

PROGRAMA DE CURSO

Código	Nombre			
CC68S	Calidad y Diseño de Software			
Nombre en Inglés				
Quality and Design in Software				
SCT	Unidades Docentes	Horas de Cátedra	Horas Docencia Auxiliar	Horas de Trabajo Personal
Requisitos			Carácter del Curso	
CC51A / Autor			Electivo de Especialidad	
Resultados de Aprendizaje				
<p>Resumen:</p> <p>Grandes empresas con una importante necesidad de extender sus software a menudo se encuentran con problemas de mantenimiento. Un software es muy dependiente del hardware sobre el que funciona, de las necesidades de los clientes y del ambiente. Un ejemplo de esto es lo que ocurre con el cambio a un nuevo milenio o con los software de bancos cuando tienen que cambiar una moneda a Euros.</p> <p>Las razones a esta situación son varias: falta de diseño, de unit tests, pobre conocimiento de programación con objetos.</p> <p>Este curso presenta técnicas (reverse engineering, program visualization, refactoring, design, data mining) que son efectivas para re-engineer y evolucionar software industriales.</p> <p>Resultados de Aprendizaje:</p> <p>Los objetivos de este curso son:</p> <ul style="list-style-type: none"> - Hacer del participante un mejor programador - Desarrollar técnicas de diseño - Analizar y discutir sobre el proceso de mantener la calidad de un software - Identificar el problema relacionado a la evolución de sistemas - Realizar una experiencia práctica con el objetivo de conseguir cartas de software que identifiquen las deficiencias de este. 				

Metodología Docente	Evaluación General
Clases expositivas y tareas individuales e incrementales de investigación y programación	La evaluación se basa en una presentación de tareas de investigación y/o programación

Unidades Temáticas

Número	Nombre de la Unidad	Duración en Semanas
1	Diseño, visualización y análisis	5
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<p>Diseño de código</p> <p>Métricas y Detección de Problemas</p> <p>Visualización de Software</p> <p>Lenguaje de programación Pharo</p> <p>Plataforma de Análisis Moose Software</p> <p>CMMI e ISO 9126</p>	<p>Identificar problemas relacionados con la calidad y evolución en sistemas de software.</p>	

Número	Nombre de la Unidad	Duración en Semanas
2	Mejorar la calidad de softwares	5
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<p>Buenas Prácticas</p> <p>Re-estructuración</p> <p>Lenguaje de programación</p>	<p>Conocer las prácticas para mejorar la calidad del software y hacer su evolución más fácil</p>	

Número	Nombre de la Unidad	Duración en Semanas
3	Mining Software Repositories	5
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<p>Que son los "Software repositories"</p> <p>Tipo de datos en Software Repositories</p> <p>Técnicas de Mining Software Repositories (MSR)</p> <p>Poner MSR en práctica</p>	<p>Aprender como utilizar datos contenidos en software repositories para facilitar la evolución de software</p> <p>Aplicar técnicas en casos verdaderos</p>	
Bibliografía		
<p>The Cathedral and the Bazaar – Eric S. Raymond Software Evolution – Background, Theory, Practice – Lehman, Ramil, 2003 http://www.moosetechnology.org/</p>		

Vigencia desde:	Verano 2010
Elaborado por:	Alexandre Bergel & Romain Robbes