Universitat de Lleida

DEGREE CURRICULUM

ANALYSIS, MODELING AND DESIGN OF INFORMATION SYSTEMS

Coordination: OLIVA SOLE, MARTA

Academic year 2023-24

Subject's general information

Subject name	ANALYSIS, MODELING AND DESIGN OF INFORMATION SYSTEMS
Code	102032
Semester	2nd Q(SEMESTER) CONTINUED EVALUATION

Typology	Degree	Course	Character	Modality
	Double bachelor's degree: Degree in Computer Engineering and Degree in Business Administration and Management	4	OPTIONAL	Attendance-based

Course number of credits (ECTS)	6

Type of activity, credits, and groups	Activity type	PRALAB	TEORIA
	Number of credits	4	2
	Number of groups	1	1

Coordination	OLIVA SOLE, MARTA
Department	COMPUTER ENGINEERING AND DIGITAL DESIGN
Teaching load distribution between lectures and independent student work	ECTS 25x6 = 150 = 6h - 40% -> 60 classroom hours. - 60% -> 90 hours autonomous work
Important information on data processing	Consult this link for more information.
Language	Catalan and English. The materials and resources will be provided in English, but the lectures will be in Catalan.

Teaching staff	E-mail addresses	Credits taught by teacher	Office and hour of attention
OLIVA SOLE, MARTA	marta.oliva@udl.cat	6	arrange by email

Subject's extra information

Course taught during the 2nd semester of 4th year of the degree, bearing in mind that to develop good software their requirements should be well identified, documented and validated.

A **Joint Project** is developed together with another subject in the 4th course:

- Management, Organization and Leadership of Technological Projects
- Analysis, Modeling and Design of Information Systems

Each subject focuses on the corresponding aspects of project development. In the case of this subject, requirements elicitation and specification. 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.

The knowledge acquired in this course will be applicable in most careers, especially for those who are dedicated to develop applications.

(*) *These two subjects must be enrolled simultaneously since they work on a common project. The only exception is if any of them has been already passed.*

Learning objectives

- Understand the importance of discovering the requirements of a project.
- Determine the scope of a project.
- Practice different techniques for requirements elicitation (functional and non functional).
- Understand what involves an iterative process for requirements engineering.
- Describe scenarios.
- Specify requirements through user stories.
- Analyze the quality of each specific requirement.
- Assess the specification requirements.
- Analyze the risks.

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

- 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-IS2. Capacity to value the customer needs and specify the software requirements to satisfy these needs, solving conflicts by means of acceptable commitments taking into account the limitations of cost, time, existence of systems already developed and of the own organisations.
- GII-IS3. Capacity to give solution to problems of integration taking into account the strategies, standards and available technologies.

Subject contents

- *Why developing software needs more time or more money than you initially expected?*
- *Why are there so many programs that, upon completion of their development, fail?*
- *Why users are usually not completely satisfied by software?*

The answer to these questions is related to how to include requirements engineering in software development. Not doing so usually leads to uncomplete or unclear requirements and, consequently, to software that does not satisfy users.

The contents of this course include various aspects to be treated in the requirements engineering phase, among other.

1. Introduction
2. The Requirement Process
3. Scoping the Business Problem
4. Requirements and Iterative Development
5. Investigating the work - Trawling techniques
6. Scenarios
7. Non-Functional Requirements - Fit criterion
8. User Stories
9. Quality Gateway
10. Requirements evaluation - Conflicting Requirements, Risk Analysis

Methodology

The classes of the subject include sessions in which the contents are debated and group work sessions in person (PraAula). In group work sessions, the project-based learning methodology is used. Thus, the students put into practice the techniques proposed in the subject within the framework of a project common to the three third-year subjects of the intensification of Software Engineering.

Theory

- Requires prior review of teaching materials prepared or compiled by teachers.
- Discussion / Debate with the students of the contents corresponding to the session. The session is conducted in such a way that all the most relevant contents are reviewed and that they will end up being assumed during the development of the proposed project.

Team work

- From the proposed project, the students progress in small weekly activities that end up forming each of the required deliverables.

