SUBJECT TEACHING GUIDE

G668 - Methods of Development

Degree in Computer Systems Engineering

Academic year 2019-2020

1. IDENTIFYING DATA					
Degree	Degree in Computer Systems Engineering			Type and Year	Optional. Year 4
Faculty	Faculty of Sciences				
Discipline	Subject Area: Software Engineering Mention in Software Engineering				
Course unit title and code	G668 - Methods of Development				
Number of ECTS credits allocated	6	Term		Semester based (1)	
Web	http://moodle.unican.es				
Language of instruction	Spanish	English Friendly	No	Mode of delivery	Face-to-face

Department	DPTO. INGENIERÍA INFORMÁTICA Y ELECTRÓNICA
Name of lecturer	PABLO SANCHEZ BARREIRO
E-mail	p.sanchez@unican.es
Office	Facultad de Ciencias. Planta: + 1. DESPACHO PROFESOR (1069)
Other lecturers	JULIO LUIS MEDINA PASAJE

3.1 LEARNING OUTCOMES
- To be able to develop software products following a well-known methodology
- To be able to use one of the main software development paradigms : model-driven development, component-based, aspect-oriented, event-based, etc.
- To be able to use the main techniques and regulations to elaborate a plan for a development or maintenance project.
- To understands how a software process works and to be able to model and specify them.
- To use techniques and tools for configuration management.
- To use methodologies and techniques to develop modernization projects or services.
- To be able to use the main methods, techniques and tools for software testing, verification and validation.
- To be able to use methods and techniques for software project management and monitoring.
- To be able to estimate size,effort and cost of a software project.
- To be able to perform a risk analysis and elaborate a risk mitigation and control plan for a software development project or for a software system in operation.

4. OBJECTIVES
To be able to manage software system configurations using a software configuration manager like Git.
To know and understand how a software configuration management process works , including processes such as continuous integration.
To know the differences between heavyweight and agile methodologies.
To know and understand the foundations of agile methodologies.
To know and understand the main techniques of agile methodologies.
To know and understand the similarities and differences between the main agile methodologies.
To be able to develop a medium-size software system using an agile methodology like Scrum.
To be able to model software development processes using a software process modelling language like SPEM.
To know and understand hoe a software maintenance process work.
To know and understand how the Métrica v3 software development process works. NOTE: Métricav3 is the software development methodology used by most of the Spanish Institutions and Public Services.
To know and understand advanced concepts related to software development processes , such as Software Product Lines or Model-Driven Development.

6. COURSE ORGANIZATION

	CONTENTS
1	Unit 1. Software Configuration Management Introduction. Terminology. Software Configuration Management Processes. Continuous Integration and Delivery. Advanced Version Control with Git.
2	Unit 2. Agile Methodologies. Heavyweight and Agile Methodologies. Agile Manifesto. Lean Principles. Foundational Techniques of Agile Methodologies: User Stories, Test-Driven Development, Planning Poker, Pair-Programming. Scrum. Additional Agile Methodologies: XP, Kanban.
3	Unit 3. Software Process Modelling Definition of Software Process. Software Process Modelling with SPEM.
4	Unit 4. Regularized Software Methdologies Métrica v3. Software Maintenance Processes in Métrica v3.
5	Unit 5. Cutting-Edge Software Methodologies Model-Driven Development. Component-Based Software Development. Software Product Lines.

7. ASSESSMENT METHODS AND CRITERIA

Description	Type	Final Eval.	Reassessn	%
Lab Assignments	Laboratory evaluation	No	Yes	30,00
Development of a Software Project using Scrum.	Work	No	Yes	70,00
TOTAL				100,00

Observations

Instructors might perform some extra checks during the semesters in order to verify the authority of the assignments delivered by the students. Plagiarism is bot allowed and it will imply that the subject will fail the subject. In addition, it will be notified to the Faculty Council so that the adequate disciplinary actions can be adopted.

Observations for part-time students

The main goal of this subject is that students learn to develop software projects in teams and following a well-defined software methodology. Therefore, a minimum assistance to the classroom is required since each student needs to interact with her team. The work in teams is limited to weeks 5th to 10th. Thus, it is recommended that all students can assist to the classroom during these weeks. Assistance is required since instructors need to observe and evaluate how each student interacts with her team. This minimum attendance will be established at the beginning of the semester, so that each student can make the arrangements she needs as soon as possible.

8. BIBLIOGRAPHY AND TEACHING MATERIALS

BASIC

Scott Chacon, Ben Straub "Pro Git". 2ª Edición. Apress. Noviembre 2014.

Object Management Group. "Software & Systems Process Engineering Meta-Model Specification". Estandar formal/2008-04-01 Abril 2008.

Jeff Sutherland. "Scrum: The Art of Doing Twice the Work in Half the Time". Crown Business. 2014