

Degree programme	Type	Course
Computer Engineering	OB	3
Computer Engineering	OP	4

Contact lecturer

Name : Marc Tallò Sendra

Email : marc.tallo@uab.cat

Group languages

You can consult this information at the [end](#) of the document.

Prerequisites

There are no prerequisites to take this course, but it is also advisable to study the subjects:

- Architecture and software technology
- Integrated software laboratory

Objectives

In the software engineering subject of the second course, you have received an overview of the software engineering process. You have learned the basics, and for those who have studied the subjects of the first semester of the mention of software engineering, you would have deepened into some of these concepts, such as the analysis of requirements, design and test and control techniques of quality.

In this subject you will learn how to deal with the changes. Every software development process is subject to changes of all kinds. Changes might be caused by the need for improvement of the developed product. Changes in initial requirements. Changes in terms or work team, budget, etc. In short, at the end of this course you will have to be able to manage any change that supposes a deviation from the initial planning.

More specifically, the objectives of this course are:

- Deepen in software control and development techniques.
- Deepen in the knowledge of the administration techniques necessary to plan, monitor and control software projects.
- Learn to use some of the existing tools to carry out the necessary control of the software development.

Finally, for the correct follow-up of this course, the teaching team assumes that all students have achieved the skills explained on software engineering.

Learning outcomes

1. Develop a capacity for analysis, synthesis and prospection.
2. Work independently.
3. Define and manage the documentation generated during the development of a software application.
4. With initiative and autonomy, resolve problems with the integration of software.
5. Plan the integration of the different components developed in the coding process.

6. Develop a mode of thought and critical reasoning.

Contents

BLOCK 1: Project Management (PM)

1. Introduction. Concepts of the PM

Tasks and activities. Products, packages and roles. Tasks models. Profiles matrix. Organization structures: Type and visualization.

2. PM activities

Planning, organization, control and completion.

3. PM methodologies

Royce Iterative Methodology and incremental development. Agile development.

BLOCK 2: Configuration Management (CM)

1. Introduction. Concepts of the CM

CM and aggregate items. Versions and configurations. Requests for changes. Promotions and releases. Repositories and work spaces. Version identification schemes. CM tools.

2. Activities of the CM

Configuration and identification items of the aggregates of the CM. Management of the promotion. Release management. Management of the branches. Management of variants. Change management.

3. Managing the CM

Documentation of the CM. Assignment of responsibilities in the CM. Organize the activities of the CM.

Learning activities and methodology

Title	Hours	ECTS	Learning outcomes
Theory classes	13	0.52	1, 3, 4, 5, 6
Preparation of exercises and resolution of theoretical assumptions	15	0.6	2, 4, 6
Tutorials	9	0.36	3, 4, 5
Preparation and completion of practical exercises	15	0.6	2, 3, 4, 5
Practical sessions	6	0.24	1, 3, 5, 6
Problem classes	13	0.52	4

The student's learning process will be based on these two types of activities:

- Theory and Problems. With the case study approach, students will practice in class the theoretical concepts that will be introduced at the beginning of the class through documentary and whiteboard support. Next, the students will work in small groups and will discuss among them the possible solutions until concluding and choosing one of them. In the final section of the class, all the solutions proposed will be shared.
- Practices. The practical classes will work in groups of two people and will be proposed a project plan

and the use of the necessary configuration management tools in all software development projects will be practiced. Attendance is mandatory for practice sessions.

All the documentation and material needed to track the course will be available on the Virtual Campus.

Use of AI for document generation and management, analysis, implementation, integration and testing:

In this subject, the use of Artificial Intelligence (AI) technologies is permitted as an integral part of the development of the work, provided that the final result reflects a significant contribution of the student in the analysis and personal reflection. The student must clearly identify which parts have been generated with this technology, specify the tools used and include a critical reflection on how these have influenced the process and the final result of the activity. The lack of transparency in the use of AI will be considered a lack of academic honesty and may lead to a penalty in the grade of the activity, or greater sanctions in serious cases.

Annotation: within the schedule set by the centre or degree programme, 15 minutes of one class will be reserved for students to evaluate their lecturers and their courses or modules through questionnaires.

Assessment

Continuous assessment activities

Title	Weight	Hours	ECTS	Learning outcomes
Reports / work practices	50%	2	0.08	1, 2, 3, 4, 5, 6
Exam of Theory	50%	2	0.08	1, 2, 3, 4, 5, 6

The evaluation of the subject consists of two parts: theory (N_{Teo}) and practices (N_{Lab}).

The weight of each of these evaluations is as follows:

- $N_{Final} = 0.5 * N_{Teo} \text{ (Individual)} + 0.5 * N_{Lab} \text{ (In a group)}$
- $N_{Lab} = 0.5 * N_{Lab1} \text{ (SPMP)} \text{ (In a group)} + 0.5 * N_{Lab2} \text{ (GC)} \text{ (In a group)}$

To calculate the N_{Final} note, the marks of each one of the different evaluations (N_{Teo} and N_{Lab}) should be equal or greater than 5. Otherwise, the subject is not passed and the final grade of the subject will be the minimum mark between N_{Teo} and N_{Lab}.

If N_{Teo} is less than 5 in the final examen, the student can have a second chance in the recovery exam.

