Course guide
804268 - DVDM - Mobile Game Development

Last modified: 06/02/2026

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). (Optional subject).

Academic year: 2025 **ECTS Credits:** 6.0 **Languages:** English

LECTURER

Coordinating lecturer: Eric Batlle

Others: Eric Batlle

PRIOR SKILLS

High-level programming knowledge. Basic knowledge of C#.

TEACHING METHODOLOGY

Active expository teaching (participatory lecture): Structured transmission of content by the teacher, combined with active participation by students (open questions, exercises).

LEARNING OBJECTIVES OF THE SUBJECT

- Identify advanced optimisation techniques in the field of video game development for different platforms and devices.

- Use advanced optimisation techniques in the development of video games and real-time graphics applications for different platforms and devices.

STUDY LOAD

Type	Hours	Percentage
Hours medium group	18,0	12.00
Guided activities	12,0	8.00
Self study	90,0	60.00
Hours large group	30,0	20.00

Total learning time: 150 h

CONTENTS

Technical Foundations

Description:
The "Technical Fundamentals" block introduces the technical foundations necessary for developing video games in Unity, working with C# applied to the context of video games through the use of data structures, reactivity, polling, and a practical understanding of asynchrony and multithreading.
In addition, common errors in Unity projects that cause scalability and performance issues will be analysed, and more advanced techniques such as testing as a design tool will be introduced.

Full-or-part-time: 30h
Theory classes: 6h
Practical classes: 6h
Self study : 18h

Basic Software Architecture and Design

Description:
The "Basic Software Architecture and Design" block focuses on improving code design quality by applying SOLID principles and fundamental design patterns, understood as tools for solving common problems in video games, and introducing data and rule modelling as the basis for more maintainable and extensible design.

Full-or-part-time: 30h
Theory classes: 6h
Practical classes: 6h
Self study : 18h

Advanced Software Architecture and Design

Description:
The "Advanced Software Architecture and Design" block delves into software organisation at the architectural level. Architectural patterns such as MVC and MVP will be studied, as well as the principles of Clean Architecture, analysing the problems they solve, their advantages and their trade-offs in video game development, with the aim of protecting the game domain from changes in presentation, engine or infrastructure.

Full-or-part-time: 30h
Theory classes: 6h
Practical classes: 6h
Self study : 18h

Performance & Mobile

Description:
The "Performance & Mobile" block focuses on performance analysis and efficient resource management in Unity. Students will learn how to use profiling tools to identify bottlenecks. They will also work on mobile UI design, adaptation to different resolutions and screen ratios, and memory and content management. The block emphasises technical decision-making that directly impacts game performance, stability, and scalability, encouraging cost estimation and technical communication in development environments.

Full-or-part-time: 60h
Theory classes: 12h
Practical classes: 12h
Self study : 36h

GRADING SYSTEM

Midterm exam (Blocks 1-2): 30%

Assignments: 60%
- Block 1 Exercise Assignment - 5%
- Block 1 Final Assignment - 10%
- Block 2 Exercise Assignment - 5%
- Block 2 Final Assignment - 10%
- Block 3 Final Assignment - 15%
- Block 4 Final Assignment - 15%

Participation and attitude: 10%

Reassessment: Students who have not passed the subject by continuous assessment have the option to be submitted to the reassessment. This will be an exam of 2 hours and the qualification will substitute the one of the exam. To be eligible, it is required to have presented the process of continuous assessment.

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:
- Martin, Robert. Clean Code. Anaya Multimedia, 2012.

RESOURCES

Other resources:
Audiovisual Material:
- Unity Official. https://unity.com/es- Unity Learn. https://learn.unity.com/ />
Links:
- Manual. https://docs.unity3d.com/Manual/ />- Unity Tutorials. https://www.youtube.com/@unity />- Asset Store. https://assetstore.unity.com/- C# Learn. https://learn.microsoft.com/en-us/dotnet/csharp/