



## Teaching Guide

Identifying Data					2020/21
Subject (*)	Requirements Engineering	Code	614G01027		
Study programme	Grao en Enxeñaría Informática				
Descriptors					
Cycle	Period	Year	Type	Credits	
Graduate	2nd four-month period	Third	Optional	6	
Language	Spanish				
Teaching method	Hybrid				
Prerequisites					
Department	Ciencias da Computación e Tecnoloxías da InformaciónComputación				
Coordinador	Rodriguez Rubio, Miguel Jose	E-mail	miguel.rodriguez.rubio@udc.es		
Lecturers	Rodriguez Rubio, Miguel Jose	E-mail	miguel.rodriguez.rubio@udc.es		
Web					
General description	<p>The purpose of the Requirements discipline is: (1) To establish and maintain agreement with the customers and other stakeholders on what the system should do. (2) To provide system developers with a better understanding of the system requirements. (3) To define the boundaries of (delimit) the system. (4) To provide a basis for planning the technical contents of iterations. (5) To provide a basis for estimating cost and time to develop the system and (6) To define a user-interface for the system, focusing on the needs and goals of the users.</p> <p>In this subject we will focus on describing the requirements engineering process following the Unified Process methodology and using its artifacts for the treatment of requirements: Vision Document, Supplementary Specification, Use Case Model, Conceptual Domain Diagram, etc.</p> <p>We will also consider how to develop requirements in agile methodologies like Scrum, comparing these techniques with those of the Unified Process.</p>				
Contingency plan	<ol style="list-style-type: none"> <li>Modifications to the contents</li> <li>Methodologies <ul style="list-style-type: none"> <li>*Teaching methodologies that are maintained</li> <li>*Teaching methodologies that are modified</li> </ul> </li> <li>Mechanisms for personalized attention to students</li> <li>Modifications in the evaluation <ul style="list-style-type: none"> <li>*Evaluation observations:</li> </ul> </li> <li>Modifications to the bibliography or webgraphy</li> </ol>				

## Study programme competences / results

Code	Study programme competences / results
A26	Capacidade para valorar as necesidades do cliente e especificar os requisitos software 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.
A28	Capacidade de identificar e analizar problemas, e deseñar, desenvolver, implementar, verificar e documentar solucións software sobre a base dun coñecemento adecuado das teorías, modelos e técnicas actuais.



B1	Capacidade de resolución de problemas
B2	Traballo en equipo
B3	Capacidade de análise e síntese
B4	Capacidade para organizar e planificar
C2	Dominar a expresión e a comprensión de forma oral e escrita dun idioma estranxeiro.
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.
C4	Desenvolverse para o exercicio dunha cidadanía aberta, culta, crítica, comprometida, democrática e solidaria, capaz de analizar a realidade, diagnosticar problemas, formular e implantar solucións baseadas no coñecemento e orientadas ao ben común.
C6	Valorar criticamente o coñecemento, a tecnoloxía e a información dispoñible para resolver os problemas cos que deben afrontarse.
C7	Asumir como profesional e cidadán a importancia da aprendizaxe ao longo da vida.
C8	Valorar a importancia que ten a investigación, a innovación e o desenvolvemento tecnolóxico no avance socioeconómico e cultural da sociedade.

Learning outcomes			
Learning outcomes	Study programme competences / results		
Identificar a análise de requisitos como unha das fases do ciclo de vida do software	A26 A28	B3 B4	C2 C6 C7 C8
Identificar o fluxo de traballo de requisitos nunha metodoloxía estándar como o Proceso Unificado	A26 A28	B3 B4	C2 C6 C7 C8
Coñecer e saber aplicar os artefactos de enxeñaría de requisitos na Fase de Inicio do Proceso Unificado	A26 A28	B1 B2 B3 B4	C2 C3 C4 C6 C7 C8
Coñecer e saber aplicar os artefactos de enxeñaría de requisitos na Fase de Elaboración do Proceso Unificado	A26 A28	B1 B2 B3 B4	C2 C3 C4 C6 C7 C8
Coñecer as principais tarefas involucradas na xestión de requisitos: elicitación, negociación, validación, xestión de cambios, trazabilidade, etc.	A26 A28	B1 B2 B3 B4	C2 C3 C4 C6 C7 C8
Identificar o fluxo de traballo de requisitos nunha metodoloxía áxil como Scrum	A26 A28	B1 B2 B3 B4	C2 C3 C4 C6 C7 C8



Desenvolver un problema desde a súa especificación de requisitos inicial até o seu deseño e implementación nunha linguaxe orientada a obxectos como Java	A26	B1	C2
	A28	B2	C3
		B3	C4
		B4	C6
			C7
			C8

