



Guía Docente				
Datos Identificativos				2012/13
Asignatura (*)	Enxeñaría de Requisitos	Código	614G01027	
Titulación	Grao en Enxeñaría Informática			
Descritores				
Ciclo	Período	Curso	Tipo	Créditos
Grao	2º cuatrimestre	Terceiro	Obrigatoria	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>O propósito da disciplina de análise de requisitos é: (1) establecer e manter un acordo cos clientes e outras partes interesadas sobre o que o sistema debe facer. (2) Proporcionar aos desenvolvedores do sistema unha mellor comprensión dos requisitos do mesmo. (3) Definir os límites do sistema. (4) Proporcionar unha base para a planificación dos contidos técnicos de próximas iteracións. (5) Proporcionar unha base para a estimación de custo e tempo para desenvolver o sistema e (6) Definir unha interfaz para o sistema, centrándose nas necesidades e obxectivos dos usuarios.</p> <p>Nesta materia centrarémonos en describir o proceso de enxeñaría de requisitos seguindo a metodoloxía do Proceso Unificado e desenvolvendo artefactos propios da devandita metodoloxía para o tratamento dos requisitos: Documento de Visión, Especificación Suplementaria, Modelo de Casos de Uso, Diagrama Conceptual do Dominio, etc.</p> <p>Tamén se exporá como desenvolver requisitos en metodoloxías áxiles como Scrum, comparando estas técnicas coas xa nomeadas do Proceso Unificado.</p>			

Competencias da titulación	
Código	Competencias da titulación
A7	Capacidade para deseñar, desenvolver, seleccionar e avaliar aplicacións e sistemas informáticos que aseguren a súa fiabilidade, seguranza e calidade, conforme a principios éticos e á lexislación e normativa vixente.
A13	Coñecemento, deseño e utilización de forma eficiente dos tipos e estruturas de datos máis adecuados á resolución dun problema.
A14	Capacidade para analizar, deseñar, construír e manter aplicacións de forma robusta, segura e eficiente, elixindo o paradigma e as linguaxes de programación máis adecuados.
A22	Coñecemento e aplicación dos principios, metodoloxías e ciclos de vida da enxeñaría do sóftware.
A25	Capacidade para desenvolver, manter e avaliar servizos e sistemas sóftware que satisfagan todos os requisitos do usuario e se comporten de forma fiable e eficiente, sexan accesibles de desenvolver e manter, e cumpran normas de calidade, aplicando as teorías, principios, métodos e prácticas da enxeñaría do sóftware.
A26	Capacidade para valorar as necesidades do cliente e especificar os requisitos sóftware para satisfacer estas necesidades, reconciliando obxectivos en conflito mediante a procura de compromisos aceptables dentro das limitacións derivadas do custo, do tempo, da existencia de sistemas xa desenvolvidos e das propias organizacións.
B1	Expresarse correctamente, tanto de forma oral coma escrita, nas linguas oficiais da comunidade autónoma.
B3	Utilizar as ferramentas básicas das tecnoloxías da información e as comunicacións (TIC) necesarias para o exercicio da súa profesión e para a aprendizaxe ao longo da súa vida.
B9	Capacidade de resolución de problemas
B10	Traballo en equipo
B11	Capacidade de análise e síntese
B13	Habilidades de xestión da información
C1	Expresarse correctamente, tanto de forma oral coma escrita, nas linguas oficiais da comunidade autónoma.



C3	Utilizar as ferramentas básicas das tecnoloxías da información e as comunicacións (TIC) necesarias para o exercicio da súa profesión e para a aprendizaxe ao longo da súa vida.
C7	Asumir como profesional e cidadán a importancia da aprendizaxe ao longo da vida.

Resultados da aprendizaxe			
Competencias de materia (Resultados de aprendizaxe)	Competencias da titulación		
Identificar a análise de requisitos como unha das fases do ciclo de vida do software	A14 A22	B1 B3 B9 B13	C1 C3 C7
Identificar o fluxo de traballo de requisitos nunha metodoloxía estándar como o Proceso Unificado	A22 A25 A26	B1 B3 B9 B13	C1 C3 C7
Coñecer e saber aplicar os artefactos de enxeñaría de requisitos na Fase de Inicio do Proceso Unificado	A22 A25 A26	B1 B3 B9 B10 B11 B13	C1 C3 C7
Coñecer e saber aplicar os artefactos de enxeñaría de requisitos na Fase de Elaboración do Proceso Unificado	A22 A25 A26	B1 B3 B10 B11 B13	C1 C3 C7
Coñecer as principais tarefas involucradas na xestión de requisitos: elicitación, negociación, validación, xestión de cambios, trazabilidade, etc.	A25 A26	B1 B3 B13	C1 C3 C7
Identificar o fluxo de traballo de requisitos nunha metodoloxía áxil como Scrum	A22 A25 A26	B1 B3 B9 B10	C1 C3 C7
Desenvolver un problema desde a súa especificación de requisitos inicial até o seu deseño e implementación nunha linguaxe orientada a obxectos como Java	A7 A13 A14 A25 A26	B1 B3 B9 B10	C1 C3 C7

