

## SUBJECT TEACHING GUIDE

### G658 - Software Engineering I

#### Degree in Computer Systems Engineering

Academic year 2023-2024

1. IDENTIFYING DATA					
Degree	Degree in Computer Systems Engineering			Type and Year	Compulsory. Year 3
Faculty	Faculty of Sciences				
Discipline	Subject Area: Software and Information Systems Engineering Compulsory Module				
Course unit title and code	G658 - Software Engineering I				
Number of ECTS credits allocated	6	Term	Semester based (1)		
Web					
Language of instruction	Spanish	English Friendly	No	Mode of delivery	Face-to-face

Department	DPTO. INGENIERÍA INFORMÁTICA Y ELECTRÓNICA				
Name of lecturer	CARLOS BLANCO BUENO				
E-mail	carlos.blanco@unican.es				
Office					
Other lecturers					

3.1 LEARNING OUTCOMES
- Acquire the principles and foundations of software engineering and its main concepts: life cycle, methodology, process.
- Know and apply a methodology for software development.
- Model functional and non-functional requirements of a software system.
- Model the architectural design of a software system.
- Model the detailed design of a software system.
- Know how to use tools that support the modeling of software systems.

**4. OBJECTIVES**

Understand and know the importance and role of the discipline of software engineering, in order to correctly apply the general principles of engineering to develop software.

Know how to apply a software development process, including the stages of requirements, architecture and detailed design.

Model software using UML.

**6. SUBJECT PROGRAM**

**CONTENTS**

1	FUNDAMENTALS OF SOFTWARE ENGINEERING. Body of Knowledge SWEBOK. Main areas. Modeling in Software Engineering. Modeling Languages: UML. Development methods. Software life cycle and processes.
2	SOFTWARE SYSTEM REQUIREMENTS. Introduction. Concept and characteristics. Stages of Software Requirements Specification: gathering, analysis, specification and validation. Modeling and specification using Use Case Models, Templates and Domain Class Diagrams. Non functional requirements.
3	SOFTWARE SYSTEMS ARCHITECTURE. Software Architecture. Component concept. 3-tier architectures. Modelling architectures by using component and deployment diagrams. Modelling architectural scenarios by using sequence diagrams.
4	DESIGN SOFTWARE. Structural description. Design patterns. Modelling structural design by using class diagrams. Procedural behaviour and modelling by using activity diagrams. Reactive behaviour and modelling by using statechart diagrams.

**7. ASSESSMENT METHODS AND CRITERIA**

Description	Type	Final Eval.	Reassessn	%
Exam	Written exam	Yes	Yes	60,00
Project Development	Work	No	Yes	40,00
<b>TOTAL</b>				<b>100,00</b>
Observations				
Observations for part-time students				
Part-time students will be able to manage their time by combining it with their professional or personal factors. They will have the class material in Moodle and will be able to carry out the development of an evaluable practical project remotely. The final exam must be taken in person on the date set by the center.				

**8. BIBLIOGRAPHY AND TEACHING MATERIALS**

BASIC

Sommerville, 2012. Ingeniería del Software. 9ª Edición, Addison-Wesley. 2012.

J.Arlow e I.Neustadt. UML 2 and the Unified Process: Practical Object-Oriented Analysis and Design. Addison-Wesley. 2005.

D.Pilone. UML 2.0 in a Nutshell. O'Reilly. 2005.

