

346/13 1
ms

**Facultad de Ingeniería
Comisión Académica de Posgrado**

Formulario de Aprobación Curso de Posgrado 2013

Asignatura: Testing en el proceso de desarrollo de software

Profesor de la asignatura ¹: Dr. Hermann Steffen, Profesor Titular, INCO

Otros docentes de la Facultad: Ing. Cecilia Apa, Asistente, INCO
(título, nombre, grado, Instituto)

Docentes fuera de Facultad: es posible, pero sus nombres no han sido definidos
(título, nombre, cargo, Institución, país)

Instituto ó Unidad: Instituto de Computación
Departamento ó Area: Ingeniería de Software

¹ Agregar CV si el curso se dicta por primera vez.
(Si el profesor de la asignatura no es docente de la Facultad se deberá designar un responsable local)

Fecha de inicio y finalización: Desde el 20/05/2013 al 21/06/2013

Horario y Salón: Lunes, miércoles y viernes de 18 a 21horas

Horas Presenciales: 60 horas
(sumar horas directas de clase – teóricas, prácticas y laboratorio – horas de estudio asistido y de evaluación)
Se deberán discriminar las mismas en el ítem Metodología de enseñanza.

Nº de Créditos: 8

Público objetivo y Cupos: Profesionales y estudiantes interesados en Ingeniería de Software, en particular profesionales informáticos involucrados con el desarrollo y la validación de software. Se trata de un curso altamente interactivo, por lo que el número de plazas no debería ser superior a 30.

Objetivos: Presentar conceptos, métodos, técnicas y procesos relativos a las múltiples actividades del testing dentro del proceso de desarrollo, de forma a mejorar la calidad del software y la eficiencia de su realización. Incluir elementos que contribuyan a la preparación de certificaciones internacionales de profesionales de testing como es el caso del Certified Tester de ISTQB.

Conocimientos previos exigidos: Profesionales informáticos con experiencia en el desarrollo de software y/o testing de software, usuarios avanzados de software involucrados con la validación de software.

Conocimientos previos recomendados: Responsables de desarrollo, responsables de testing, responsables de calidad de software, desarrolladores experimentados, testadores, usuarios avanzados con responsabilidad en la validación de software.

Metodología de enseñanza:

- Horas clase (teórico/práctico): 36
 - Horas clase (práctico):
 - Horas consulta: 20
 - Horas evaluación: 4
 - Subtotal horas presenciales: 60
 - Horas estudio: 10
 - Horas resolución ejercicios/prácticos: 12
-

2
dos.

Facultad de Ingeniería Comisión Académica de Posgrado

- Horas obligatorio/laboratorio: 38
 - Total de horas de dedicación del estudiante: 120
-

Forma de evaluación:

El curso se evaluará a partir de:

- trabajos obligatorios
 - presentaciones
 - examen final
-

Temario:

- 1- Introducción: elementos fundamentales del test de software
 - a. Desarrollo y calidad de software
 - b. Testing y Calidad de software
 - c. Evolución del área testing de software y el enfoque ISTQB
 - d. Tendencia de normalización de procesos de testing (Nueva norma ISO 29119)
 - 2- El test en el ciclo de desarrollo.
 - a. Modelos de desarrollo de software y lugar del testing
 - b. Testing en el Modelo en V
 - c. Testing en el Modelo Rational Unified Process (RUP)
 - d. Testing en el Modelo Extreme Programming (XP)
 - e. Otros modelos de desarrollo (Métodos Ágiles)
 - 3- Procesos de test de software
 - a. Modelo TMM (Test Maturity Model)
 - b. Modelo TPI (Test Process Improvement)
 - 4- Técnicas de testing y de diseño de casos de prueba
 - a. Test Estático
 - b. Test Dinámico (Funcional y Estructural)
 - c. Test no Funcional
 - 5- Principales niveles de test
 - a. Test de Componentes
 - b. Test de Integración
 - c. Test de Sistema
 - d. Test de Aceptación
 - 6- El Plan Maestro de Testing
 - a. La norma IEEE 829 y la nueva norma ISO 29119
 - b. Definición del alcance del testing
 - c. Definición de la estrategia de test
 - d. Estimación del esfuerzo de test
 - e. Definición de las etapas de testing
 - f. Formación de los equipos
 - 7- El nivel de Test de Componentes
 - 8- El nivel de Test de Integración
 - 9- Gestión de incidentes
 - a. El papel de la gestión de incidentes
 - b. Clasificación normalizada de incidentes
 - c. Herramientas de Gestión de Incidentes
 - 10- Testing basado en riesgos
 - a. Identificación de riesgos
 - b. Priorización de testing según riesgos
 - 11- Mediciones de test y modelos de confiabilidad
 - a. Principales medidas prácticas
 - b. Modelos de estimación de confiabilidad
 - 12- Herramientas, automatización y ambientes de testing
 - a. Introducción y papel de herramientas de testing
 - b. Categoría de herramientas de testing
-

Facultad de Ingeniería Comisión Académica de Posgrado

- 13- Estrategia y Plan de mejora de la actividad de testing en el ciclo de desarrollo de software
 - a. Objetivos del plan de mejora
 - b. Plan a corto y mediano plazo
 - c. Proceso de implantación
 - d. Perfil de los testadores
 - 14- Conclusiones
-

Bibliografía:

Software Testing Foundations. Andreas Spillner, 2011
Software Testing Practice. Test Management, Andreas Spillner, 2012
Syllabus Foundation Level, ISTQB, 2011
Syllabus Management Level, ISTQB, 2012
IEEE standard 829
ISO/IEC 29119 Software Testing
Software Testing with Visual Studio 2005 Team System, Tom Arnold, 2007
Pragmatic Unit Testing in C# with NUnit, Andrew Hunt, 2007
Unit Test Frameworks, Paul Hamill, 2005
Test Process Improvement, Tim Koomen, 1999
Managing the Testing Process, Rex Black, 2009
