

## 804246 - MVJ - Game Engines

Coordinating unit: 804 - CITM - Image Processing and Multimedia Technology Centre  
 Teaching unit: 804 - CITM - Image Processing and Multimedia Technology Centre  
 Academic year: 2019  
 Degree: BACHELOR'S DEGREE IN VIDEO GAME DESIGN AND DEVELOPMENT (Syllabus 2014). (Teaching unit Compulsory)  
 BACHELOR'S DEGREE IN VIDEO GAME DESIGN AND DEVELOPMENT (Syllabus 2014). (Teaching unit Compulsory)  
 ECTS credits: 6 Teaching languages: Catalan, Spanish, English

### Teaching staff

Coordinator: Pillosu González, Ricard

### Prior skills

Coding in C++. Previous knowledge and experience coding 2D games.

### Teaching methodology

During each class, the lecturer will first show the students the theory behind the problem that needs solving. Together with the students, the lecturer will explore the different solutions that exist in the present that solve and simplify the complexities of real time applications like videogames.

### Learning objectives of the subject

- Understand how to organize the rendering pipeline and proper loading of a 3D scene.
- Knowledge in how to integrate 3D animation systems.
- Internal structure for entities and their components.
- Audio for 3D environments.
- Most common graphic techniques.

### Study load

Total learning time: 150h	Hours large group:	18h	12.00%
	Hours medium group:	30h	20.00%
	Hours small group:	0h	0.00%
	Guided activities:	12h	8.00%
	Self study:	90h	60.00%

## 804246 - MVJ - Game Engines

### Content

<p>OpenGL basics</p>	<p>Learning time: 15h Theory classes: 6h Self study : 9h</p>
<p>Description: Initialization Rendering in direct mode Vertex Buffers</p>	
<p>Loading 3D models</p>	<p>Learning time: 10h Theory classes: 4h Self study : 6h</p>
<p>Description: Loading of model information: geometry and materials Rendering of single models</p>	
<p>Camera and scene loading</p>	<p>Learning time: 15h Theory classes: 6h Self study : 9h</p>
<p>Description: Free roaming camera, FPS style and single model Loading scene information Execution in threads</p>	
<p>Basic rendering optimizations</p>	<p>Learning time: 10h Theory classes: 4h Self study : 6h</p>
<p>Description: Frustum culling Level of details Octree</p>	

## 804246 - MVJ - Game Engines

<p>Animation systems</p>	<p>Learning time: 25h Theory classes: 10h Self study : 15h</p>
<p>Description: Implementing a Transformation Tree Structure of an animation system Loading of animations Playing and blending of animations</p>	
<p>Component structure and player control</p>	<p>Learning time: 30h Theory classes: 12h Self study : 18h</p>
<p>Description: Component system for entities Messaging and event system Physics and player control</p>	
<p>3D Audio</p>	<p>Learning time: 10h Theory classes: 4h Self study : 6h</p>
<p>Description: Loading and playing music Playing 3D effects</p>	
<p>Graphics effects</p>	<p>Learning time: 20h Theory classes: 8h Self study : 12h</p>
<p>Description: Particle systems Postprocess effects Illumination models</p>	

## 804246 - MVJ - Game Engines

### Qualification system

Final exam with a weight of 40% with all subject knowledge will be put to test.

First assignment about GameObjects hierarchy, components and Bounding boxes with frustum culling with a weight of 15%.

Second assignment about Quadrees, Time Management, Mouse Picking, optimized formats and serialization with a weight of 15%.

Third assignment about a single high level system to choose from: animation, particles, audio, scripting, physics, shaders or UI with a weight of 20%.

The final exam can be reevaluated for its weight of 40%

Attitude and class participation will weight 10%

### Bibliography

Basic:

Gregory, Jason. Game engine architecture. 2nd ed. Boca Raton: CRC Press, Taylor and Francis Group, cop. 2014. ISBN 9781466560017.

Nystrom, Robert. Game programming patterns. [United States?]: Genever Benning, 2014. ISBN 9780990582908.