



Guía docente				
Datos Identificativos				2013/14
Asignatura (*)	Diseño de Sistemas de Información		Código	614111403
Titulación	Enxeñeiro en Informática			
Descritores				
Ciclo	Periodo	Curso	Tipo	Créditos
1º y 2º Ciclo	2º cuatrimestre	Cuarto	Troncal	6
Idioma	Castellano			
Prerrequisitos				
Departamento	Computación			
Coordinador/a	Castro Souto, Laura Milagros	Correo electrónico	laura.milagros.castro.souto@udc.es	
Profesorado	Castro Souto, Laura Milagros	Correo electrónico	laura.milagros.castro.souto@udc.es	
Web	campusvirtual.udc.es			
Descripción general	Diseño, propiedades e mantemento do software.			

Competencias de la titulación	
Código	Competencias de la titulación
A1	Aprender de manera autónoma nuevos conocimientos y técnicas avanzadas adecuadas para la investigación, el diseño y el desarrollo de sistemas y servicios informáticos.
A3	Concebir y planificar el desarrollo de aplicaciones informáticas complejas o con requisitos especiales.
A6	Evaluar, definir, seleccionar y auditar plataformas hardware y software para la ejecución y desarrollo de aplicaciones y servicios informáticos.
A7	Saber especificar, diseñar e implementar un sistema de información, empleando bases de datos.
A9	Dirigir equipos de trabajo ligados al diseño de productos, procesos, servicios informáticos y otras actividades profesionales.
A10	Saber especificar, diseñar e implementar una política de seguridad en el sistema.
B1	Aprender a aprender.
B2	Resolver problemas de forma efectiva.
B3	Aplicar un pensamiento crítico, lógico y creativo.
B4	Aprendizaje autónomo.
B5	Trabajar de forma colaborativa.
B7	Comunicarse de manera efectiva en cualquier entorno de trabajo.
B8	Trabajar en equipos de carácter interdisciplinar.
B9	Capacidad para tomar decisiones.
B10	Capacidad de gestión de la informática (captación y análisis de la información).
B11	Razonamiento crítico.
B12	Capacidad para el análisis y la síntesis.
B13	Capacidad de comunicación.
B14	Conocimiento de idiomas.
B15	Motivación por la calidad.
C3	Utilizar las herramientas básicas de las tecnologías de la información y las comunicaciones (TIC) necesarias para el ejercicio de su profesión y para el aprendizaje a lo largo de su vida.
C6	Valorar críticamente el conocimiento, la tecnología y la información disponible para resolver los problemas con los que deben enfrentarse.
C7	Asumir como profesional y ciudadano la importancia del aprendizaje a lo largo de la vida.
C8	Valorar la importancia que tiene la investigación, la innovación y el desarrollo tecnológico en el avance socioeconómico y cultural de la sociedad.

Resultados de aprendizaje	
Competencias de materia (Resultados de aprendizaje)	Competencias de la titulación



Conocer los conceptos y técnicas propios de la Ingeniería del Software.	A3 A7 A9	B2 B9 B12 B15	C3 C6 C7 C8
Manejar y adaptar las herramientas de diseño y construcción de aplicaciones.	A1 A3 A6 A7 A10	B1 B2 B3 B4 B5 B7 B8 B9 B10 B11 B14 B15	C3 C6 C7 C8
Aplicar con agilidad los lenguajes de modelado y los patrones de diseño.	A3 A7	B2 B3 B7 B9 B10 B11 B12 B15	C3 C6 C7 C8
Interpretar las situaciones en que se presentan los problemas típicos de diseño.	A1 A3 A6 A7	B1 B2 B3 B4 B5 B7 B8 B9 B10 B11 B12 B13	C3 C6 C7 C8
Sintetizar los casos de éxito propios.	A1 A3 A6 A7 A9	B1 B3 B4 B7 B9 B10 B11 B12 B13 B15	C3 C6 C7 C8

