



Guía Docente				
Datos Identificativos				2014/15
Asignatura (*)	Principios de Análise Informática		Código	614211301
Titulación				
Descritores				
Ciclo	Período	Curso	Tipo	Créditos
1º e 2º Ciclo	2º cuatrimestre	Terceiro	Troncal	6
Idioma	Castelán			
Prerrequisitos				
Departamento	Computación			
Coordinación	Mosqueira Rey, Eduardo	Correo electrónico	eduardo.mosqueira@udc.es	
Profesorado	Mosqueira Rey, Eduardo	Correo electrónico	eduardo.mosqueira@udc.es	
Web				
Descrición xeral	<p>Esta asignatura sumerge al alumno en el ámbito de la Ingeniería del Software y de los proyectos de desarrollo de software, proporcionándole conocimientos básicos que le serán necesarios en un conjunto de asignaturas que irán descubriendo en sus estudios y que se centran en el contexto de la Ingeniería del Software.</p> <p>Esta materia también resulta muy interesante a la hora de configurar habilidades en el contexto del ámbito profesional ya que el Proceso Unificado es el modelo de desarrollo de software más utilizado en la actualidad de forma profesional. También el paradigma de la orientación a objetos es el dominante dentro de los lenguajes de programación más utilizados profesionalmente (Java, C#, C++, VisualBasic, Delphi, etc.).</p>			

Competencias da titulación	
Código	Competencias da titulación

Resultados da aprendizaxe			
Competencias de materia (Resultados de aprendizaxe)	Competencias da titulación		
Conocer las diferencias entre las distintas metodologías de desarrollo de software.	A1 A3	B2	C3
Comprender los conceptos básicos en los que se basa el Proceso Unificado de desarrollo de software	A1 A3	B2	C3
Analizar, comprender y relacionar los artefactos básicos que forman la Fase de Inicio del Proceso Unificado	A1 A4	B2	C3
Comprender las diferencias entre los aspectos de análisis (modelo conceptual del dominio, diagramas de secuencia del sistema) y diseño (arquitectura lógica, modelado estático vs. dinámico) dentro de la Fase de Elaboración del Proceso Unificado	A1 A5 A6	B2	C3
Conocer los aspectos básicos del diseño UML, sobre todo lo que involucra el diseño estático (diagrama de clases) y dinámico (diagrama de secuencia) de un software orientado a objetos.	A1 A5 A6	B2 B4	C3
Conocer, comprender e identificar los principios y patrones de diseño básicos para lograr un software orientado a objetos flexible y reusable.	A1 A5 A6	B2 B4	C3
Conocer los aspectos fundamentales de análisis y diseño de los componentes del software que no forman el dominio de aplicación, como el interfaz gráfico del usuario o el modelo de persistencia	A1 A5 A6	B2 B4	C3
Comprender la importancia de las pruebas del software e introducirse en el diseño dirigido por los tests y en el proceso de refactorización.	A1 A5 A6 A7	B2 B15	C3



Analizar e interpretar los requisitos de desarrollo de software suministrados por el cliente para la realización de un programa	A1 A4 A9	B2	C3
Desarrollar los artefactos básicos propios de la Fase de Inicio (visión, especificación suplementaria, glosario y modelo de casos de uso).	A1 A4	B2 B5 B10	C1 C3
Desarrollar los artefactos básicos propios de la Fase de Elaboración (modelo conceptual del dominio y modelo de diseño).	A1 A5 A6	B2 B5 B10	C1 C3
Desarrollar y organizar clases sacando provecho de las propiedades propias de la orientación a objetos para desarrollar código flexible y reusable.	A1 A5 A6	B2 B3 B5 B12	C3
Diseñar un programa orientado a objetos utilizando el estándar UML y desarrollar posteriormente el código resultante de dicho diseño.	A1 A5 A6	B2 B3 B5 B12	C3
Diseñar un programa utilizando principios y patrones de diseño garantizando así que las propiedades propias de la orientación a objetos se utilizan adecuadamente.	A1 A5 A6	B2 B3 B5 B12	C3
Desarrollar el interfaz gráfico del usuario siguiendo los principios del patrón Modelo-Vista-Controlador	A1 A2 A10	B2 B3	C3
Desarrollar la persistencia de las clases del domino	A1 A2 A8 A10	B2 B3	C3
Manejar con solvencia un entorno integrado de desarrollo (IDE) en la programación	A3	B2 B3 B4	C3
Asumir la responsabilidad del trabajo y las acciones propias.		B5 B6	C4
Valorar y respetar el trabajo propio y el ajeno.		B5 B6	C4
Colaborar con los compañeros para llevar a cabo una práctica en grupo.		B5 B7	C4
Valorar la importancia del análisis y el diseño en el desarrollo de programas orientados a objetos.		B10 B15	C3 C6
Preferir las buenas prácticas de programación orientada a objetos sobre los antipatrones (malas prácticas) propias de los principiantes o de los malos programadores.		B15	C6
Fomentar sus capacidades de razonamiento, abstracción, crítica, solución de problemas y trabajo en equipo.		B2 B3 B5 B7 B11	C4

Contidos	
Temas	Subtemas



1. Introducción	1.1. Metodoloxías 1.2. La programación orientada a objetos de desarrollo del software 1.3. El lenguaje de modelado visual UML
2. El Proceso Unificado de desarrollo de software	2.1. Características del Proceso Unificado 2.2. Fases del Proceso Unificado 2.3. Disciplinas del Proceso Unificado 2.4. Artefactos del Proceso Unificado 2.5. Casos de estudio
3. Fase de inicio	3.1. Introducción 3.2. Artefactos de la fase de inicio 3.3. Modelo de casos de uso
4. Fase de elaboración: modelado	4.1. Introducción 4.2. Artefactos de la fase de elaboración 4.3. Modelo del dominio 4.4. Modelado dinámico 4.5. Modelo de diseño
5. Fase de elaboración: principios y patrones	5.1. Principios de diseño 5.2. Patrones de diseño
6. Fase de construcción	6.1. Diseño del interfaz gráfico del usuario 6.2. Diseño de la persistencia
7. Fase de transición	7.1. Desarrollo dirigido por los tests 7.2. Refactorización

Planificación			
Metodoloxías / probas	Horas presenciais	Horas non presenciais / traballo autónomo	Horas totais
Proba obxectiva	3	0	3
Actividades iniciais	146	0	146
Atención personalizada	1	0	1

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

Metodoloxías	
Metodoloxías	Descrición
Proba obxectiva	Proba escrita na que se preguntarán polos contidos teóricos e prácticos seguindo o temario do curso 2011/2012
Actividades iniciais	Repaso do temario visto no curso

Atención personalizada	
Metodoloxías	Descrición
Proba obxectiva	Resolución de dudas y cuestionese que podan surxir durante o desenvolvemento da proba obxectiva.

Avaliación		
Metodoloxías	Descrición	Cualificación
Proba obxectiva	Constará dunha parte teórica (50%) e dunha parte práctica (50%). En caso de sacar un 5 o máis dentro dunha parte gardarase esa nota para todas as convocatorias celebradas no mesmo curso.	100
Outros		



Observacións avaliación

Fontes de información
Bibliografía básica
Bibliografía complementaria

Recomendacións
Materias que se recomenda ter cursado previamente
Estrutura de Datos e da Información/614211102 Programación/614211107 Programación Orientada a Obxectos/614211636
Materias que se recomenda cursar simultaneamente
Programación Orientada a Obxectos/614211636
Materias que continúan o temario
Proxecto fin de Carreira/614211310
Observacións

(*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