



## Teaching Guide

Identifying Data					2014/15
Subject (*)	Ferramentas de Desenvolvemento	Code	614G01054		
Study programme	Grao en Enxeñaría Informática				
Descriptors					
Cycle	Period	Year	Type	Credits	
Graduate	1st four-month period	Fourth	Obligatoria	6	
Language	SpanishGalicianEnglish				
Prerequisites					
Department	Computación				
Coordinador	Parapar López, Javier	E-mail	javier.parapar@udc.es		
Lecturers	Parapar López, Javier	E-mail	javier.parapar@udc.es		
Web	<a href="http://www.dc.fi.udc.es/~parapar/">http://www.dc.fi.udc.es/~parapar/</a>				
General description	<p>Nesta materia explicaranse os fundamentos conceptuais e prácticos en canto a ferramentas de axuda ao desenvolvemento de proxectos software. En particular explicaranse as seguintes ferramentas, as súas vantaxes e o seu correcto aproveitamento desde un punto de vista da Enxeñaría do Software:</p> <p>Contornas integradas de desenvolvemento: Eclipse</p> <p>Ferramentas de automatización de empaketado: Maven</p> <p>Ferramentas de control de versións e traballo cooperativo : Git</p> <p>Ferramentas de integración continúa: Apache Jenckins</p> <p>Ferramentas de task/time/bug tracking: Redmine</p> <p>Ferramentas de análise de código e dependencias: Codepro Analytix, Sonar</p> <p>Ferramentas de análise de rendemento e monitorización: JStat, JConsole, JVisualVM</p> <p>A materia ten pois un carácter marcadamente práctico e tenta achegar ao estudante coñecemento profundo sobre as ferramentas de desenvolvemento máis usadas a día de hoxe na contorna empresarial da TI.</p>				

## Study programme competences

Code	Study programme competences
A19	Coñecemento e aplicación das ferramentas necesarias para o almacenamento, procesamento e acceso aos sistemas de información, incluídos os baseados en web.
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.
B2	Traballo en equipo
B4	Capacidade para organizar e planificar
B8	Capacidade de traballar nun equipo interdisciplinar
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.

## Learning outcomes

Subject competencies (Learning outcomes)	Study programme competences		
Software Development	A19	B2	C3
	A22	B4	C6
	A25	B8	

## Contents

Topic	Sub-topic



Integrated Development Environments	Eclipse
Tools for packing automation	Maven
Tools for version control and cooperative work	Git
Tools for continue integration	Jenkins
Tools for task/time/bug tracking	Redmine
Tools for analysis of code and dependencies	Codepro Analytix, Sonar
Tools for analysis of performance and monitoring	JStat, JConsole, JVisualVM

Planning			
Methodologies / tests	Ordinary class hours	Student?s personal work hours	Total hours
Supervised projects	7	21	28
Laboratory practice	14	42	56
Mixed objective/subjective test	0.5	0	0.5
Guest lecture / keynote speech	18	47.5	65.5
Personalized attention	0		0

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

Methodologies	
Methodologies	Description
Supervised projects	Traballos tutelados propostos polo profesor e desenvolvidos polos estudantes ou ben en grupo ou ben individualmente.
Laboratory practice	Ao tratarse dunha materia eminentemente práctica o desenvolvemento por parte do alumno dun caso de práctica para o uso de todas as ferramentas comentadas nas leccións maxistrais será fundamental
Mixed objective/subjective test	Avaliarase o dominio dos coñecementos teóricos e operativos da materia.
Guest lecture / keynote speech	Clases maxistrais na exposición dos coñecementos teóricos utilizando diferentes recursos: a lousa, transparencias, proxeccións, demostracións e a facultade virtual. Pode incluír conferencia convidada.

Personalized attention	
Methodologies	Description
Laboratory practice	Se propondran pequenos traballos tutelados para a resolución por parte do alumno co soporte do coñecemento do profesor.
Supervised projects	Ao tratarse dunha materia eminentemente práctica o desenvolvemento por parte do alumno dun caso de práctica para o uso de todas as ferramentas comentadas nas leccións maxistrais será fundamental

Assessment		
Methodologies	Description	Qualification
Laboratory practice	Correctness and completeness of the assignments proposed to the students while adequately using the dev tools introduced in the subjects. It is mandatory to achieve at least the 40% of the marks in order to pass the course	40
Supervised projects	Tracking of the working process and evaluation of the final output from the students. It is mandatory to achieve at least the 40% of the marks in order to pass the course	20
Mixed objective/subjective test	Questions about the acquired knowledge. Questions involving critical reasoning for solving practical problems of the real world.It is mandatory to achieve at least the 40% of the marks in order to pass the course	40

Assessment comments
---------------------



Para a segunda oportunidade, tanto as prácticas e traballos como a teorías avaliaranse no exame mixto. Se non se acada a nota mínima nas distintas probas a nota máxima do alumno será 4.5

## Sources of information

<b>Basic</b>	<ul style="list-style-type: none"><li>- Alan Berg (). Jenkins Continuous Integration Cookbook,. Packt Publishing</li><li>- John Ferguson Smart (). Jenkins: The Definitive Guide. O'Reilly</li><li>- Andriy Lesyuk (). Mastering Redmine. Packt Publishing</li><li>- Sonatype Company (). Maven: The Definitive Guide. O'Reilly</li><li>- G. Ann Campbell, Patroklos P. Papapetrou (). Sonar in Action. Manning Publications</li><li>- Jon Loeliger &amp; Matthew McCullough (). Version Control with Git: Powerful tools and techniques for collaborative software development. O'Reilly</li></ul>
<b>Complementary</b>	

## Recommendations

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

### Subjects that are recommended to be taken simultaneously

Metodoloxías de Desenvolvemento/614G01051

Validación e Verificación do Software/614G01225

### Subjects that continue the syllabus

Internet e sistemas distribuídos/614G01023

Programación Avanzada/614G01030

### 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.