To calculate the N_{Lab} note, each and every one of the N_{Lab} (i) marks, that is, each and every one of the practice deliveries, must be higher than 4. Otherwise, N_{Lab} is the minimum note of the notes N_{Lab} (i). The evaluations through practical cases and the writing of practical reports are part of the practice evaluation block (N_{Lab}). The delivery dates and the delivery method of the different case studies will be recorded in the same statements. There is no second call for the N_{Lab} note or any of the N_{Lab} (i) notes, that is, there is no second call for the laboratory sessions.

For each assessment activity, a place, date and time of revision will be indicated in which the student will be able to review the activity with the teacher. In this context, claims can be made about the activity note, which will be evaluated by the teachers responsible for the subject. If the student does not submit to this review, this activity will not be reviewed later.

Single evaluation: The student who opts for the single evaluation must carry out the same practical activities as in continuous evaluation, handing them in on the day of the theoretical exam. For the practical evaluation, the student must make an oral defence of the work carried out before the person incharge of the subject. The evaluation of this defence will be considered along with the evaluation of the practicals delivered. The theoretical exam will be the same as for the continuous evaluation. The recovery exam will be the same as for the continuous evaluation.

A student is considered Non-Appraising only if the assessment activities he has undertaken are approved (but he has not done all the activities of the course evaluation). If an evaluation activity is not passed, the subject is not passed (regardless of the number of activities that have been presented).

The honor grade can be obtained with a mark average superior or equal to 9.0. Due to the fact that there is a limited number of honor grades that can be given by group, they will be awarded in order of note from major to minor. Students who have not attended all of the of the practical classes or have to retake any exams are not eligible for a distinction.

Only the practice note (NLab) of previous years will be kept between years.

We will notify the students using the Virtual Campus where they will always be informed about possible changes in the course's development since it is understood to be the usual mechanism for exchanging information between teacher and students.

Notwithstanding other disciplinary measures deemed appropriate, and in accordance with the current academic regulations, irregularities committed by a student that may lead to a variation of the qualification will be classified by zero (0). For example, plagiarizing, copying, copying, ... will involve suspending this evaluation activity with zero (0). Assessment activities qualified in this way and by this procedure will not be recoverable. If it is necessary to pass any of these assessment activities to pass the subject, this subject will be suspended directly, without opportunity to recover it in the same course.

Bibliography

- Anderson, David J. (2004). Agile management for software engineering : applying the theory of constraints for business results. Prentice Hall PTR - [Available](#) in hard copy from the library
- Berczuk, Stephen P. & Appleton, Brad. (2003). Software configuration management patterns : effective teamwork, practical integration. Addison-Wesley - [Available](#) in hard copy from the library
- Bruegge, Bernd & Dutoit, Allen H. (2010). Object-oriented software engineering : using UML, Patterns, and Java. (3rd ed.) Pearson - [Available](#) in hard copy from the library
- McConnell, Steve. (1996). Rapid development : taming wild software schedules. Microsoft Press - [Available](#) in hard copy from the library
- Reifer, Donald J. & Rutkowski, Michael. (2013). Software war stories : case studies in software management . IEEE Xplore; Wiley - [Available](#) online
- Stephens, Rod. (2023). Beginning software engineering. (2nd ed.) John Wiley & Sons - [Available](#) in hard copy from the library
- Stephens, Rod. (2015). Beginning software engineering. (1st ed.) Wrox - [Available](#) online

Software

GIT:

- Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.
- <https://git-scm.com>

BITBUCKET:

- Bitbucket is more than just Git code management. Bitbucket gives teams one place to plan projects, collaborate on code, test, and deploy.
- <https://bitbucket.org/product>

SUITE MICROSOFT OFFICE:

- Word, Excel, PowerPoint and Project.
- <https://www.microsoft.com/es-gq/store/collections/officesuites>

CANVAN:

- Project planning.
- <https://www.canva.com/education/students>

TRELLO:

- Trello is a client and web interface project management software for iOS and android for organizing projects.
- <https://trello.com>

MONDAY PROJECT MANAGEMENT:

- Project management software, from simple to complex projects, with monday.com.
- <https://monday.com>

AI TOOLS:

- ChatGPT / Claude / Gemini (General Support and Reasoning)
- Microsoft Copilot / Google Workspace (Corporate Productivity)
- Cursor / Claude Code / Lovable (Software Development)
- Make / n8n / Power Automate and Dynamics 365 - Microsoft (Process Automation)
- NotebookLM / Notebook - Microsoft (Knowledge Management and Study)

Course groups and languages

The information provided is provisional until November 30. After this date, you will be able to consult the language of each group through this [link](#). To access the information, you will need to enter the course CODE

Type of teaching	Group	Language	Semester	Shift
(TE) Theory	420	Catalan	second semester	morning-mixed
(PAUL) Classroom practices	421	Catalan	second semester	morning-mixed
(PLAB) Practical laboratories	421	Catalan	second semester	afternoon
(PAUL) Classroom practices	422	Catalan	second semester	morning-mixed
(PLAB) Practical laboratories	422	Catalan	second semester	afternoon
(PLAB) Practical laboratories	423	Catalan	second semester	morning-mixed
(PLAB) Practical laboratories	424	Catalan	second semester	morning-mixed