

Course Unit	Software Development and Technologies		Field of study	Informatics	
Master in	Informatics		School	School of Technology and Management	
Academic Year	2023/2024	Year of study	2	Level	2-2
Type	Semestral	Semester	1	ECTS credits	6.0
Workload (hours)		162	Contact hours	T -    TP 60    PL -    TC -    S -    E -    OT -    O -	
Code: 5060-710-2102-00-23					

T - Lectures; TP - Lectures and problem-solving; PL - Problem-solving, project or laboratory; TC - Fieldwork; S - Seminar; E - Placement; OT - Tutorial; O - Other

Name(s) of lecturer(s) Rui Pedro Sanches de Castro Lopes, Pedro Filipe Fernandes Oliveira

### Learning outcomes and competences

At the end of the course unit the learner is expected to be able to:

1. Plan integrative projects of software development
2. Cooperate, organize and design strategies for the resolution of technological problems
3. Analyze and design tools, approaches, and methodologies with the potential for solving complex problems
4. Recognize and use different areas of knowledge to incorporate and interpret social problems in the technological area

### Prerequisites

Not applicable

### Course contents

Software development should be contextualized in the problem and depend on a set of tools and APIs, contextualized in a specific programming language. Each problem should be well perceived, followed by options about the development methodologies. Students should be able to implement methodologies that allow them to perceive the problem in light of the state-of-the-art.

### Course contents (extended version)

1. Context of software development problems
  - Research in software development
  - Methodologies for the development of the state-of-the-art
  - Reflection and applied research
  - Definition of intervention areas
  - Data collection and analysis in context
2. Design of development project(s)
  - Problem definition
  - Design of the intervention project(s) and the approach to follow in practice
  - Definition of the research requirements
3. Specialization multi-language
  - Study and adoption of programming languages
  - Study and adoption of APIs
4. From the concept to the prototype

### Recommended reading

1. Complete Guide to Test Automation: Techniques, Practices, and Patterns for Building and Maintaining Effective Software Projects, Arnon Axelrod, Apress, ASIN: B07FKGVQP6, 2018.
2. Release It!: Design and Deploy Production-Ready Software, Michael T. Nygard, Pragmatic Bookshelf, ASIN: B079YWMY2V, 2018
3. Continuous Delivery with Docker and Jenkins: Create secure applications by building complete CI/CD pipelines, 2nd Edition, Rafal Leszko, Packt Publishing, ASIN: B07SJKHJZ7, 2019
4. Curso "Engenharia de Software: Design de Software e Gerenciamento de Projetos - The Hong Kong University of Science and Technology", plataforma Coursera, 2023
5. Curso "Engenharia de Software: Implementação e Teste - The Hong Kong University of Science and Technology", plataforma Coursera, 2023

### Teaching and learning methods

Project work, group work and argumentative discussion. Individual reflexion about subjects and questions resulting from the teaching-learning experiences. It will be recommended to the students the completion of Coursera courses, according to the bibliography.

### Assessment methods

- Project and written report. 0,75 each Coursera. - (Regular, Student Worker) (Final, Supplementary, Special)

### Language of instruction

English

### Electronic validation

Rui Pedro Sanches de Castro Lopes	Tiago Miguel Ferreira Guimaraes Pedrosa	José Eduardo Moreira Fernandes	José Carlos Rufino Amaro
11-10-2023	25-10-2023	30-10-2023	04-11-2023