# Course guide
# 804228 - ASO - Architecture and Operative Systems

**Last modified:** 15/09/2024

| | |
|---|---|
| **Unit in charge:** | Image Processing and Multimedia Technology Centre |
| **Teaching unit:** | 804 - CITM - Image Processing and Multimedia Technology Centre. |

| | |
|---|---|
| **Degree:** | BACHELOR'S DEGREE IN VIDEO GAME DESIGN AND DEVELOPMENT (Syllabus 2014). (Compulsory subject). |

**Academic year:** 2024  **ECTS Credits:** 6.0  **Languages:** Catalan, English

## LECTURER

| | |
|---|---|
| **Coordinating lecturer:** | Costa Prats, Juan José |
| **Others:** | García Almiñana, Jordi |

## DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

**Generical:**
1. Analyse and evaluate the structure and architecture of computers and their basic constituent components.
2. Interpret the basics of the use and programming of computers, operating systems, databases and in general, computer programs with applications for engineering.

**Transversal:**
3. EFFICIENT ORAL AND WRITTEN COMMUNICATION. Communicating verbally and in writing about learning outcomes, thought-building and decision-making. Taking part in debates about issues related to the own field of specialization.
4. TEAMWORK: Being able to work in an interdisciplinary team, whether as a member or as a leader, with the aim of contributing to projects pragmatically and responsibly and making commitments in view of the resources that are available.

5. EFFECTIVE USE OF INFORMATION RESOURCES: Managing the acquisition, structuring, analysis and display of data and information in the chosen area of specialisation and critically assessing the results obtained.

## TEACHING METHODOLOGY

The course will combine presentations of fundamental concepts by the teacher with participatory sessions, where students will prepare, present and defend works on specific concepts of the subject. Collaborative learning techniques are used to motivate students to carry out the activities. Additionally, practical work will be encouraged where the concepts studied are put into practice.
Therefore, the teaching methodologies will be used:
. Expository method / master lesson.
. Participatory class.
. Learning based on problems and exposures and defenses of practices or works.
. Practical programming sessions.

## LEARNING OBJECTIVES OF THE SUBJECT

· Demonstrating knowledge and understanding of the internal operation of a computer, and of each of its basic components: processor, memory, interconnection bus, and input and output devices.

· Demonstrating knowledge and understanding of the type of information that a computer stores and processes, as well as the machine language that controls it. Know the relationship between programming languages â□□â□□and machine language.

· Demonstrating knowledge and understanding of the mechanisms of access to input and output devices, and of the mechanisms of interruption and data transmission.

· Knowing the particularity of the different specific devices in the field of video games and mobile applications. Knowing the relationship and differences with generic devices.

· Demonstrating knowledge and understanding of the characteristics, functionalities and internal structure of operating systems, which allows an adequate and efficient use of the computer.

· Demonstrating knowledge and understanding of the overview of computer systems, and know the different interfaces and components that interact with the operating system (hardware, system libraries, language libraries, user programs).

· For each basic functionality of the operating system (process management, memory management and input / output management), demonstrating knowledge and understanding of the relationship between program, computer and operating system.

· Knowing the particularity of the management of the operating system on specific systems or devices in the field of video games and mobile applications.

## STUDY LOAD

| Type | Hours | Percentage |
|---|---|---|
| Self study | 90,0 | 60.00 |
| Guided activities | 20,0 | 13.33 |
| Hours medium group | 16,0 | 10.67 |
| Hours large group | 24,0 | 16.00 |

**Total learning time:** 150 h

## CONTENTS

**Theme 1: Introduction. Evolution of computers and operating systems**

**Description:**
Introduction to the course

**Full-or-part-time:** 5h
Theory classes: 2h
Self study : 3h

## Part I: Computer's architecture

**Description:**
· Theme 2: Computer's architecture
· Theme 3: The assembly language
· Theme 4: Current computers

**Full-or-part-time:** 85h
Theory classes: 34h
Self study : 51h

## Part II: Operating systems

**Description:**
· Topic 5: Basic structure of the operating system
· Topic 6: Memory management
· Topic 7: Process management
· Topic 8: Input / output management
· Topic 9: Specific aspects of the OS for video games

**Full-or-part-time:** 60h
Theory classes: 24h
Self study : 36h

## ACTIVITIES

### Hands On Lab (HOL)

**Description:**
Practical part of the course where the theoretical concepts explained are put into practice: assembler, process management, report and input / output.

**Full-or-part-time:** 15h
Laboratory classes: 15h

### Research Work

**Description:**
Research work to apply the theoretical concepts of the course to a specific use case in the world of video games

**Full-or-part-time:** 8h
Theory classes: 8h

### Test

**Description:**
Exam to demonstrate the achievement of theoretical concepts

**Full-or-part-time:** 2h
Theory classes: 2h

## GRADING SYSTEM

The course is evaluated by the evaluation of these components:
Exam part1 (EX1)
. Research work (TR)
. Exam part2 (EX2)
. Practical work (TP)
Both exams are carried out in the middle and end of the course respectively and are of a theoretical nature, where the student must demonstrate knowledge and understanding of the concepts worked during the course.
The research work consists of the search for documentation during the course on specific concepts of the subject, and includes a presentation and debate. The practical work consists of solving a series of programming problems that show the concepts studied. These works will be carried out in groups. The final grade is calculated according to the relationship:

F = 22.5% EX1 + 22.5% TR + 22.5% EX2 + 22.5% TP + 10% AA

Where AA corresponds to participation and attitude of learning.

Students who fail in the continuous assessment can be re-evaluated (as long as the grade is different from NP). The grade obtained in the reevaluation replaces, if higher, the set of those obtained in the midterm and final exams. The final grade for the course, calculated from the reevaluation exam, can not exceed 5.

Irregular actions that may lead to a significant variation of the grade of one or more students constitute a fraudulent performance of an evaluation act. This action entails the descriptive grade of failure and a numerical grade of 0 for the ordinary global evaluation of the course, without the right to re-evaluation.

If the lecturers have indications of the use of AI tools not allowed in the evaluation tests, they may summon the students concerned to an oral test or a meeting to verify the authorship.

## BIBLIOGRAPHY

**Basic:**
- Randal E. Bryant, David R. O'Hallaron. Computer Systems: A programmer's perspective. Pearson Education,
- Kip R. Irvine. Lenguaje ensamblador para computadoras basadas en Intel®. Pearson Educación , 2008.