

## GUÍA DOCENTE

Curso 2013-2014

### 1. DESCRIPCIÓN DE LA ASIGNATURA

<b>Grado:</b>	Ingeniería Informática en Sistemas de Información
<b>Doble Grado:</b>	
<b>Asignatura:</b>	Ingeniería del Software II
<b>Módulo:</b>	Módulo 6
<b>Departamento:</b>	Deporte e Informática
<b>Año académico:</b>	2011-2012
<b>Semestre:</b>	Segundo semestre
<b>Créditos totales:</b>	6
<b>Curso:</b>	2º
<b>Carácter:</b>	Obligatoria
<b>Lengua de impartición:</b>	Español

<b>Modelo de docencia:</b>	C1	
<b>a. Enseñanzas Básicas (EB):</b>		50%
<b>b. Enseñanzas de Prácticas y Desarrollo (EPD):</b>		50%
<b>c. Actividades Dirigidas (AD):</b>		

## GUÍA DOCENTE

Curso 2013-2014

### 2. EQUIPO DOCENTE

#### 2.1. Responsable de la asignatura Norberto Díaz Díaz

2.2. Profesores	
<b>Nombre:</b>	Norberto Díaz Díaz
<b>Centro:</b>	Escuela Politécnica Superior
<b>Departamento:</b>	Departamento de Deportes e Informática
<b>Área:</b>	Área de Lenguajes y Sistemas Informáticos
<b>Categoría:</b>	Profesor Colaborador
<b>Horario de tutorías:</b>	Mediante cita previa: Miércoles de 9:00 a 11:30 y de 16:30 a 20:00
<b>Número de despacho:</b>	Despacho 16, Ed. 11, Primera Planta
<b>E-mail:</b>	ndiaz@upo.es
<b>Teléfono:</b>	9549 77368

## GUÍA DOCENTE

Curso 2013-2014

### 3. UBICACIÓN EN EL PLAN FORMATIVO

#### 3.1. Descripción de los objetivos

La asignatura de Ingeniería del Software 2 (IS2) tiene el objetivo de enseñar a los alumnos a identificar e integrar las diferentes fases del proceso de desarrollo software. Entre ellas se hace hincapié en los procesos de toma de requisitos, análisis de temporal/coste, de análisis y diseño del producto software y la implementación y pruebas de la aplicación desarrollada. Siguiendo la misma línea, los alumnos aprenden los diferentes patrones arquitectónicos software y profundizan en los aspectos positivos/negativos de tomar una solución u otra para cubrir los requisitos no funcionales impuestos por el cliente

#### 3.2. Aportaciones al plan formativo

Enmarcada en el Módulo de Tecnología Específica de Sistemas de Información, la asignatura IS2 afianza las diferentes etapas de un proceso de desarrollos software continuando su formación en materia de Ingeniería del Software iniciada en la asignatura del mismo nombre. Así, la asignatura de IS2 enseña a los alumnos a identificar e integrar las diferentes fases para desarrollar un producto software de calidad. Entre ellas se hace hincapié en los procesos de toma de requisitos, análisis de temporal/coste, de análisis y diseño del producto software y la implementación y pruebas de la aplicación desarrollada. Siguiendo la misma línea, los alumnos aprenden los diferentes patrones arquitectónicos software y profundizan en los aspectos positivos/negativos de tomar una solución u otra para cubrir los requisitos no funcionales impuestos por el cliente.

Desde un punto de vista más amplio, la asignatura IS2 sería una ampliación en materia de ingeniería del software y estaría basada en la formación del alumnado en materia de programación.

En conjunción con las asignaturas del subbloque Sistemas de Información, IS2 aporta los conceptos teórico/prácticos para identificar las etapas y artefactos a llevar a cabo para obtener un producto software de calidad. Esta asignatura sería, por tanto, el punto de encuentro del resto de asignaturas de tal submódulo (Ingeniería de Proyectos, Calidad y Seguridad).

Atendiendo al segundo submódulo, Gestión de Base de Datos, podría decirse que tal submódulo es básico, ya que en IS2 se trata el problema de acceso a datos.



## GUÍA DOCENTE

Curso 2013-2014

### **3.3. Recomendaciones o conocimientos previos requeridos**

Es recomendable que el alumno tenga superada la asignatura de Ingeniería del Software I.

## GUÍA DOCENTE

Curso 2013-2014

### 4. COMPETENCIAS

#### 4.1 Competencias de la Titulación que se desarrollan en la asignatura

4.1.1. Capacidad para concebir, redactar, organizar, planificar, desarrollar y firmar proyectos en el ámbito de la ingeniería en informática que tengan por objeto, de acuerdo con los conocimientos adquiridos, la concepción, el desarrollo o la explotación de sistemas, servicios y aplicaciones informáticas. (Competencia G01)

4.1.2. Capacidad para dirigir las actividades objeto de los proyectos del ámbito de la informática. (Competencia G02)

4.1.3. Capacidad para dirigir las actividades objeto de los proyectos del ámbito de la informática. (Competencia G03)

4.1.4. Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones informáticas empleando los métodos de la ingeniería del software como instrumento para el aseguramiento de su calidad. (Competencia G05)

4.1.5. Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería. (Competencia EB5)

4.1.6. Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados. (Competencia EC08)

4.1.7. Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software. (Competencia EC16)

4.1.8. Capacidad para diseñar y evaluar interfaces persona computador que garanticen la accesibilidad y usabilidad a los sistemas, servicios y aplicaciones informáticas. (Competencia EC17)

#### 4.2. Competencias del Módulo que se desarrollan en la asignatura

4.2.1. Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social. (Competencia EC02)

4.2.2. Capacidad para comprender la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software. (Competencia EC03)

## GUÍA DOCENTE

Curso 2013-2014

### **4.3. Competencias particulares de la asignatura**

4.3.1. Enseñar al alumno el papel fundamental que desempeña el análisis dentro del proceso software. Comprender la importancia de identificar requisitos y realizar una gestión disminuya los riesgos de su desarrollo.

