estimación del proyecto de software. Analizar y Sintetizar. Identificar la viabilidad de desarrollo de software. Ejemplos	Exposición del Profesor; Discusión grupal y ejemplos	Salón, pizarrón, plumones, proyector de acetatos o de video.
2.2	Técnicas de Descomposición (LDC, PF, etc.)	4		Comprender Ideas. Identificar las distintas	Exposición del Profesor; Discusión	Ídem.

			técnicas para estimar el tamaño de software utilizando LDC, PF, etc. Analizar y sintetizar Realizar la estimación basada en LDC y PF en un proyecto. Reflexión ventajas y desventajas.	grupal y ejemplos	
2.3	Modelos empíricos de Estimación	4	Comprensión de Ideas. La importancia y estructura de los modelos empíricos (Walson-Félix, Bailey-Basili, etc) Analizar el modelo COCOMO y sus características. Analizar y sintetizar. Realizar la estimación de un proyecto mediante los modelos empíricos	Exposición del Profesor; Discusión grupal y ejemplos	Ídem.
HORAS TOTALES:		12			

UNIDAD: 3		TÍTULO: CONTROL DE CALIDAD DEL SOFTWARE			
<p>OBJETIVO ESPECÍFICO: El estudiante conocerá las diferentes técnicas y metodologías mediante las cuales se diseñan y producen productos de programación de calidad Bibliografía [1,2,,5,7,8]</p>					
CONTENIDO DE LA UNIDAD	Tiempo de impartición (hrs).	Actividades de Aprendizaje	Técnicas	Recursos Necesarios	

		HT	HP			
3.1	Conceptos de Calidad	2		Motivación y Comprensión de Ideas. Importancia de la calidad en el desarrollo de proyectos de software. Definición de calidad, control de calidad, garantía, costos y factores de calidad.	Exposición del Profesor; Discusión grupal y ejemplos Reflexión	Salón, pizarrón, plumones, proyector de acetatos o de video.
3.2	Técnicas para el control de calidad	5		Comprensión de ideas. Identificar modelos y técnicas de calidad para el control de desarrollo de software. Análisis y síntesis Estudio de casos prácticos.	Exposición del Profesor; Discusión grupal y ejemplos Reflexión	Ídem.
3.3	Normas de Calidad	5		Comprensión de ideas. Identificar las normas e indicadores de calidad de los distintos modelos de calidad. Analizar y sintetizar. Realizar un estudio de caso del control de calidad de una empresa de software, aplicando técnicas y normas.	Exposición del Profesor; Discusión grupal y ejemplos Reflexión	Ídem.
HORAS TOTALES:		12				

UNIDAD: 4	TÍTULO: INGENIERÍA DE SOFTWARE ORIENTADA A OBJETOS
OBJETIVO ESPECÍFICO: El estudiante será capaz de desarrollar un proyecto de software aplicando el enfoque orientado a objetos Bibliografía [1,2,3 y4]	

CONTENIDO DE LA UNIDAD		Tiempo de impartición (hrs.).		Actividades de Aprendizaje	Técnicas	Recursos Necesarios
		HT	HP			
4.1	Conceptos del paradigma	2		Motivación. Lectura de artículos de la aplicación del paradigma en proyectos de software. Comprensión de Ideas. Identificar elementos básicos del paradigma (Clases, objetos, atributos, métodos, mensajes, encapsulamiento, herencia y polimorfismo).Ejemplos.	Exposición del Profesor; Discusión grupal y ejemplos Reflexión	Salón, pizarrón, plumones, proyector de acetatos o de video.
4.2	Modelo de objeto	2		Comprensión de Ideas. Identificar los elementos de un modelo de objetos (identificar clases y objetos, especificación de atributos, definición de operaciones y fin de un objeto) Análisis y sintetizar. Realizar estudio de un problema.	Exposición del Profesor; Solución de problemas, Discusión grupal y ejemplos Reflexión.	Ídem.
4.3	Administración de un proyecto orientado a objetos	3		Comprensión de Ideas. Identificar las actividades de planificación, métricas y estimación en proyectos orientados a objetos. Análisis y síntesis. Realizar e identificar diferencias del enfoque OO y de la Ingeniería de Software convencional.	Exposición del Profesor; Discusión grupal y ejemplos Reflexión.	Ídem.
4.4	Reingeniería de Software	3		Comprensión de Ideas. Definición e identificar del	Exposición del Profesor; Discusión	Ídem.

				proceso de reingeniería de software .	grupal y ejemplos Reflexión.	
4.5	El plan de proyecto de software orientado a objetos	3		Compresión de Ideas. Identificar las etapas de un proyecto de software orientado a objetos. Analizar y sintetizar. Realizar el plan de un proyecto y verificar si es viable mediante el enfoque OO	Exposición del Profesor; Discusión grupal y ejemplos Reflexión.	Ídem.
	HORAS TOTALES:	13				

