



Universitat de Lleida

DEGREE CURRICULUM
**SPECIFICATION AND ANALYSIS
OF INTERACTIVE SYSTEMS**

Coordination: SAYAGO BARRANTES, SERGIO

Academic year 2021-22

Subject's general information

Subject name	SPECIFICATION AND ANALYSIS OF INTERACTIVE SYSTEMS			
Code	102382			
Semester	2nd Q(SEMESTER) CONTINUED EVALUATION			
Typology	Degree	Course	Character	Modality
	Bachelor's degree in Digital Interaction and Computing Techniques	2	COMPULSORY	Attendance-based
Course number of credits (ECTS)	6			
Type of activity, credits, and groups	Activity type	PRALAB	TEORIA	
	Number of credits	3	3	
	Number of groups	1	1	
Coordination	SAYAGO BARRANTES, SERGIO			
Department	COMPUTER SCIENCE AND INDUSTRIAL ENGINEERING			
Teaching load distribution between lectures and independent student work	<p>According to the academic framework of bachelor's degrees of the EPS:</p> <ul style="list-style-type: none"> - 1 ECTS = 25 hours; 6 ECTS = 150 hours - 40% (60h) of in-class work and 60% (90h) of autonomous work <p>The distribution of hours in this course is:</p> <ul style="list-style-type: none"> - In-class work: theory (28h) + laboratories (28h) = 56 h + 4 hours of exams = 60h - Autonomous work: study (45h) + project (45h) = 90h 			
Important information on data processing	Consult this link for more information.			
Language	English, Spanish and Catalan			
Distribution of credits	See type of activity, credits and groups			

Teaching staff	E-mail addresses	Credits taught by teacher	Office and hour of attention
SAYAGO BARRANTES, SERGIO	sergio.sayago@udl.cat	6	Make an appointment via e-mail. Office 12 Campus Igualada (Pla de la Massa) / Online via videoconference

Subject's extra information

Software dominates the world and the development of professional software is a formal task. The Specification and Analysis of Interactive Systems (SPAIS) is a mandatory course of 6 ECTS. SPAIS is held during the second semester of the second year of the Bachelor's degree in Digital Interaction and Computing Techniques (GTIDIC).

GTIDIC aims to train qualified professionals in the computing field with a very practical side, giving special emphasis to the design and implementation of interactive applications. The graduates will acquire solid programming knowledge, focusing on mobile and web applications, Internet technologies, administration tools and security systems, and interface design and development.

SPAIS is designed to provide students with a guided and hands-on introduction to Software Engineering, given special emphasis to the design (specification and analysis) of interactive systems, so that students can integrate themselves in professional software engineering teams quickly. To achieve this objective, the student will carry out a project in which they will apply aspects related to Requirement Engineering, and other important elements of Software Engineering, such as domain analysis, UML (Unified Modelling Language), standards and quality in software, and management of projects. The project will be conducted alongside three other subjects (Applications for Mobile Devices, Innovation in ICT, and User Experience), to provide students with a real software engineering experience, and an integrated perspective.

SPAIS complements Object Oriented Programming and Application Platforms, providing students with a Software Engineering perspective, and is an introduction to Interactive Applications Design, where students will learn further aspects about design patterns.

Learning objectives

1. To know basic aspects of Software Engineering and development process models
2. To be able to design and create use cases
3. To gather and specify different types of requirements in a rigorous and professional way
4. To design the class diagram of an interactive system following the paradigm of object oriented design
5. To be able to use UML modelling tools
6. To understand software code as something that evolves over time
7. To be able to write unit tests
8. To understand the key tenets of object oriented design
9. To understand the concept of responsible design in object oriented design

Competences

According to the table of competences of the GTIDIC (<https://ja.cat/zvyK4>):

Basic competences

CB3. That students have the ability to gather and interpret relevant data (usually within their area of study) to make judgments that include a reflection on relevant social, scientific or ethical issues.

Transversal competences

CT3. Acquire training in the use of new technologies and information and communication technologies

General competences

CG1. Capacity to conceive, plan and develop projects in the field of ICT

CG2. Capacity to design, develop, evaluate and ensure the accessibility, ergonomics, usability and security of computer systems

CG4. Capacity to use software engineering methods in the development of interactive computer applications.

CG9. Capacity to analyze and synthesize

Specific competences

CE3. Basic knowledge of the use and programming of computers, operating systems and databases, and their use in the development of interactive applications.

CE6. Capacity to design, develop, select and evaluate applications and computer systems, ensuring its reliability, security and quality.

CE10. Capacity to analyse, design, build and maintain safe and efficient applications, choosing the most suitable paradigm and programming languages.

CE13. Knowledge and application of the characteristics, functionalities and structure of the databases, that allow their suitable use, and the design and the analysis and implementation of interactive applications based on them.

CE15. Knowledge and application of the principles, methodologies and life cycles of software engineering

CE16. Capacity to design and evaluate person-computer interfaces that guarantee the usability of systems, services and computer applications

CE25. Being able to analyze, organize, label and visualize the structure that defines the interaction with digital content, through the application of information architecture methods, techniques and tools that facilitate accessibility

CE26. Knowing how to apply the principles and standards of accessibility and universal design of the main digital products and services to design experiences that guarantee equal opportunities among their users.