Contenidos	
Tema	Subtema



Introducción al DSI	a) Definición y objetivos b) El diseño como fase en el proceso de desarrollo c) Principios básicos de diseño d) Introducción a las arquitecturas de software e) El lenguaje de modelado unificado UML f) El modelo de vista de la arquitectura 4+1
Estrategias de diseño	a) Conceptos fundamentales b) Diseño estructurado c) Diseño orientado a objetos
Arquitecturas del software	a) Conceptos fundamentales b) El patrón Modelo/Vista/Controlador c) Diseño por capas
Introducción a los patrones de diseño	a) Concepto de patrón de diseño b) Composicionalidad (Composición)
Asignación de responsabilidades y delegación	a) Asignación de responsabilidades (Proxy, Decorador) b) Delegación (Decorador, Cadena de Responsabilidad)
Bajo acoplamiento, instanciación, alta cohesión y división en subsistemas	a) Bajo acoplamiento e instanciación (Prototipo, Negociador, Instancia Única) b) Alta cohesión y división en subsistemas (Fachada)
Modelado de comportamiento dinámico	a) Modelado de comportamiento dinámico (Estado)
Reutilización	a) Reutilización (Estrategia, Iterador)
Dependencias entre objetos	a) Dependencias entre objetos (Observador)
Refactorización del diseño y preparación para el cambio	a) Refactorización del diseño (Método de Fabricación, Método Plantilla) b) Preparación para el cambio (Visitante)
Encapsulación y abstracción	a) Encapsulación y abstracción (Comando, Recuerdo)
Adaptación de interfaces y manejo de la complejidad	a) Adaptación de interfaces (Adaptador, Puente, Fábrica Abstracta) b) Manejo de la complejidad (Mediador)
Interfaces de usuario	a) Conceptos fundamentales de interfaces de usuario b) Interfaces en aplicaciones web c) Dependencias entre objetos, encapsulación, abstracción, delegación y división en subsistemas.
Patrones de persistencia y concurrencia	a) Introducción

**Planificación**

Metodologías / pruebas	Horas presenciales	Horas no presenciales / trabajo autónomo	Horas totales
Sesión magistral	21	21	42
Seminario	10	15	25
Prácticas de laboratorio	15	15	30
Prueba objetiva	5	0	5
Trabajos tutelados	0	24	24
Atención personalizada	24	0	24

(\*) Los datos que aparecen en la tabla de planificación són de carácter orientativo, considerando la heterogeneidad de los alumnos

**Metodologías**

Metodologías	Descripción
Sesión magistral	Clases magistrales en la exposición de los conocimientos teóricos utilizando diferentes recursos: pizarra, proyección de material en formato electrónico, apuntes en formato electrónico y recursos facilitados por el docente de la asignatura en la facultad virtual.



Seminario	Seminarios prácticos en los que se desarrollan los conocimientos adquiridos, tomando un ejemplo concreto como hilo conductor. El docente de la asignatura dirige la realización del seminario, incentivado en la medida de lo posible la participación del estudiantado.
Prácticas de laboratorio	Prácticas diseñadas por el docente de la asignatura basadas en los conocimientos que el estudiante va adquiriendo. Los estudiantes desarrollarán estos trabajos en pequeños grupos, preferiblemente de dos personas.
Prueba objetiva	Prueba escrita mediante a que se valoran los conocimientos adquiridos por el estudiante. El estudiante deberá desarrollar tanto sus conocimientos a nivel teórico mediante la respuesta a preguntas, como sus conocimientos aplicados mediante la resolución de ejercicios y problemas similares a los planteados en las prácticas de laboratorio y trabajos tutelados.
Trabajos tutelados	Trabajos tutelados propuestos por el docente de la asignatura y desarrollados por los estudiantes, bien en grupo, bien individualmente. Estos trabajos tutelados podrán ser evaluados mediante controles tipo test o en las tutorías personalizadas.

