

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
COORDINACIÓN DE FORMACIÓN BÁSICA
COORDINACIÓN DE FORMACIÓN PROFESIONAL Y VINCULACIÓN UNIVERSITARIA

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

1. Unidad Académica (s): Facultad de Ciencias Administrativas, Mexicali
Facultad de Contaduría y Administración, Tijuana
Facultad de Ciencias Administrativas y Sociales, Ensenada

2. Programa (s) de estudio: (Técnico, Licenciatura (s)) Licenciatura en Informática

3. Vigencia del plan: 2009-2

4. Nombre de la Unidad de Aprendizaje Implantación del Software

5. Clave 11874

6. HC: 2 HL HT 2 HPC HCL HE CR 6

7. Ciclo Escolar: 2012-1

8. Etapa de formación a la que pertenece: Terminal

9. Carácter de la Unidad de aprendizaje: Obligatoria X

Optativa

10. Requisitos para cursar la unidad de aprendizaje: 11864 Ingeniería de Software.

II. PROPÓSITO GENERAL DE LA UNIDAD DE APRENDIZAJE

Este curso se ubica dentro de la etapa terminal, es de carácter obligatorio y es la integración de diferentes disciplinas para la implantación y pruebas del software. El propósito de la unidad es el estudio, comprensión, aplicación y evaluación de técnicas y herramientas que faciliten el proceso de integración de software así como la documentación de un producto de software con calidad.

III. COMPETENCIA DE LA UNIDAD DE APRENDIZAJE

Implantar y documentar sistemas de información a través del uso de técnicas y herramientas computacionales para apoyar la operación de la organización. Con actitud de compromiso y confidencialidad.

IV. EVIDENCIA (S) DE DESEMPEÑO

- Documento técnico de pruebas de un proyecto de informática.
- Implementación de metodologías estandarizadas siguiendo un modelo de calidad (Ej. Moprosoft) en el desarrollo de software.
- Manual técnico y manual del usuario de proyecto de informática.
- Implementación y mantenimiento de software utilizando estandarizaciones de calidad siguiendo modelos de actualidad.

V. DESARROLLO POR UNIDADES

Competencia

Aplicar los principios básicos que guían las pruebas del software para la implementación del sistema funcional en las organizaciones, mediante la utilización de técnicas, casos y herramientas aplicables que garanticen la calidad del software, con responsabilidad, eficiencia y compromiso.

Contenido

Duración 20 hrs

Unidad I Verificación y validación de software

1 Verificación y validación de software

- 1.1 Introducción al proceso de planeación, verificación y validación de software.
- 1.2 Inspección del software.
- 1.3 Planificación de pruebas.
- 1.4 Análisis estático automatizado.
- 1.5 Desarrollo de software de sala limpia.
- 1.6 Pruebas de software.
 - 1.6.1 Pruebas de defectos.
 - 1.6.2 Pruebas de integración.
 - 1.6.3 Pruebas orientadas a objetos.
 - 1.6.4 Herramientas automáticas de pruebas.
 - 1.6.5 Revisiones de Software.
- 1.7 Validación de sistemas críticos.
 - 1.7.1 Métodos formales y sistemas críticos.
 - 1.7.2 Validación de la fiabilidad.
 - 1.7.3 Garantía de seguridad.
 - 1.7.4 Evaluación de la seguridad.

V. DESARROLLO POR UNIDADES

Competencia

Evaluar los diferentes tipos de mantenimiento para aplicarlos en productos de software implantados, mediante el uso de tecnologías y metodologías adecuadas al control y evolución del producto, con un ahorro de tiempo y en plataformas en las que se puedan hacer futuras adaptaciones con responsabilidad, disciplina, compromiso y calidad.

Contenido

Duración 16 hrs

Unidad II Mantenimiento de software

2 Mantenimiento de software

2.1 Introducción a los conceptos básicos.

2.1.1 Definiciones y terminología.

2.1.2 La necesidad del mantenimiento.

2.2 Dinámica de evolución de programas.

2.3 Mantenimiento de software.

2.3.1 Modelos del proceso de mantenimiento.

2.3.2 Actividades del mantenimiento.

2.4 Técnicas de Mantenimiento.

2.4.1 Reingeniería y mantenimiento del software.

2.4.2 Ingeniería inversa y mantenimiento del software

2.4.3 Análisis de impacto.

2.5 Estimación de costos del mantenimiento.

2.6 Evolución de la arquitectura.

V. DESARROLLO POR UNIDADES

Competencia

Aplicar el proceso de Reingeniería de software en productos de software , mediante el uso de tecnologías y metodologías adecuadas al desarrollo y evolución del producto, para asegurar la calidad del software de forma proactiva, eficiente y con calidad.

Contenido

Duración 16 hrs

Unidad III Reingeniería de software

3 Reingeniería de software

3.1 Introducción.

3.2 Traducción de código.

3.3 Ingeniería inversa.

3.4 Mejora de la estructura del programa.

3.5 Modularización de datos.

3.6 Reingeniería de datos.

V. DESARROLLO POR UNIDADES

Competencia

Realizar planes de gestión de configuración que involucren los planes de gestión de cambios necesarios, siguiendo estándares y técnicas para la construcción de software de calidad que apoyen en las versiones resultantes y entregas, con disciplina, organización y compromiso.

