

Academic Year: (2022 / 2023)

Review date: 03-06-2022

Department assigned to the subject: Computer Science and Engineering Department

Coordinating teacher: SANCHEZ SEGURA, MARIA ISABEL

Type: Compulsory ECTS Credits : 6.0

Year : 4 Semester : 2

REQUIREMENTS (SUBJECTS THAT ARE ASSUMED TO BE KNOWN)

Software Engineering (course: second- semester: first)

OBJECTIVES

Learn and apply concepts that enable the integrated application of management, control and development processes in the software development life cycle.

BASIC AND GENERAL COMPETENCIES

CG1 - Apply appropriate theoretical and practical methods for the analysis, design and solution of problems, providing IT solutions

that respect accessibility, ergonomics and safety at work standards and that comply with the legislation existing.

CG4 - Technical, economic and commercial management of IT projects, planning proposals, organizing teams and applying engineering techniques that are rigorous, responsible and that respect current regulations and are in accordance with ethics professional.

CG5 - Use IT tools, general purpose, collaborative and work optimization for planning and effective implementation of projects.

CG9 - Efficiently use IT means to write technical reports and reports on projects and work on Computer science, as well as quality presentations.

CGO1 - Ability to conceive, draft, organize, plan, develop and sign projects in the field of engineering in computer science whose object, in accordance with the knowledge acquired, the conception, development or exploitation of computer systems, services and applications.

CGO2 - Ability to direct the activities that are the object of projects in the field of information technology in accordance with the acquired knowledge.

CGO7 - Ability to know, understand and apply the necessary legislation during the development of the profession of Engineer

Computer technician and handle specifications, regulations and mandatory standards.

CGO3 - Ability to design, develop, evaluate and ensure the accessibility, ergonomics, usability and security of the systems,

computer services and applications, as well as the information they manage.

CGO4 - Ability to define, evaluate and select hardware and software platforms for the development and execution of systems,

computer services and applications, in accordance with the knowledge acquired.

CGO5 - Ability to conceive, develop and maintain computer systems, services and applications using the methods of software engineering as an instrument for quality assurance, in accordance with the knowledge acquired.

CB2 - That students know how to apply their knowledge to their work or vocation in a professional way and possess the

competencies that are often demonstrated through the development and defense of arguments and problem solving within

your study area

CROSS COMPETENCES

CECRI1 - Ability to design, develop, select and evaluate computer applications and systems, ensuring their reliability, safety and quality, in accordance with ethical principles and current legislation and regulations.

CECRI2 - Ability to plan, conceive, deploy and direct projects, services and computer systems in all areas, leading its start-up and continuous improvement and assessing its economic and social impact.

CECRI3 - Ability to understand the importance of negotiation, effective work habits, leadership and communication skills in all software development environments.

CECRI4 - Ability to prepare the technical specifications of a computer installation that meets the standards and current regulations.

CECRI8 - Ability to analyze, design, build and maintain applications in a robust, safe and efficient way, choosing the paradigm and the most suitable programming languages.

CECRI16 - Knowledge and application of the principles, methodologies and life cycles of software engineering.

DESCRIPTION OF CONTENTS: PROGRAMME

1. Modern software development methodologies
2. Interrelation between engineering processes, control and management processes in the software development lifecycle
3. Software or service initial offer
4. Software configuration management
5. Quality management
6. Viability analysis
7. Software projects estimation techniques
8. Software projects planning techniques
9. Software projects analysis techniques
10. Software projects design techniques
11. Tools to support management, control and development processes

LEARNING ACTIVITIES AND METHODOLOGY

The teaching methodology is based on theory lectures and group work by students . It is pertinent to make the practice team to promote the education , training and information potential of group work : quantity; quality ; creativity ; structuring and mental flexibility . Additionally , he serves as training students for their future professional work , then, as you know , most of the time, Computer and Software Engineering , always working as a team .

The students' work is performed externally to the classes and these are used to correct errors and detect improvements in work groups and individual analysis of pupils

Group work of the assigned part . This step must be performed by each group so independently. This second step of the methodology is fully active , unschooled and individualized to the group ; ie self-taught.

Analysis and discussion groups in the classroom, the issues worked . This analysis will involve the decomposition of problems into subproblems , modularization of tasks, etc. .

Sharing of the findings of each group and partial synthesis . These two steps of the method are described that take place simultaneously in the same work session.

Theory Classes: 1.5 ECTS . Aim to achieve specific cognitive skills of the subject.

Practical lessons : 1.5 ECTS . Develop generic and specific competencies explicit by performing an extended case group that brings together the entire syllabus of the course in one year whose resolution is defended on public display.

Making Targeted Academic Activities .

- With the presence of the teacher : 1.5 Working ECTS deepening in some aspect in relation to the theme of the course will culminate with a public presentation of the results thereof.

- No teacher presence : 1.5 ECTS . Exercises and readings proposed by the teacher .

Tutorials: Weekly the teacher makes available to the student 2 hours for possible tutorials that may arise as the subject matter is being worked on.

ASSESSMENT SYSTEM

Final examination will not be realized for the students who follow continuous assessment

Exercises and examinations are both learning and evaluation activities. The evaluation system includes

the assessment of guided academic activities and practical cases, with the following weights:

Practical case: 50%

Guided academic activities:

- Present teacher: 25%
- Absent teacher: 25%

Practical lectures: Develop generic and specific skills explicit by performing a group event that brings together extended the entire agenda of the subject in a single year whose decision is upheld in public.

Guided academic activities.

- With the presence of the teacher: Deepening in some way related to the theme of the course will culminate with public presentation of the results.
- No teacher presence: Teamwork and Conflict Management / Effective Communication

% end-of-term-examination:	0
% of continuous assessment (assignments, laboratory, practicals...):	100

BASIC BIBLIOGRAPHY

- Dr David Tuffley Software Configuration Management: A How To Guide for Project Staff, Createspace , 2011
- Dr David Tuffley Software Configuration Management: A How To Guide for Project Staff, Createspace , 2011
- Dr David Tuffley Software Configuration Management: A How To Guide for Project Staff, Createspace , 2011
- Aybüke Aurum Managing Software Engineering Knowledge, Springer, 2003
- Craig Larman Applying UML and patterns: an introduction to object-oriented analysis and design and iterative development, Prentice Hall, 2005
- Craig Larman Applying UML and patterns: an introduction to object-oriented analysis and design and iterative development, Prentice Hall, 2005
- Craig Larman Applying UML and patterns: an introduction to object-oriented analysis and design and iterative development, Prentice Hall, 2015
- María Isabel Sánchez-Seguram Fuensanta Medina-Domínguez, Antonio de Amescua-Seco, Jose-Arturo Mora-Soto. The software configuration management process. A practical approach. , CreateSpace., 2020
- Pressman, R., & Maxim, B Software engineering: A practitioner's approach (8th ed.), McGraw-Hill., 2015
- Richardson, G. Project management theory and practice, Auerbach Publications, 2015
- Richardson, G. Project management theory and practice, Auerbach Publications, 2015
- Watts S. Humphrey Managing Technical People: Innovation, Teamwork and the Software Process (SEI), Addison-Wesley Educational Publishers Inc, 1996