## Atención personalizada

Metodologías	Descripción
Prácticas de laboratorio Prueba objetiva Trabajos tutelados	<p>La atención personalizada al estudiantado comprende no sólo las clásicas tutorías, presenciales o virtuales, para la discusión de dudas, sino también las siguientes actuaciones:</p> <ul style="list-style-type: none"><li>- Seguimiento de la labor realizada en los trabajos tutorizados propuestos por el docente.</li><li>- Evaluación de los resultados obtenidos en los trabajos y prácticas realizadas por el estudiante.</li><li>- Discusión con el fin de solucionar los problemas encontrados por el estudiante en la realización de los diversos trabajos de carácter tanto teórico como práctico.</li></ul>

## Evaluación

Metodologías	Descripción	Calificación
Prácticas de laboratorio	<p>Evaluación de las prácticas realizadas de forma continua a lo largo del curso en una defensa final. A pesar de que las prácticas son realizadas en grupos, cabe diferenciar dos componentes en la evaluación de las prácticas de un alumno:</p> <ul style="list-style-type: none"><li>- Valoración del trabajo en grupo, en la que se califica el grado de coordinación y compenetración entre los miembros.</li><li>- Valoración personal, en la que se califica la aportación concreta del alumno al grupo de prácticas.</li></ul> <p>Entre los aspectos a considerar a la hora de valorar las prácticas, se encuentran:</p> <ul style="list-style-type: none"><li>- Rigor en la consecución de los objetivos perseguidos en la práctica usando las técnicas propuestas en la asignatura.</li><li>- Asimilación de los conceptos perseguidos por la práctica.</li><li>- Originalidad en las propuestas acometidas durante la realización de la práctica.</li><li>- Responsabilidad en la entrega de las prácticas en tiempo y forma, así como en el uso apropiado de los recursos habilitados al efecto.</li></ul>	30
Prueba objetiva	Prueba escrita dividida en tres bloques: cuestiones cortas, pequeñas extensiones de diseño e implementación y un problema de diseño haciendo hincapié en la justificación de las decisiones tomadas.	60



Trabajos tutelados	Se valorarán los siguientes aspectos: <ul style="list-style-type: none"><li>- Dominio de los contenidos de los trabajos académicos propuestos.</li><li>- Dominio de los conocimientos teóricos y operativos de la materia.</li><li>- Seguimiento apropiado del desarrollo de la asignatura.</li></ul>	10
Otros		

### Observaciones evaluación

### Fuentes de información

<b>Básica</b>	<ul style="list-style-type: none"><li>- Gamma, E.; Helm, R.; Johnson, R. y Vlissides J. (1996). Design Patterns: Elements of Reusable Object-oriented Software. Addison Wesley</li><li>- Arnold K., Gosling J. y Holmes D. (2005). The Java Programming Language. Prentice Hall</li><li>- Rumbaugh, J.; Jacobson, I. y Booch, J. (2004). The Unified Modeling Language Reference Manual. Addison Wesley</li><li>- Booch J.; Rumbaugh J. y Jacobson I. (2005). The Unified Modeling Language User Guide. Addison Wesley</li></ul>
<b>Complementaria</b>	<ul style="list-style-type: none"><li>- Grand M. (2002). Patterns in Java. Volumen 1. John Wiley &amp; Sons</li><li>- Page-Jones, M. (2004). Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. Prentice Hall PTR</li><li>- Cooper J. (2000). Java Design Patterns: A Tutorial. Addison Wesley</li><li>- Stevens, P. y Pooley, R. (1999). Using UML. Software Engineering with Objects and Components. Addison Wesley</li></ul>

### Recomendaciones

#### Asignaturas que se recomienda haber cursado previamente

Integración de Sistemas/614111503

#### Asignaturas que se recomienda cursar simultáneamente

Análisis y Diseño Orientado a Objetos/614111602

Interfaces con el Usuario/614111624

Programación Orientada a Objetos/614111636

#### Asignaturas que continúan el temario

Análisis de Sistemas Informáticos/614111402

### Otros comentarios

(\*) La Guía Docente es el documento donde se visualiza la propuesta académica de la UDC. Este documento es público y no se puede modificar, salvo cosas excepcionales bajo la revisión del órgano competente de acuerdo a la normativa vigente que establece el proceso de elaboración de guías