



Teaching Guide

Teaching Guide				
Identifying Data				2020/21
Subject (*)	Development Methodologies		Code	614G01051
Study programme	Grao en Enxeñaría Informática			
Descriptors				
Cycle	Period	Year	Type	Credits
Graduate	2nd four-month period	Fourth	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	Metodoloxías existentes para distintos ciclos de vida (áxil, clásica, ferverza, espiral,...). Metodoloxías clásicas. Proceso unificado de desenvolvemento. Metodoloxías áxiles de desenvolvemento. Programación extrema. Desenvolvemento colaborativo. Evolución e mantemento do software (integración, sistemas legacy,...). Metodoloxía e aspectos éticos/sociais/legais característicos en distintos dominios de aplicación.			
Contingency plan	1. Modifications to the contents 2. Methodologies *Teaching methodologies that are maintained *Teaching methodologies that are modified 3. Mechanisms for personalized attention to students 4. Modifications in the evaluation *Evaluation observations: 5. Modifications to the bibliography or webgraphy			

Study programme competences

Code	Study programme competences
A25	Capacidade para desenvolver, manter e avaliar servizos e sistemas software 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 software.
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.
A27	Capacidade de dar solución a problemas de integración en función das estratexias, estándares e tecnoloxías dispoñibles.
A30	Capacidade para deseñar solucións apropiadas nun ou máis dominios de aplicacións utilizando métodos da enxeñaría do software que integren aspectos éticos, sociais, legais e económicos.
A54	Capacidade para empregar metodoloxías centradas no usuario e a organización para o desenvolvemento, a avaliación e a xestión de aplicacións e sistemas baseados nas tecnoloxías da información que aseguren a accesibilidade, ergonomía e a usabilidade dos sistemas.
B1	Capacidade de resolución de problemas
B2	Traballo en equipo



B3	Capacidade de análise e síntese
B4	Capacidade para organizar e planificar
B5	Habilidades de xestión da información
B6	Toma de decisións
B7	Preocupación pola calidade
B8	Capacidade de traballar nun equipo interdisciplinar
B9	Capacidade para xerar novas ideas (creatividade)
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.
C6	Valorar criticamente o coñecemento, a tecnoloxía e a información dispoñible para resolver os problemas cos que deben enfrontarse.
C7	Asumir como profesional e cidadán a importancia da aprendizaxe ao longo da vida.

Learning outcomes			
Learning outcomes		Study programme competences	
Capacidade para desenvolver, manter e avaliar servizos e sistemas software 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 software. Competencias SE1, SE3 e SE6 na memoria do Grado.		A25	B1 C3
		A26	B2 C6
		A27	B3 C7
		A30	B4
		A54	B5
			B6
			B7
			B8
			B9

Contents	
Topic	Sub-topic
PARTE A. Metodoloxías e ciclos de vida	Tema 1. Evolución das metodoloxías e os seus enfoques Tema 2. Metodoloxías clásicas Tema 3. Encadre das metodoloxías actuais
PARTE B. Metodoloxías para Desenvolvemento Incremental	Tema 4. Estructura e alcance do Proceso Unificado de Desenvolvemento Software Tema 5. Proceso: dirixido por casos de uso; centrado en arquitecturas; iterativo e incremental Tema 6. Fluxos de traballo: captura de requisitos, análise, deseño, implementación, validación e proba
PARTE C. Metodoloxías áxiles de desenvolvemento	Tema 7. Introducción a metodoloxías áxiles Tema 8. SCRUM: roles, equipos e auto-organización; planificación de reunións; sprints; xestión e implementación. Tema 9. Programación extrema: principios de XP; conceptualización; planificación; desenvolvemento; entrega.
PARTE D. Complementos	Tema 10. Desenvolvemento colaborativo Tema 11. Evolución e mantemento do software Tema 12. O impacto legacy Tema 13. Outros aspectos do desenvolvemento. Deontoloxía e encadre legal.

Planning				
Methodologies / tests	Competencies	Ordinary class hours	Student's personal work hours	Total hours
Guest lecture / keynote speech	A25 A27 A30 B1 B3 B5 C3 C6 C7	21	52.5	73.5



Laboratory practice	A26 A54 B2 B4 B8	14	14	28
Problem solving	B7 B9	7	14.5	21.5
Objective test	B6	3	18	21
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 de teoría onde se imparten os contidos do temario
Laboratory practice	Elaboración de traballos prácticos tipo proxecto.
Problem solving	Resolución de traballos tutelados planteados e resoltos en horario de tutorías de grupos reducidos
Objective test	Examen escrito

Personalized attention	
Methodologies	Description
Guest lecture / keynote speech Laboratory practice Problem solving Objective test	Atención personalizada. Resolución de dúbidas de teoría ou prácticas, exercicios, etc.

Assessment			
Methodologies	Competencies	Description	Qualification
Laboratory practice	A26 A54 B2 B4 B8	Entrega de traballos prácticos tipo proxecto de modo programado e en prazos establecidos ao longo do curso. Computa un máximo de 5 puntos sobre o total de 10 da nota final.	50
Problem solving	B7 B9	Formulación de pequenos problemas relacionados co práctica/proxecto e resolución en grupos de tutorías reducidos. Computa un máximo de 3 puntos sobre o total de 10 da nota final.	30
Objective test	B6	Realizarase unha proba de valoración dos conceptos teóricos discutidos e aplicados nas prácticas. A puntuación máxima desta proba será de 3 puntos. A presentación a esta proba é obrigatoria para aprobar a materia.	30

Assessment comments

Sources of information	
Basic	- Ivar Jacobson, Grady Booch and James Rumbaugh (). El Proceso Unificado de Desarrollo Software. Addison-Wesley - James Rumbaugh, Ivar Jacobson and Grady Booch (). El Lenguaje Unificado de Modelado. Manual de Referencia. Addison-Wesley - Craig Larman (). Agile & Iterative Development. Addison-Wesley - Robert C. Martin (). Agile Software Development. Prentice Hall - Ken Schwaber and Mike Beedle (). Agile Software Development with Scrum. Prentice Hall - David Astels, Geanville Miller and Miroslav Novak (). A Practical Guide to Extreme Programming. Prentice Hall
Complementary	



Recommendations
Subjects that it is recommended to have taken before
Quality Assurance/614G01028
Subjects that are recommended to be taken simultaneously
Development Frameworks/614G01052
Software Verification and Validation/614G01053
Subjects that continue the syllabus
Software Development Projects/614G01087
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.