UNIVERSIDADE DA CORUÑA

Guia docente					
Datos Identificativos					2017/18
Asignatura (*)	Diseño Software			Código	614G01015
Titulación	Grao en Enxeñaría Informática				
Descriptores					
Ciclo	Periodo	Curso		Tipo	Créditos
Grado	1º cuatrimestre	Segundo		Obligatoria	6
Idioma	CastellanoInglés				
Modalidad docente	Presencial				
Prerrequisitos					
Departamento	Computación				
Coordinador/a	Mosqueira Rey, Eduardo		Correo electrónico	eduardo.mosqueira@udc.es	
Profesorado	Alonso Ríos, David Cabrero Canosa, Mariano Javier Monroy Camafreita, Juan Mosqueira Rey, Eduardo Paz López, Alejandro Pérez Sánchez, Beatriz Sanchez Maroño, Noelia		Correo electrónico	david.alonso@udc.es mariano.cabrero@udc.es juan.monroy@udc.es eduardo.mosqueira@udc.es alejandro.paz.lopez@udc.es beatriz.perezs@udc.es noelia.sanchez@udc.es	
Web					
Descripción general	El Diseño Software es una fase clave dentro del ciclo de vida del software que establece el enlace entre los requisitos de un sistema y su posterior implementación. El diseño más habitual hoy en día es el diseño basado en la orientación a objetos, que consiste en desarrollar un programa en base a objetos que intercambian mensajes. Esta asignatura introducirá a los alumnos en los elementos y propiedades básicas de la orientación a objetos usando un lenguaje orientado a objetos como Java. Se aprenderá también a cómo reflejar los artefactos propios del diseño en un lenguaje de modelado como el Lenguaje Unificado de Modelado (UML). Finalmente se presentarán aquellos principios básicos que representan un buen diseño y se aprenderá a identificar aquellos problemas típicos de diseño y sus soluciones más comunes representadas como patrones de diseño.				

Competencias / Resultados del título	
Código	Competencias / Resultados del título
A7	Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.
A13	Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.
A14	Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
B1	Capacidad de resolución de problemas
B2	Trabajo en equipo
B3	Capacidad de análisis y síntesis
B4	Capacidad para organizar y planificar
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.

Resultados de aprendizaje	
Resultados de aprendizaje	Competencias / Resultados del título

Identificar el diseño software como una de las fases del ciclo de vida del software	A7 A13 A14	B3 B4	C3
Conocer los principios y propiedades básicas de la orientación a objetos	A7 A13 A14	B1 B2 B3 B4	C3 C6
Plasmar un diseño software utilizando los artefactos propios de un lenguaje de modelado como UML	A7 A13 A14	B1 B2 B3 B4	C3 C6
Conocer los principios básicos que representan un buen diseño software	A7 A13 A14	B1 B2 B3 B4	C3 C6
Identificar problemas típicos de diseño y sus soluciones más comunes	A7 A13 A14	B1 B2 B3 B4	C3 C6
Usar un diseño como guía para la implementación del software	A7 A13 A14	B1 B2 B3 B4	C3 C6
Aprender un lenguaje orientado a objetos y aspectos relacionados (IDE, pruebas, repositorios, etc.)	A13	B1 B2 B3 B4	C3 C6

Contenidos	
Tema	**Subtema**
1. Introducción	? Diseño software ? Diseño orientado a objetos
2. Elementos Básicos de la Orientación a Objetos	? Clases y objetos ? Identidad de objetos ? Estado de objetos ? Comportamento de objetos
3. Propiedades Básicas de la Orientación a Objetos	? Abstracción y encapsulamiento ? Modularidad ? Jerarquía ? Polimorfismo ? Tipificación ? Ligadura dinámica
4. Lenguaje Unificado de Modelado (UML)	? Introducción ? Bloques básicos del UML ? Diseño estático: Diagramas de clases ? Diseño dinámico: Diagramas de interacción ? Otros diagramas
5. Principios de Diseño	? Calidad en el diseño ? Principios SOLID ? Tipos de herencia

6. Patrones de Diseño	? Introducción a los patrones de diseño ? Patrones elementales ? Diseños adaptables a los cambios ? Diseños débilmente acoplados ? Patrones y colecciones de objetos ? Otros patrones y principios
Prácticas	? Introducción a Java y NetBeans ? Pruebas de software ? Gestión de excepciones ? Uso de un repositorio de código

