



DEGREE CURRICULUM

# QUALITY MANAGEMENT AND IMPROVEMENT

Coordination: GONZÀLEZ CAPDEVILA, MARC

Academic year 2023-24

## Subject's general information

<b>Subject name</b>	QUALITY MANAGEMENT AND IMPROVEMENT			
<b>Code</b>	102053			
<b>Semester</b>	2nd Q(SEMESTER) CONTINUED EVALUATION			
<b>Typology</b>	Degree	Course	Character	Modality
	Bachelor's Degree in Computer Engineering	3	COMPULSORY	Attendance-based
	Bachelor's Degree in Computer Engineering	3	OPTIONAL	Attendance-based
<b>Course number of credits (ECTS)</b>	6			
<b>Type of activity, credits, and groups</b>	<b>Activity type</b>	PRALAB	TEORIA	
	<b>Number of credits</b>	3	3	
	<b>Number of groups</b>	1	1	
<b>Coordination</b>	GONZÁLEZ CAPDEVILA, MARC			
<b>Department</b>	COMPUTER ENGINEERING AND DIGITAL DESIGN			
<b>Teaching load distribution between lectures and independent student work</b>	20% On-site 20% Online 60% Autonomous work			
<b>Important information on data processing</b>	Consult <a href="#">this link</a> for more information.			
<b>Language</b>	English			

Teaching staff	E-mail addresses	Credits taught by teacher	Office and hour of attention
GONZÁLEZ CAPDEVILA, MARC	marc.gonzalez@udl.cat	7,2	

## Subject's extra information

This course is part of the specialisation in Software Engineering of the Degree in Computer Engineering offered by the Polytechnic School at the University of Lleida.

A **Joint Project** is developed together with other subjects in the third course of the "Software Engineering" specialization:

- Quality Management and Improvement
- Process Model
- Requirements Engineering

Each subject is focused on its corresponding aspects of the project development. In the case of this subject, the focus is software quality management and improvement. The aim of this **Joint Project** is to bring the students with what would be a real-world project of Software Engineering and at the same time, we see that results of the aspects studied in the contexts of a course serve to meet the challenges presented in other courses.

To follow this course properly, some previous knowledge/skills on Software Engineering are recommended. A good level of English (reading, writing, oral) is also recommended.

## Learning objectives

- To understand that software quality goes beyond the quality of the code.
- To design a reasonably professional and rigorous software quality plan.
- To implement key aspects of a software quality plan, i.e., conduct software quality assurance tasks, in an effective way.
- To be conversant with the field of software quality assurance (metrics, tools, costs, etc.).
- To keep strengthening important skills good software engineers should have (i.e. ability to make decisions, to meet deadline, to work in teams, to write technical documentation and make oral presentations).

## Competences

### Strategic Competences of the UdL

- CT2. Mastering a foreign language, especially English.
- CT3. Training Experience in the use of the new technologies and the information and communication technologies.

### Cross-disciplinary competences

- EPS11. Capacity to understand the needs of the user expressed in a no technical language.

### Specific competences (Software Engineering)

- GII-IS1. Capacity to develop, maintain and evaluate services and software systems that satisfy all the requirements of the user and behave in a reliable and efficient way, they can develop, keep and fulfil quality requirements, applying the theories, principles, methods and uses of the software engineering.
- GII-IS5. Capacity to identify, evaluate and manage the potential risks that can arise.

## Subject contents

### Block I

- Chapter 1. Introduction to Software Quality
- Chapter 2. Quality Management
- Chapter 3. Further Aspects

### Block II

- Chapter 4. Testing and Metrics
- Chapter 5. Introduction to costs
- Chapter 6. Standards and Models

## Methodology

Classes for the course include sessions in which the contents are introduced, through support of educational materials prepared or compiled by the lecturer, and group work sessions.

In group work sessions, we use the project-based learning methodology. So, students must apply techniques, concepts and tools learned in the subject under the common project to the three courses in the third year of Software Engineering module.

### Theory

- Presentation of the contents of the course.
- Discussion with the students.

