



Guía Docente				
Datos Identificativos				2015/16
Asignatura (*)	Metodoloxía da Programación	Código	614111205	
Titulación				
Descritores				
Ciclo	Período	Curso	Tipo	Créditos
1º e 2º Ciclo	1º cuatrimestre	Segundo	Obrigatoria	6
Idioma	Castelán			
Modalidade docente	Presencial			
Prerrequisitos				
Departamento	Tecnoloxías da Información e as Comunicaci3ns			
Coordinaci3n		Correo electr3nico		
Profesorado		Correo electr3nico		
Web				
Descrici3n xeral	<p>Esta asignatura se centra en uno de los principales paradigmas existentes para el desarrollo de sistemas software: la Orientaci3n a Objetos. M3s concretamente, el alumno aprender3 las diferentes fases que esta aproximaci3n conlleva, as3 como las diferentes t3cnicas y herramientas con las que cuenta.</p> <p>Finalmente, se destaca la importancia de la documentaci3n y las pruebas en el conjunto del producto software.</p>			

Competencias / Resultados do t3tulo	
C3digo	Competencias / Resultados do t3tulo

Resultados da aprendizaxe			
Resultados de aprendizaxe	Competencias / Resultados do t3tulo		
Conocer m3todos, t3cnicas y herramientas de desarrollo de software Orientado a Objetos.	A7	B2 B12	C3
Capacidad para aplicar los anteriores elementos a casos pr3cticos.		B2 B3	
Capacidad para adaptarse a nuevas situaciones y contextos.	A1	B1 B2 B3 B11 B12	C3 C6
Capacidad para trabajar en equipo.	A9	B5 B7 B8 B9 B13	
Capacidad para aprender de forma aut3noma.	A1	B1 B3 B4 B11	

Contidos	
Temas	Subtemas
Introducci3n a la Ingenier3a del Software	<p>Antecedentes hist3ricos</p> <p>El proceso de resoluci3n de problemas</p> <p>Introducci3n a los ciclos de vida</p> <p>Introducci3n a las metodolox3as de desarrollo software</p>



Conceptos básicos de la Orientación a Objetos	Introducción Objetos Clases Relaciones Polimorfismo Ejemplos básicos
Análisis y diseño Orientado a Objetos	Análisis textual de Abbot Técnica CRC Estrategias de análisis
Notación UML	Elementos comunes a los diagramas Diagramas de clases Diagramas de casos de uso Diagramas de secuencia Diagramas de colaboración Diagramas de estado
Patrones de Diseño Orientado a Objetos	Introducción Ventajas Principales patrones
Práctica de Programación Orientado a Objetos en JAVA	Aprendizaje autónomo del lenguaje de programación JAVA Programación en JAVA de un sistema software
Trabajo en grupo: Análisis y diseño de un sistema software	Planteamiento de un dominio de aplicación Elaboración del diagrama de casos de uso Elaboración del diagrama de clases asociado Elaboración, para las principales funcionalidades del sistema, de los diagramas de secuencia, colaboración y de estado que se consideren necesarios. Seguimiento y tutorización del trabajo en grupo

### Planificación

Metodoloxías / probas	Competencias / Resultados	Horas lectivas (presenciais e virtuais)	Horas traballo autónomo	Horas totais
Sesión maxistral		36	36	72
Traballos tutelados		0	30	30
Proba obxectiva		4	10	14
Prácticas de laboratorio		16	16	32
Atención personalizada		2	0	2

\*Os datos que aparecen na táboa de planificación son de carácter orientativo, considerando a heteroxeneidade do alumnado

### Metodoloxías

Metodoloxías	Descrición
Sesión maxistral	Presentación de los conocimientos teóricos relacionados con los temas abordados.
Traballos tutelados	Trabajo en grupo en el que se aplican los conocimientos adquiridos a un caso práctico propuesto por el grupo.
Proba obxectiva	Examen escrito para valorar los conocimientos teóricos y prácticos adquiridos a lo largo del curso.
Prácticas de laboratorio	Análisis, diseño, implementación y discusión de las soluciones planteadas para un caso práctico común.

### Atención personalizada

Metodoloxías	Descrición
--------------	------------



Traballos tutelados	<p>Se combinará el método magistral, las clases de prácticas, el trabajo autónomo y el trabajo en grupo.</p> <p>El método magistral se empleará para la presentación de los conocimientos teóricos relacionados con los distintos bloques temáticos.</p> <p>Las clases de prácticas se dedicarán a la realización de ejemplos y ejercicios, que serán discutidos con los alumnos con objeto de corregir errores cometidos. Para ello, el trabajo autónomo del alumno es fundamental.</p> <p>Finalmente, el trabajo en grupo permite a los alumnos enfrentarse con los problemas inherentes al análisis y diseño de un sistema software. Para el seguimiento y tutorización de este trabajo el grupo se reunirá con el tutor en diferentes ocasiones para valorar y asegurar el progreso en la adquisición y aplicación de los conocimientos adquiridos.</p> <p>En todo momento se fomentará la participación de los alumnos.</p>
---------------------	--