Planificación				
Metodologías / pruebas	Competencias / Resultados	Horas lectivas (presenciales y virtuales)	Horas trabajo autónomo	Horas totales
Sesión magistral	A7 A13 A14 B1 B3 C6	30	45	75
Prácticas de laboratorio	A7 A13 A14 B1 B2 B3 B4 C3 C6	20	30	50
Seminario	A7 A13 A14 B1 B2 B3 B4 C3 C6	10	10	20
Prueba objetiva	A7 A13 A14 B1 B3 C6	3	0	3
Atención personalizada		2	0	2
(*)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 usando diferentes recursos: pizarra, proyección de material en formato electrónico, apuntes en formato electrónico y los recursos facilitados por el docente de la asignatura en el Campus Virtual de la UDC.
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 preferiblemente en grupo (que serán como máximo de dos personas). Se utilizará una herramienta de modelado para construir los artefactos de diseño y se empleará un lenguaje orientado a objetos (Java) para realizar la implementación de los mismos.
Seminario	Seminarios en los que se propondrán actividades relacionadas con los conocimientos adquiridos en teoría o práctica
Prueba objetiva	Prueba escrita mediante la que se valora los conocimientos adquiridos por el estudiante. Cada estudiante deberá aplicar sus conocimientos tanto a nivel teórico como a nivel práctico.

Atención personalizada	
Metodologías	Descripción

Prácticas de laboratorio Seminario	La atención personalizada al estudiante comprende no solo las tutorías, presenciales o virtuales, para la discusión de dudas, sino también las siguientes actuaciones: - Seguimiento de la labor realizada en las prácticas de laboratorio propuestos por el docente. - Evaluación de los resultados obtenidos en las prácticas, participación en seminarios realizados por el estudiante. - Encuentros personalizados para resolver dudas sobre los contenidos de la asignatura.

Evaluación			
Metodologías	**Competencias / Resultados**	**Descripción**	**Calificación**
Prácticas de laboratorio	A7 A13 A14 B1 B2 B3 B4 C3 C6	Dos boletines de ejercicios basados en la programación en Java, en la orientación a objetos y el diseño de pruebas. Una práctica de diseño centrada en el uso de principios y patrones de diseño. Practicas copiadas podrán suponer un cero en la nota, tanto para el original como para la copia	40
Seminario	A7 A13 A14 B1 B2 B3 B4 C3 C6	Los seminarios son prácticas de laboratorio desarrolladas por los alumnos con la ayuda directa del profesor que, al final de los mismos, mostrará cuál era la solución a la que los alumnos tenían que llegar. Los seminarios están directamente relacionados con la teoría y la práctica y su evaluación se realizará en estos apartados	0
Prueba objetiva	A7 A13 A14 B1 B3 C6	Prueba escrita realizada al final del curso sobre contenidos teórico-prácticos. La prueba objetiva es obligatoria para aprobar la asignatura y también es obligatorio obtener una nota mínima de 4 para poder hacer media con los otros elementos evaluables.	60

Observaciones evaluación
En caso de no llegar a la nota mínima, en cualquiera de las oportunidades, implicará que no se pueda obtener más de un 4,5 en la nota final de la materia. Aspectos a tener en cuenta para la evaluación de segunda oportunidad (Julio): La nota de las prácticas de laboratorio serán las obtenidas en la primera oportunidad (no habrá entrega de prácticas en la segunda oportunidad).Aspecto a tener en cuenta en caso de matrícula a tiempo parcial: Se elimina la obligatoriedad de asistir a las actividades en las que se pueda exigir presencialidad.

Fuentes de información	
Básica	- Sierra, K., Bates, B. (2005). Head First Java. O?Reilly - Eckel, B. (2007). Piensa en Java (4ª ed.). Thinking in Java (4th ed.). Prentice-Hall - Booch J.; Rumbaugh J. y Jacobson I. (2006). El Lenguaje Unificado de Modelado (2ª ed.) The Unified Modeling Language (2nd ed.). Addison Wesley - Martin, R.C. (2004). UML para programadores Java. UML for Java Programmers. Pearson - Gamma, E.; Helm, R.; Johnson, R. y Vlissides J. (1996). Design Patterns: Elements of Reusable Object-oriented Software.. Addison Wesley

Complementária	- Arnold K., Gosling J. y Holmes D. (2005). The Java Programming Language. Prentice-Hall - Rumbaugh, J.; Jacobson, I. y Booch, J. (2004). The Unified Modeling Language Reference Manual. Addison Wesley - Stevens, P. y Pooley, R. (2006). Using UML. Software Engineering with Objects and Components. Addison Wesley - Freeman, E., Freeman, E., Bates, B. (2004). Head First Design Patterns. O'Reilly - Grand M. (2002). Patterns in Java. John Wiley & Sons

Recomendaciones
Asignaturas que se recomienda haber cursado previamente
Programación I/614G01001 Programación II/614G01006
Asignaturas que se recomienda cursar simultáneamente
Paradigmas de Programación/614G01014
Asignaturas que continúan el temario
Proceso Software/614G01019 Interfaces Hombre Máquina/614G01022 Internet y Sistemas Distribuidos/614G01023
Otros comentarios
La asignatura asume que los estudiantes saben programar y conocen las estructuras de datos (Programación II) aunque nunca han utilizado un lenguaje orientado a objetos. Al principio del curso, y según se van introduciendo los conceptos propios de la orientación a objetos, los alumnos se familiarizan con los aspectos básicos del lenguaje de programación Java.

(*) 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