### Joint Project

# QUALITY MANAGEMENT AND IMPROVEMENT 2023-24

- Deliverable 1. Definition of a Quality Plan. First Iteration - Quality Plan.
- Deliverable 2. Improvement and implementation of a Quality Plan. Second Iteration - Quality Plan.
- Deliverable 2. Improvement and implementation of a Quality Plan. Third Iteration - Quality Plan.
- Oral presentation of the project to a committee involving the professors of the three subjects. The presentation, common to these subjects, addresses the overall project development.

**A note about Joint Project:** Both activities are conducted within the context of the Software Engineering Joint Project developed in three courses (Requirements Engineering, Process Models and Quality Management and Improvement). Given that each course addresses important aspects of Software Engineering (e.g. requirements, agile methodologies and quality), the project aims to encourage students to deal with them in a real-life scenario, which is simulated in this project.

## Development plan

Week	Course Activities	Joint Project Activities	Autonomous Student Work
1	Presentation and Chapter 1	Deliverable 1	Study and Deliverable 1
2	Chapter 1 (continuation)	Deliverable 1	Study and Deliverable 1
3	Chapter 2	Deliverable 1	Study and Deliverable 1
4	Chapter 2 (continuation)	Deliverable 1	Study and Deliverable 1
5	Chapter 2	Deliverable 1 (submission)	Study and Deliverable 1
6	Chapter 3	Deliverable 2	Study and Deliverable 2
7	Chapter 3 (continuation)	Deliverable 2	Study and Deliverable 2
8	Chapter 3	Deliverable 2	Study and Deliverable 2
9	Written Exam	Deliverable 2	Study and Deliverable 2
10	Chapter 4	Deliverable 2 (submission)	Study and Deliverable 2
11	Chapter 4 (continuation)	Deliverable 3	Study and Deliverable 3
12	Chapter 4 (continuation)	Deliverable 3	Study and Deliverable 3
13	Chapter 4 (continuation)	Deliverable 3	Study and Deliverable 3
14	Chapter 5	Deliverable 3	Study and Deliverable 3
15	Chapter 6	Deliverable 3	Study and Deliverable 3
16-18	Written Exam	Deliverable 3 (submission). Final presentation.	Study and Deliverable 3
19	Tutorials		
20	Remedial Exam		Study

## Evaluation

### Continuous assesment:

Block	Evaluation Activity	Rate	Minimum Qualification	Components	Rate of Each Component	Group Activity	Remedial Exam
B1	Deliverable 1	20%	NO			YES	NO
B2	Deliverable 2	20%	NO			YES	NO
B3	Deliverable 3	20%	NO			YES	NO
B4	Final Presentation	10%	NO			YES	NO
B5	Theory	30%	5	1st Written Exam	50%	NO	YES
				2nd Written Exam	50%	NO	YES

$$\text{Final Qualification} = 0,2 * B1 + 0,2 * B2 + 0,2 * B3 + 0,1 * B4 + 0,3 * B5$$

### Continuous assesment:

Block	Description	Weight	Mandatory	Recoverable	Ind/Grup
Unique Block	Unique Exam <ul style="list-style-type: none"> <li>• Chapters 1, 2, 3, 4, 5 and 6</li> </ul>	100%	Yes	Yes	Ind

The course is passed with a qualification  $\geq 5$ .

## Bibliography

**Mario Piattini, Félix García, Ignacio García.** Calidad de Sistemas de Información. 3rd edition. Ra-Ma Editorial. 2015.

**Ian Sommerville.** Software Engineering. Ninth version. Addison-Wesley, 2011.

**Ronald Kirk.** Software Engineering Quality Practice. Auerback Publications, 2006.

**Jeff Tian.** Software Quality Engineering: Testing, Quality Assurance, and Quantifiable Improvement. Wiley - IEEE Computer Society Press, 2005.

**Daniel Galin.** Software Quality Assurance. From theory to implementation. Pearson Education Limited, 2004.

**Daniel Galin.** Software Quality: Concepts and Practice. From theory to implementation. Wiley-IEEE Computer Society, 2018

**Stephen Kan.** Metrics and Models in Software Quality Engineering. 2nd Edition. Addison-Wesley, 2002.

**Gordon Schulmeyer.** Handbook of Software Quality Assurance. 4th edition. Artech House, 2002.