Avaliación			
Metodoloxías	Competencias / Resultados	Descrición	Cualificación
Traballos tutelados		Trabajo tutelado en grupo en el que se valorará: - Dominio de los conocimientos prácticos de la materia. - Calidad de la solución aportada. - Defensa del trabajo. - Dominio de los conocimientos adquiridos.	30
Proba obxectiva		Examen escrito individual sobre el dominio de los conocimientos teóricos y prácticos de la asignatura; correspondiendo el 45% del total a la parte teórica de la prueba y el 55% restante a la parte práctica de la misma.	70
Prácticas de laboratorio		La correcta realización de las prácticas de laboratorio es obligatoria, pero no computa a efecto de la nota final; siendo "apto" o "no apto"; las posibles calificaciones de las mismas.	0
Outros			

### Observacións avaliación



La evaluación se realizará tanto desde un punto de vista formativo como de acreditación.

El primer aspecto se refiere a la obtención de la información necesaria para determinar como se va desarrollando el proceso formativo, establecer la calidad de los aprendizajes del alumnado y aplicar las medidas correctivas necesarias para cubrir las lagunas de aprendizaje detectadas. Para ello, a lo largo del curso se analizará la evolución de cada alumno mediante su participación en las clases teóricas, prácticas, así como en el trabajo en grupo obligatorio.

Por su parte, el objetivo de la acreditación es constatar que los alumnos poseen las competencias fundamentales necesarias para llevar a cabo el desarrollo de un sistema software desde el punto de vista Orientado a Objetos. Esta acreditación se realizará en dos momentos temporales distintos: una vez finalizado el trabajo en grupo, mediante su defensa y al finalizar el curso, mediante un examen escrito individual teórico y uno práctico.

La nota final de cada alumno se obtendrá según lo indicado a continuación:

Examen escrito individual: 70% (correspondiendo el 45% a la parte teórica y el 55% a la parte práctica)

Trabajo en grupo: 30%

Para aprobar la asignatura es preciso obtener una puntuación global mínima de 5 puntos sobre 10 y cumplir las siguientes restricciones:

- Haber superado las diferentes prácticas obligatorias planteadas por el profesor a lo largo del curso. Estas prácticas son obligatorias, pero no computan a efecto de la nota final.
- Tener una nota mínima de 5 sobre 10 en el trabajo en grupo.
- Tener una nota mínima de 5 sobre 10 en el examen (sumando parte teórica y práctica).

Aspectos a tener en cuenta:

- Los grupos de alumnos para realizar el trabajo en grupo se formarán bajo las directrices del profesor cuando éste así lo indique al principio del curso.
- La defensa del trabajo en grupo será en horario oficial de la asignatura y a estos actos deberán asistir todos los miembros del grupo.
- Una vez aprobado el trabajo en grupo, su nota se mantendrá para futuras convocatorias mientras no varíe el formato del trabajo.

Los alumnos que no superen la asignatura tendrán que demostrar la correcta adquisición de las competencias fundamentales de la asignatura mediante la realización de un nuevo examen teórico y práctico. Además, aquellos alumnos que no hayan superado el trabajo en grupo y, o, las prácticas obligatorias deberán rehacerlos bajo las directrices del profesor hasta que éstos cumplan con los requisitos mínimos exigidos.

### Fontes de información

<b>Bibliografía básica</b>	- Pressman, Roger S. Ingeniería del Software. Un Enfoque Práctico. McGraw Hill. 6 Edición. 2005. - Booch, G., Rumbaugh, J., Jacobson, I. El Lenguaje Unificado de Modelado. Addison Wesley Iberoamericana. 2004. - Sommerville, Ian. Software Engineering. Addison-Wesley. 7ª edición. 2005.- Arnow, David M. Introducción a la programación con JAVA, Un enfoque orientado a objetos. Addison-Wesley. 2000.
<b>Bibliografía complementaria</b>	- Booch, G., Rumbaugh, J., Jacobson, I. El proceso unificado de desarrollo del software. Addison Wesley Iberoamericana. 2001. - López-Cortijo, R., de Amescua, A. Ingeniería del Software, Aspectos de Gestión. IIIS. 1998. - Niemeyer, P., Knudsen, J. Curso de JAVA. Anaya Multimedia. 2000. - Braude, Eric J. Software Engineering. An object-Oriented Perspective. John Wiley & Sons. 2001. - <a href="http://java.sun.com/reference/api">http://java.sun.com/reference/api</a>

### Recomendacións

**Materias que se recomienda ter cursado previamente**

**Materias que se recomienda cursar simultaneamente**

**Materias que continúan o temario**

Estrutura de Datos e da Información/614111102

Programación/614111109

**Observacións**



La asignatura tiene un carácter eminentemente práctico, por lo que es fundamental que los alumnos sepan aplicar los conocimientos teóricos aprendidos a la práctica.

(\*A Guía docente é o documento onde se visualiza a proposta académica da UDC. Este documento é público e non se pode modificar, salvo casos excepcionais baixo a revisión do órgano competente dacordo coa normativa vixente que establece o proceso de elaboración de guías