4.3.2. Enseñar al alumno las técnicas para el análisis de problemas mediante técnicas descriptivas de modelado. Usar y elegir correctamente un tipo de proceso de desarrollo software dependiendo de los requisitos a cumplir.

4.3.3. Fomentar la disciplina en la programación, siguiendo planteamientos que desarrollen la capacidad analítica del alumno para enfrentarse a problemas reales. Ser consciente de las ventajas de identificar requisitos software en etapas tempranas del ciclo de vida software y su utilidad en el proceso de desarrollo software.



## GUÍA DOCENTE

Curso 2013-2014

### 5. CONTENIDOS DE LA ASIGNATURA (TEMARIO)

- Tema 1: Introducción
- Tema 2: Diagramas de Comportamiento
- Tema 3: Relación entre Casos de Uso
- Tema 4: Ampliación del Modelo de Dominio
- Tema 5: Patrones de Diseño
- Tema 6: Arquitectura Software
- Tema 7: Diseño de Arquitectura Lógica
- Tema 8: Patrón MVC
- Tema 9: Organización de Paquetes

## GUÍA DOCENTE

Curso 2013-2014

### 6. METODOLOGÍA Y RECURSOS

Los conceptos principales se explican y debaten en clase de Enseñanzas Básicas cuya documentación está disponible con suficiente antelación en la página Web de la asignatura. Se trata de clases dinámicas donde los estudiantes y profesores interactúan debatiendo aspectos importantes de los ejemplos y ejercicios que en estas clases se proponen.

Además el estudiante dispone con antelación en la página Web de la asignatura de la documentación de las Enseñanzas Práctica y de Desarrollo correspondiente que se lleva a cabo para una mejor comprensión de los conceptos desarrollados en la clase de Enseñanzas Básicas. En dicha documentación se introduce al alumno en la resolución de problemas mediante unos experimentos guiados, y se enumeran una serie de ejercicios y problemas que el alumno debe resolver en clase y en su casa. Esta actividad permite el trabajo en pequeños grupos de dos o tres alumnos, o de manera individual. Se realizan con grupos inferiores a 15 alumnos, por lo que el seguimiento del profesor es cercano al trabajo realizado por el alumno, pudiendo comprobar la evolución del trabajo realizado por cada uno.

En general, siempre se pretende la integración de los procedimientos y modalidades didácticas de la enseñanza presencial con los de la enseñanza virtual. La asignatura dispone de una página Web de comunicación fácil y amigable que favorece los procesos de intercambio de información y la comunicación de experiencias. Además del espacio para colgar el material necesario para las clases (transparencias de teoría, boletines de prácticas, soluciones de los ejercicios, notas, etc.) y los anuncios de los distintos eventos que acontezcan, esta página dispone de espacios reservados para el foro de la asignatura. En el foro se debaten algunos temas que los alumnos desean y otros propuestos por los profesores. Se fomenta el uso de las tutorías a través del foro y correo electrónico y se fomenta la plataforma WebCT disponible en nuestra Universidad.



## GUÍA DOCENTE

Curso 2013-2014

### 7. EVALUACIÓN

#### CONVOCATORIA DE JUNIO:

La evaluación de la asignatura se acogerá al modelo de evaluación continua y sólo será aplicable para la convocatoria de junio (1<sup>TM</sup> convocatoria de curso). Aquellos alumnos que no superen la asignatura en dicha convocatoria, se podrán evaluar en la convocatoria de julio como se describe más adelante.

La evaluación se basará principalmente en los conocimientos adquiridos tanto en clase de teoría como en laboratorio de informática. La nota final oscilará entre 0 y 10 puntos, los cuales se acumularán en función de la calificación de las pruebas evaluables y porcentajes descritos en el apartado 1 de la presente memoria:

- Pruebas teóricas: 50%
- Pruebas prácticas: 50%

Las pruebas prácticas, que serán individuales o grupales y obligatorias, consistirán en la resolución de diferentes ejercicios en los laboratorios de informática. La nota correspondiente a esta parte se calculará mediante la media ponderada de las calificaciones obtenidas en cada una de las pruebas.

Las pruebas teóricas consistirán en la resolución de cuestiones y problemas sobre los contenidos de la asignatura.

Para poder calcular la nota final, el alumno deberá obtener al menos 1 punto sobre el cómputo de 10 en la parte de pruebas prácticas y al menos 1 sobre el cómputo de 10 en las pruebas teóricas.

Para la realización de cualquier prueba evaluable no se permitirá el uso o consulta de documentación, salvo indicación expresa del profesor en convocatoria oficial.

#### CONVOCATORIA DE JULIO:

Para la convocatoria de julio, la asignatura se evaluará mediante una prueba escrita que integre la materia correspondientes a Enseñanzas Básicas y Enseñanzas Prácticas y de Desarrollo desarrolladas a lo largo de la asignatura. Las notas obtenidas en las pruebas evaluables durante la evaluación continua no se guardarán para la convocatoria de julio ni posibles convocatorias extraordinarias.



## GUÍA DOCENTE

Curso 2013-2014

### 8. BIBLIOGRAFÍA GENERAL

1. Applying UML and patterns : an introduction to object-oriented analysis and design and iterative development (3rd edition). Craig Larman. Prentice Hall, 2008.
2. UML y patrones : una introducción al análisis y diseño orientado a objetos y al proceso unificado (2a edición, traducida). Craig Larman. Prentice Hall, 2004.