Contidos	
Temas	Subtemas
1. Introducción	? Introducción á enxeñaría de requisitos ? A enxeñaría de requisitos nas metodoloxías de desenvolvemento
2. O proceso unificado de desenvolvemento	? Introducción ? Fases ? Disciplinas ? Fluxo de traballo de requisitos



3. Os requisitos na fase de inicio	? Características da fase de inicio ? Artefactos ? Documento de visión ? Especificación suplementaria ? Glosario
4. Casos de uso	? Obxectivos ? Actores ? Casos de uso ? Diagrama de casos de uso
5. Os requisitos na fase de elaboración	? Características da fase de elaboración ? Modelado conceptual do dominio ? Patróns de análise ? Diagramas de secuencia do sistema
6. Xestión de requisitos	? Tarefas de xestión de requisitos: elicitación, negociación, validación, xestión de cambios, trazabilidade
7. Enxeñaría de requisitos nas metodoloxías áxiles	? Características das metodoloxías áxiles ? Metodoloxía Scrum ? Requisitos en Scrum

Planificación			
Metodoloxías / probas	Horas presenciais	Horas non presenciais / traballo autónomo	Horas totais
Sesión maxistral	21	42	63
Prácticas de laboratorio	14	28	42
Seminario	7	14	21
Solución de problemas	4	8	12
Proba obxectiva	3	6	9
Atención personalizada	3	0	3

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

Metodoloxías	
Metodoloxías	Descrición
Sesión maxistral	Clases maxistras nas que se expoñen os conceptos teóricos da materia utilizando como recursos a pizarra, a proxección de materiais en formato electrónico, apuntes en formato electrónico, consulta de páxinas web e outro tipo de materiais que estean dispoñibles no campus virtual
Prácticas de laboratorio	Prácticas realizadas en grupos pequenos, xeralmente de dúas persoas, na que os estudantes poñen en práctica os coñecementos teóricos impartidos nas sesións maxistras. As prácticas consistirán na realización das tarefas de enxeñaría de requisitos dun problema particular, partindo das definicións iniciais até chegar a unha implementación final nunha linguaxe de programación como Java
Seminario	Os seminarios ou Tutorías de Grupo Reducido (TGRs) consistirán en pequenos exercicios nos que se pon en práctica, de forma sinxela, os coñecementos vistos nas sesións maxistras e que logo han de ampliar nas prácticas de laboratorio.
Solución de problemas	Os exercicios realizados nos seminarios xeralmente serán corrixis polos propios alumnos seguindo as guías que o profesor facilite para cada caso. Desta forma os alumnos serán máis conscientes dos fallos cometidos e poderán evitalos á hora de realizar as prácticas de laboratorio
Proba obxectiva	Trátase dunha proba escrita mediante a cal se valoran os coñecementos adquiridos polo estudante

Atención personalizada	
Metodoloxías	Descrición





<b>Bibliografía básica</b>	<ul style="list-style-type: none"><li>- Larman, Craig (2005). Applying UML and Patterns (3rd ed.). Pearson Education</li><li>- McLaughlin, Brett; Pollice, Gary &amp; West, David (2007). Head First Object-Oriented Analysis &amp; Design. O'Reilly Media</li><li>- Alvarez, Alonso; De las Heras, Raquel; Lasa, Carmen (2002). Métodos ágiles y Scrum. Anaya Multimedia</li><li>- Eckel, Bruce (2007). Piensa en Java (4ª ed.). Pearson Educación</li><li>- Kruchten, Philippe (2004). The Rational Unified Process: An Introduction. Addison-Wesley</li><li>- Cockburn, Alistair (2001). Writing effective use cases. Addison-Wesley</li></ul>
<b>Bibliografía complementaria</b>	<ul style="list-style-type: none"><li>- Fowler, Martin (1997). Analysis Patterns: Reusable Object Models. Addison-Wesley</li><li>- Pilone, Dan &amp; Miles, Russ (2008). Head First Software Development. O'Reilly Media</li><li>- Adolph, Steve; Bramble, Paul (2002). Patterns for Effective Use Cases. Addison-Wesley Professional</li><li>- Braude, Eric J.; Bernstein, Michael E. (2011). Software Engineering: Modern Approaches (2nd ed.). John Wiley and Sons</li><li>- Larman, Craig (2002). UML y Patrones (2ª ed.). Pearson Educación</li></ul>

## Recomendacións

### Materias que se recomenda ter cursado previamente

Validación e Verificación do Software/614G01053

### Materias que se recomenda cursar simultaneamente

### Materias que continúan o temario

Deseño Software/614G01015

Proceso Software/614G01019

### 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