Contents	
Topic	Sub-topic
1. Introducción	? Introducción á enxeñaría de requisitos ? A enxeñaría de requisitos nas metodoloxías de desenvolvemento ? A enxeñaría de requisitos no Proceso Unificado
2. Os requisitos na fase de inicio	? Características da fase de inicio ? Documento de visión ? Especificación suplementaria ? Glosario
3. Casos de uso	? Obxectivos ? Actores ? Casos de uso ? Diagrama de casos de uso ? Problemas comúns con casos de uso ? Casos de uso no fluxo de requisitos
4. Os requisitos na fase de elaboración	? Introducción ? Artefactos da fase de elaboración ? Modelo conceptual do dominio ? Modelado dinámico
5. Requisitos e Deseño	? Modelo de deseño ? Padróns GRASP ? Realización de casos de uso ? Casos de uso e interfaces de usuario
6. Xestión de requisitos	? Características dos requisitos ? Elicitación de requisitos ? Requisitos non ambiguos ? Verificación de requisitos ? Atributos dos requisitos ? Xestión de cambios ? Trazabilidade
7. Enxeñaría de requisitos nas metodoloxías áxiles	? Introducción ? O Proceso Unificado Áxil ? Metodoloxía Scrum ? Casos de Uso vs. Historias de Usuario

Planning				
Methodologies / tests	Competencies / Results	Teaching hours (in-person & virtual)	Student?s personal work hours	Total hours
Guest lecture / keynote speech	A26 B1 C7	21	51	72
Laboratory practice	A28 B2 B4 C3	13	28	41
Supervised projects	B3 C4 C6 C8	8	20	28
Objective test	C2	3	0	3
Personalized attention		6	0	6



(\*The information in the planning table is for guidance only and does not take into account the heterogeneity of the students.

Methodologies	
Methodologies	Description
Guest lecture / keynote speech	Clases maxistras nas que se expoñen os conceptos teóricos da materia utilizando entre outros recursos, materiais en formato electrónico, apuntamentos en formato electrónico, consulta de páxinas web e outro tipo de materiais que estean dispoñibles no campus virtual.
Laboratory practice	Prácticas realizadas en grupos nas 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.
Supervised projects	Traballos realizados en grupo nos que os estudantes poñen en práctica cómo transitar dende a especificación de requisitos realizada nas prácticas ata o deseño e a implementación final do software.
Objective test	Proba escrita mediante a que se valoran os coñecementos adquiridos polo estudantado. Cada estudante deberá aplicar tanto os seus coñecementos tanto a nivel teórico coma a nivel práctico.

Personalized attention	
Methodologies	Description
Supervised projects	A atención personalizada aos alumnos comprende: <ul style="list-style-type: none"> <li>- Tutorías presenciais</li> <li>- Tutorías virtuais a través dos foros do campus virtual</li> <li>- Seguimento do labor realizado nas prácticas de laboratorio e traballos tutelados</li> <li>- Encontros personalizados para resolver dúbidas</li> </ul>

Assessment			
Methodologies	Competencies / Results	Description	Qualification
Objective test	C2	Proba realizada ao final do curso sobre contidos teórico-prácticos.  A proba obxectiva é obrigatoria para aprobar a materia.	30
Laboratory practice	A28 B2 B4 C3	Realizaranse un seguimento continuo e programado de prácticas ao longo do curso. O desenvolvemento será incremental e cada presentación ou discusión dunha parte estará baseada na anterior.	40
Supervised projects	B3 C4 C6 C8	Os traballos tuteados completan as prácticas e tamén seguen una metodoloxía incremental.  A calificación do último traballo, ao englobar ao resto de traballos e prácticas, considérase parte da avaliación final e entregarase coas calificacións finais da asignatura.	30

Assessment comments

Sources of information



<b>Basic</b>	<ul style="list-style-type: none"><li>- Larman, Craig (2005). Applying UML and Patterns (3rd ed.). Pearson Education</li><li>- Kruchten, Philippe (2004). The Rational Unified Process: An Introduction. Addison-Wesley</li><li>- McLaughlin, Brett; Pollice, Gary &amp; West, David (2007). Head First Object-Oriented Analysis &amp; Design. O'Reilly Media</li><li>- Cockburn, Alistair (2001). Writing effective use cases. Addison-Wesley</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>- James Rumbaugh y otros (). El Lenguaje Unificado de Modelado. Manual de Referencia. Addison Wesley</li></ul> <p>Se mantiene la bibliografía de los cursos anteriores para respetar la continuidad. Se añaden dos nuevas referencias bibliográficas fundamentales en la orientación actual de la asignatura.</p>
<b>Complementary</b>	<ul style="list-style-type: none"><li>- Larman, Craig (2002). UML y Patrones (2ª ed.). Pearson Educación</li><li>- Braude, Eric J.; Bernstein, Michael E. (2011). Software Engineering: Modern Approaches (2nd ed.). John Wiley and Sons</li><li>- Pitone, 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>- Stevens, P. y Pooley, R. (2006). Using UML. Software Engineering with Objects and Components. Addison-Wesley Professional</li></ul>

## Recommendations

### Subjects that it is recommended to have taken before

Software Design/614G01015

Software Process/614G01019

### Subjects that are recommended to be taken simultaneously

### Subjects that continue the syllabus

Software Verification and Validation/614G01053

### Other comments

(\*)The teaching guide is the document in which the URV publishes the information about all its courses. It is a public document and cannot be modified. Only in exceptional cases can it be revised by the competent agent or duly revised so that it is in line with current legislation.