Contenido

Duración 12 hrs

Unidad IV Gestión de configuraciones

4 Gestión de configuraciones

- 4.1 Introducción de la gestión de configuración.
- 4.2 Gestiones de configuración.
 - 4.2.1 Actividades de la Gestión.
 - 4.2.2 Plan de gestión de configuración de software.
 - 4.2.3 Auditorías y revisiones de la configuración.
- 4.3 Gestión de cambios.
- 4.4 Gestión de versiones y entregas.
- 4.5 Construcción del sistema.

VI. ESTRUCTURA DE LAS PRÁCTICAS

No. de Práctica	Competencia(s)	Descripción	Material de Apoyo	Duración
1	Aplicar los principios básicos que guían las pruebas del software para la implementación del sistema funcional en las organizaciones, mediante la utilización de técnicas, casos y herramientas aplicables que garanticen la calidad del software, con responsabilidad, eficiencia y compromiso.	Diseñar un plan de pruebas, aplicable al proyecto de software implementado.	Computadora y paquetería.	3 hrs
2		Aplicar el plan de pruebas Diseñado.	Computadora, software a probar.	5 hrs
3	Aplicar los principios básicos que guían las pruebas del software para la implementación del sistema funcional en las organizaciones, mediante la utilización de técnicas, casos y herramientas aplicables que garanticen la calidad del software, con responsabilidad, eficiencia y compromiso.	Aplicar técnicas de mantenimiento en base al resultado obtenido en la práctica de pruebas del software.	Computadora, lenguaje de programación en el que está desarrollado el producto de software.	16 hrs
4	Aplicar el proceso de Reingeniería de software en productos de software, mediante el uso de tecnologías y metodologías adecuadas al desarrollo y evolución del producto, para asegurar la calidad del software de forma proactiva, eficiente y con calidad.	Evaluar los diferentes módulos del producto de software y en base a ello, elegir un módulo para aplicarle el proceso de reingeniería de software completo.	Computadora, lenguaje de programación en el que está desarrollado el producto de software, herramienta CASE.	16 hrs

5	Realizar planes de gestión de configuración que involucren los planes de gestión de cambios necesarios, siguiendo estándares y técnicas para la construcción de software de calidad que apoyen en las versiones resultantes y entregas, con disciplina, organización y compromiso.	Desarrollar el plan de configuración del producto de software. Aplicar una auditoria completa al software implantado.	Computadora.	12 hrs
---	--	--	--------------	--------

VII. METODOLOGÍA DE TRABAJO

Exposiciones por parte del maestro y del alumno, quien trabajará en equipo para realizar prácticas de laboratorio y en el desarrollo de un trabajo final. Se elabora la documentación pertinente al avance del desarrollo del producto, a partir de su implementación, tomando en cuenta el proceso de pruebas y la evolución del software, el cual está determinado por el alcance del curso.

VIII. CRITERIOS DE EVALUACIÓN

La evaluación de cada unidad de aprendizaje consiste en exámenes, exposición de un tema relacionado, tareas y prácticas, así como la exposición oral y escrita del desarrollo de un sistema de software.

Rubros para evaluación

Rubro	Porcentaje
Exámenes	30
Tareas	20
Exposiciones en clase	10
Proyecto (Producto de software, documentación y exposición final)	40

Criterio de evaluación

Exámenes

- Tres exámenes parciales
 - Un examen teórico.
 - Dos exámenes prácticos.

Tareas

Las tareas que consistan en investigación de un tema en particular deberán incluir lo siguiente:

- Referencias bibliográficas
- Conclusión
- Ejemplos.

Exposición

- Será determinada por el profesor
- Referente a casos empresariales actuales.
- Referencias bibliográficas

Proyecto

- Se evaluará que el producto de software esté totalmente finalizado y sea funcional.
- Se entregará los documentos solicitados y se hará una presentación del proyecto ante el grupo.

IX. BIBLIOGRAFÍA

Básica

Pressman, Roger S.
Ingeniería del software : un enfoque práctico
6ta. Edición , 2010
México: McGraw-Hill Interamericana Editores S.A. de C.V.
ISBN-10: 6071503140
ISBN-13: 978-6071503145

Sommerville, Ian.
Ingeniería de Software,
9na Edición, Pearson Addison Wesley, 2011.
Pág: 792 p. : ISBN: 9786073206037

Piattini Velthuis, Mario Gerardo.
Medición y estimación del software: Técnicas y métodos para mejorar la calidad y la productividad
Editor: Alfaomega ; 2008.
Páginas: 308 p. : ISBN: 9789701514139

Complementaria

Braude, Eric J.
Ingeniería de software : una perspectiva orientada a objetos
Editor: Alfaomega, 2003.
Páginas: 539 p. : ISBN: 9701508513

Bruegge, Bernd.
Ingeniería de software orientada a objetos
Editor: Pearson Educación, Fecha de pub: 2002.
Páginas: 553 p. : ISBN: 9702600103

Piattini Velthuis, Mario Gerardo.
Análisis y diseño de aplicaciones informáticas de gestión : una perspectiva de ingeniería del software
Editor: Alfaomega ; 2004.
Páginas: 710 p. : ISBN: 9701509870