Subject contents

- Key concepts of Software Engineering
- Requirements engineering (analysis and specification)
- Domain analysis
- Introduction to test-driven development
- Principles of software design

Methodology

Methodology	Theory	Laboratory	Independent student's work
Lectures	X		
Integrated project (in laboratories)		X	
Integrated project (at home)			X
Study			X

Both plenary and laboratory sessions will be conducted face-to-face unless the academic authorities at the UdL

indicate that these activities must be virtual.

Integrated project

The laboratories are carried out within the context of an integrated project. This project is conducted in four courses: Innovation, Mobile Development, Specification and Design of Interactive Systems, and this course.

The integrated project aims to have students work in an agile software development project, which is designed to set up an start-up by designing and developing an mobile app. The project also aims to enable students to develop important skills and competences, such as being able to work in teams, present ideas in public, and team organization.

Development plan

Week	Dates	Theory	Labs	Comments
1	7 Feb - 10 Feb	Presentation T1. Intro. Software Eng.	P1. Some introductory aspects	
2	14 Feb - 17 Feb	T2. Intro. Req. Eng.	P1 (cont.)	Submission P1 20 Feb 23.55
3	21 Feb - 24 Feb	T2 (cont.) T3. Modelling	P2. Requirements + modelling	
4	28 Feb - 3 Mar	T2 (cont.) T3 (cont.)	P2 (cont.)	Submission P2 6 March 23.55
5	7 Mar - 10 Mar	T2 (cont.) T4. Proces models	P3. Domain analysis + modelling	
6	14 Mar - 17 Mar	T4 (cont.) T5. Software design principles (SOLID, GRASP)	P3 (cont.)	Submission P3 20 March 23.55
7	21 Mar - 24 Mar	P4. DAFO + evolution requ. + modelling	HOLIDAYS	
8	28 Mar - 31 Mar	Mock-up examen_1	P4 (cont.)	Submission P4 3 April 23.55
9	4 Apr - 7 Apr	EXAMS		
10	18 Apr - 21 Apr	FESTIU	P5. Quality + Agile, Part I	
11	25 Apr - 28 Apr	T6. Intro. to quality	P5 (cont.)	Submission P5 1 May 23.55
12	2 May - 5 May	T7. Intro. to software project management	P6. Quality + Agile, Part II	
13	9 May - 12 May	P6 (cont.)	P6 (cont.)	Submission P6 15 May 23.55
14	16 May - 19 May	P7. Quality + Agile, Part III	P7 (cont.)	
15	23 May - 26 May	P7 (cont.)	Pt (cont.)	Submission P7 29 May 23.55
16	30 May - 2 Jun	EXAMS		
17	6 Jun - 9 Jun	EXAMS		

18	13 Jun - 16 Jun	Office hours		
19	20 Jun - 23 Jun	Retakes		
20	27 Jun - 30 Jun	Retakes		

The development plan of the integrated project is available at the Campus Virtual.

Evaluation

The assessment instruments and its relationship with the learning objectives and specific competences are listed below:

Instrument	Learning objectives	Specific competences
Integrated project	All	6-10-15-16
Exams	1-2-3-6-8-9	6-10

The assessment of this course is detailed next:

Continuous evaluation
<p>Final Grade (FG) = Theory * 0.5 + Labs* 0.5, FG >= 5 Theory (50% of the FG) >= 5</p> <ul style="list-style-type: none"> • (30% of the FG) <u>First exam</u>. 2h. Written exam. <ul style="list-style-type: none"> ◦ Observation: The final grade of the first exam = (Mock-up exam_1, first exam) • (20% of the FG) <u>Second exam</u>. 2h. Written exam. • NP (50% of the FG) >= 5 <ul style="list-style-type: none"> ◦ (30% of the FG) Labs <ul style="list-style-type: none"> ▪ $(P1 + P2 + P3 + P4 + P5 + P6 + P7) / 7$ ◦ (20% of the FG) Integrated project <ul style="list-style-type: none"> ▪ <u>Presentation</u> (10%) ▪ <u>Progress and management</u> (10%)
<p>Retakes - minimum grade = 5. Maximum grade = 7.5</p> <ul style="list-style-type: none"> • Retakes are not for getting a higher grade • If <i>Theory</i> < 5: <u>Final exam</u>. 2 hours. Written exam. • If <i>Laboratories</i> < 5: <u>Submission of all labs</u>

Assessment activities are conducted face-to-face unless the academic authorities at the UdL indicate that they must be virtual.

Integrated project

The integrated project corresponds to 20% of the FG. It consists of two parts:

- presentation. Public presentation. 10%
- evolution and management. We will assess the progress of the teams. 10%

The grade of the integrated project will be the same for all the courses.

At the end of the course, the students will be provided with a summary report with information regarding their progress in the project.

Note 1: If a student is not enrolled in all courses, the student will be evaluated according to his or her work in the courses in which s/he is enrolled. Thus, it is possible to carry out the project without being enrolled in the four courses.

Note 2: If a group splits up, the grades will be kept for those activities that have already been assessed, and the members will carry on working on the project in different groups by drawing on the common aspects

Bibliography

[Applying UML and patterns : an introduction to object-oriented analysis and design and iterative development / Craig Larman](#)

[Requirements engineering / Jeremy Dick, Elizabeth Hull, Ken Jackson](#)

[Software engineering / Ian Sommerville](#)

[Software quality assurance : from theory to implementation / Daniel Galin](#)