UNIDAD: 5				TÍTULO: TÉCNICAS DE INGENIERÍA DE SOFTWARE ORIENTADA A OBJETOS		
<p>OBJETIVO ESPECÍFICO: El estudiante será capaz de desarrollar un proyecto de software aplicando técnicas específicas de desarrollo del enfoque orientado a objetos Bibliografía [1,2,3 y4]</p>						
CONTENIDO DE LA UNIDAD		Tiempo de impartición (hrs.).		Actividades de Aprendizaje	Técnicas	Recursos Necesarios
		HT	HP			
5.1	Análisis orientado a objetos	6		Comprensión de Ideas. Identificar panorama AOO, el enfoque unificado para el AOO. Analizar y sintetizar Componentes del modelo de AOO. Realizar el proceso de AOO (casos de uso,	Exposición del Profesor; Solución de problemas, Discusión grupal y ejemplos Reflexión.	Salón, pizarrón, plumones, proyector de acetatos o de video.

				modelado de clases-responsabilidades-colaboraciones, estructuras y jerarquías y definición de subsistemas) utilizando ejemplos. Aplicación. Realizar el AOO a un caso de estudio de desarrollo de software		
5.2	Diseño orientado a objetos	6		Comprensión de Ideas. Identificar panorama DOO, el enfoque unificado para el DOO. Analizar y sintetizar Realizar el proceso del diseño del sistema, de objetos, patrones de diseño, programación OO. Aplicación. Revisar el caso de estudio y diseñar el Diseño Orientado a Objetos.	Exposición del Profesor; Solución de problemas, Discusión grupal y ejemplos Reflexión.	Ídem.
5.3	Pruebas orientadas a objeto	4		Comprensión de Ideas. Identificar los tipos de pruebas de los modelos de AOO y DOO. Analizar y sintetizar. Reconocer las estrategias de pruebas OO. Aplicación. Diseñar casos de prueba para software OO.	Exposición del Profesor; Solución de problemas, Discusión grupal y ejemplos Reflexión.	Ídem.
HORAS TOTALES:		16				

UNIDAD: 6	TÍTULO: TENDENCIAS FUTURAS
OBJETIVO ESPECÍFICO: El estudiante conocerá diferentes tendencias de desarrollo de la Ingeniería de Software	

Bibliografía [1,2,3 y4]						
CONTENIDO DE LA UNIDAD		Tiempo de impartición (hrs).		Actividades de Aprendizaje	Técnicas	Recursos Necesarios
		HT	HP			
6.1	Perspectiva	2		Motivación. Perspectiva de la Ingeniería de Software en el desarrollo de proyectos de software en diferentes áreas. (Ingeniería de software basada en componentes, comercio electrónico, cliente-servidor, Ingeniería Web)	Exposición del Profesor; ,Discusión grupal y ejemplos Reflexión.	
6.2	El nuevo proceso de la ingeniería de software	2		Analizar y reflexionar campo de desarrollo de la Ingeniería de software (Modelos d calidad, aplicaciones Web, etc.)	Exposición del Profesor; ,Discusión grupal y ejemplos Reflexión.	Ídem.
6.3	La tecnología como conductor	2		Análisis y reflexión. Importancia de la tecnología (hardware y software) para el avance de la Ingeniería de Software. Elaborar un ensayo de las tendencias de I.S.	Exposición del Profesor; ,Discusión grupal y ejemplos Reflexión.	Ídem.
HORAS TOTALES:		6				

PRACTICAS			
UNIDAD	NOMBRE DE LA PRACTICA	OBJETIVO	HORAS

CRITERIOS DE EVALUACIÓN

EXÁMENES PARCIALES DEPARTAMENTALES		
Parcial	Contenido a evaluar	Periodos
PRIMERO 20%	UNIDADES 1, 2, 3	SEMANA 8
SEGUNDO 20%	UNIDADES 4, 5, 6	SEMANA 16

	%
Asistencias:	10(incluye participación en clase)
Proyecto Final:	15
Exámenes	40
Tareas:	25
Trabajos de Investigación:	10
Prácticas de Laboratorio:	
TOTAL:	100

REQUISITOS DE ACREDITACIÓN:
80% de asistencia a clases y obtener los criterios de evaluación.

FOMENTO DE VALORES:
El alumno adquirirá los valores de respeto, amistad y responsabilidad en el desarrollo de tareas y proyectos. Obtendrá actitudes de trabajo en equipo, disposición y la habilidad de pensamiento crítico y creativo para el desarrollo de proyectos de software en el paradigma Orientado a Objetos.

BIBLIOGRAFÍA:

1. Pressman, R.S., "Software Engineering: A practitioner's approach", Mc Graw Hill, edición 2000.
 2. Sommerville, "Software Engineering", fourth edition, Addison Wesley.
 3. Gane, C and Sarson T., "Structured System Analysis: Tools and techniques improved", Systems Technology Inc., New York 1999.
 4. Gilb, T., "Principles of Software Engineering Management", Addison Wesley, 1998
 5. Bohem, B. W., "Software Engineering Economics", Prentice Hall, 1999.
 6. Jag, Sodhi, "Software Engineering Methods, Management, and Case Tools, TAB Professional and Reference Books"
 7. Handbook of Software Quality (no disponible)
 8. Metrics and Models in Software Quality (no disponible)
- B: Básico
C: Complementario

TITULAR (RESPONSABLE) DE LA MATERIA:

M.C Ma. Beatriz Bernábe Loranca