GUÍA DOCENTE ABREVIADA DE LA ASIGNATURA

G666 - Diseño de Software

Grado en Ingeniería Informática

Curso Académico 2016-2017

1. DATOS IDENTIFICATIVOS					
Título/s	Grado en Ingeniería Informática			Tipología y Curso	Optativa. Curso 4
Centro	Facultad de Ciencias				
Módulo / materia	MATERIA INGENIERÍA DEL SOFTWARE MENCION EN INGENIERIA DE SOFTWARE MENCIÓN EN INGENIERÍA DEL SOFTWARE				
Código y denominación	G666 - Diseño de Software				
Créditos ECTS	6	Cuatrimestre	Cuatrimestral (1)		
Web	http://moodle.unican.es/moodle2/course/view.php?id=558				
Idioma de impartición	Español	English friendly	No	Forma de	Presencial

Departamento	DPTO. INGENIERÍA INFORMÁTICA Y ELECTRÓNICA
Profesor responsable	PABLO SANCHEZ BARREIRO
E-mail	p.sanchez@unican.es
Número despacho	Facultad de Ciencias. Planta: + 1. DESPACHO PROFESOR (1069)
Otros profesores	

3.1 RESULTADOS DE APRENDIZAJE
- Conocer las principales técnicas y criterios para analizar, diseñar y estructurar software.
- Aplicar las principales notaciones, estrategias y herramientas para el análisis y diseño de software.
- Utilizar patrones software para encontrar buenas soluciones a problemas de diseño y construcción del software.
- Aplicar las principales técnicas de modelado y diseño orientado a objetos, incluidos los aspectos avanzados de UML.

4. OBJETIVOS
El alumno entenderá y sabrá aplicar los principios GRASP y SOLID durante la elaboración de un diseño software.
El alumno entenderá y sabrá aplicar los principios del Diseño por Contrato.
El alumno entenderá el funcionamiento y sabrá aplicar un conjunto de patrones de diseño microarquitectónicos.
El alumno entenderá el funcionamiento y sabrá aplicar un conjunto de patrones de diseño arquitectónicos.
El alumno será capaz de evaluar las ventajas e incovenientes asociados a la aplicación de un determinado patrón de diseño.

6. ORGANIZACIÓN DOCENTE	
CONTENIDOS	
1	Tema 1 - Principios Fundamentales del Diseño Software: Modularidad. Principios GRASP. Principios SOLID. Diseño por Contrato. Modularización utilizando técnicas de Orientación a Aspectos.
2	Tema 2 - Patrones de Diseño Software. Concepto de Patrón. Patrones GoF. Otros patrones no GoF. Inyección de Dependencias. Antipatrones.
3	Tema 3 - Diseño e Implementación de Arquitecturas Software Revisión del Concepto de Arquitectura. Vistas Arquitectónicas. Lenguajes de Descripción Arquitectónica. Evaluación de Arquitecturas Software.
4	Tema 4 - Patrones Arquitectónicos Para Arquitecturas Empresariales Arquitecturas en Capas. Arquitectura Cliente-Servidor. Distribución de Capas en Arquitecturas Empresariales. Patrones para la Capa de Negocio. Patrones para la Capa de Servicio. Patrones para la Capa de Presentación. Patrones para la Capa de Persistencia. Patrones para la Integración de Aplicaciones. Principios de las Arquitecturas Basadas en Microservicios.
5	Tema 5 - Medición de las Propiedades Fundamentales de un Diseño Software. Estabilidad del diseño. Métricas Orientadas a Objetos. Métricas Orientadas a Asuntos.

7. MÉTODOS DE LA EVALUACIÓN

Descripción	Tipología	Eval. Final	Recuper.	%
Desarrollo de Prácticas en Aula	Evaluación en laboratorio	Sí	Sí	40,00
Prueba escrita final	Examen escrito	Sí	Sí	60,00
TOTAL				100,00

Observaciones

La calificación final de la asignatura estará determinada por la media ponderada de las diferentes actividades calificables. Para ello, será condición necesaria superar las calificaciones mínimas exigidas en cada actividad. En caso de no superarse, la calificación será de suspenso. Aquellas partes en las cuales se supere dicha calificación mínima se considerarán como superadas por el estudiante, no siendo necesaria su recuperación en las siguientes convocatorias.

Con objeto de verificar la autoría de las soluciones a los casos prácticos y los informes, y evitar plagios indeseados, el profesor podrá solicitar a los alumnos en cualquier momento pequeñas aclaraciones o modificaciones del material entregado. En caso de que estas aclaraciones o modificaciones hagan sospechar de forma claramente fundada que el alumno no es el autor del material entregado, dicho alumno obtendrá de forma inmediata una calificación de cero en la presente convocatoria.

Para la realización de las pruebas escritas se dejará hacer uso de todo tipo de material impreso, incluyendo apuntes, ejercicios y libros. En ningún caso se dejará hacer uso de portátiles, agendas personales, teléfonos móviles inteligentes u otros dispositivos electrónicos parecidos. Especialmente, queda totalmente prohibido el uso de dispositivos con capacidades de comunicación inalámbricas.

Observaciones para alumnos a tiempo parcial

Dado que no hay actividades de grupo ni actividades que requieran presencialidad, no parecen requerirse medidas especiales para los alumnos matriculados a tiempo parcial. Se procurará que el diseño de las prácticas permita su cómoda realización desde casa en aras de facilitar el trabajo y el seguimiento de la asignatura por parte de estos alumnos.

8. BIBLIOGRAFÍA Y MATERIALES DIDÁCTICOS

BÁSICA

Erich Gamma, Richard Helm, Ralph Johnson y John Vlissides. “Design Patterns: Elements of Reusable Object-Oriented Software”. Marzo 2000.

Craig Larman. “Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development”. 3º Edición. Prentice Hall. Octubre 2004.

Bertrand Meyer. “Construcción de Software Orientado a Objetos”. 2º Edición. Prentice Hall. Marzo 2000.

Robert C. Martin y Micah Martin. “Agile Principles, Patterns and Practices in C#”. Prentice Hall. Julio 2006.

William J. Brown, Raphael C. Malveau, Hays W. "Skip" McCormick y Thomas J. Mowbray. “AntiPatterns: Refactoring Software, Architectures and Projects in Crisis”. Wiley. Abril 1998.

Martin Fowler. "Patterns of Enterprise Application Architecture". Addison-Wesley Professiona. Spetiembre 2002.

Richard Taylor, Nenad Medvidovic y Eric M. Dashofy. “Software Architecture: Foundations, Theory, and Practice”. Wiley. Enero 2009.

Esta es la Guía Docente abreviada de la asignatura. Tienes también publicada en la Web la información más detallada de la asignatura en la Guía Docente Completa.