Joint Project

- Deliverable 1. Scope of the project and First Iteration Requirements Modelling
- Deliverable 2. Second and third iteration Requirements Modelling
- Deliverable 3. Requirements Evaluation & Risk Analysis
- Oral presentation of the project to a committee involving the professors of all 3 subjects. The presentation, common to these subjects, addresses the overall project development.

A note about Joint Project: all 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 course). 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	Introduction. The Requirement Process. Scoping the Business Problem	Project Presentation Project launch (Scope, Goals, Stakeholders and Definitions)	Materials Review
2	Requirements and Iterative Development Investigating the work. Trawling Techniques	Context diagram (Data Flow, Events) Trawling Techniques Plan	Materials review, Joint project development
3	Scenarios. Non-Functional Requirements - Fit Criterion	Requirements Trawling	Materials review, Joint project development
4	(project development)	Functional and Non Functional Requirements Modelling - 1rst sprint	Joint project development
5	Starting the solution. User Stories	Prototypes and User Stories - 1rst sprint	Materials review, Joint project development
6	Quality Gateway	Trawling, Functional and Non Functional Requirements Modelling - 2on sprint	Project Development
7	(project development)	Prototypes, User Stories and Qualitu gateway - 2on sprint	Study, Joint Project Development
8	1rst Midterm Exam		Study
9		Trawling, Functional and Non Functional Requirements Modelling - 3th sprint	Joint Project Development
10	(project development)	Prototypes, User Stories and Qualitu gateway - 3th sprint	Joint Project Development
11	Requirements Evaluation (Requirements Completeness)	Requirements Evaluation/Modelling (Requirements Completeness)	Materials Review, Joint Project Development
12	(project development)	Requirements Evaluation/Modelling (Requirements Completeness)	Joint Project Development
13	Requirements Evaluation (Conflicting Requirements)	Requirements Evaluation/Modeling (Conflicting Requirements)	Materials Review, Joint Project Development
14	Requirements Evaluation (Risk Analysis)	Requirements Evaluation/Modeling (Risk Analysis)	Joint Project Development
15	(project development)	Requirements Evaluation/Modelling (Risk Analysis)	Joint Project Development

16	(project development)	Requirements Evaluation/Modelling (Risk Analysis)	Joint Project Development
17-18	2on Midterm Exam		Study
19		Project Presentation	Presentations Preparation
20	Remedial Exam		Study

Evaluation

Acr.	Evaluation Activity	Rate	Minimum Qualification	Group Activity	Mandatory	Remedial Exam
E	Exam (17-18 weeks)	30%	4	NO	YES	YES
D1	1st Project Deliverable	20%	NO	4 to 6	YES	NO
D2	2a Project Deliverable	20%	NO	4 to 6	YES	NO
D3	3a Project Deliverable	20%	NO	4 to 6	YES	NO
FP	Final Presentation	10%	NO	4 to 6	YES	NO
Final Qualification = 0,3*E + 0,2*D1 + 0,2*D2 + 0,2*D3 + 0,1*FP						

Bibliography

Suzanne Robertson & James Robertson. *Mastering the Requirements Process: Getting Requirements Right* (3rd ed.). Addison-Wesley, 2012.

Dean Leffingwell. *Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise* , Addison Wesley, 2011.

Mike Cohn. *User Stories Applied: For Agile Software Development*, Addison Wesley, 2004.

Klaus Pohl & Chris Rupp. *Requirements Engineering Fundamentals: A Study Guide for the Certified Professional for Requirements Engineering Exam - Foundation Level - IREB compliant*. Rocky Nook Computing, 2011.

Elizabeth Hull, Ken Jackson & Jeremy Dick. *Requirements Engineering* (3rd edition), Springer, 2011.

Klaus Pohl. *Requirements Engineering. Fundamentals, Principles, and Techniques*. Springer, 2010.

Karl Wiegers & Joy Beatty. *Software Requirements* (3rd edition). Microsoft Press, 2013.

Ralph M. Stair & George W. Reynolds. *Principles of Information Systems*. (13th edition). Course Technology, 2017.

R. Kelly Rainer Jr., Brad. Prince & Casey Cegielski. *Introduction to Information Systems. Suporting and Tranforming Business* (5th edition). Wiley, 2014.

Pedro Isaias & Tomayess Issa. *High Level Models and Methodologies for Information Systems*. Springer, (2015).