

BASIC DETAILS:

Subject:	ANIMACIÓN Y SIMULACIÓN		
Id.:	31377		
Programme:	GRADUADO EN DISEÑO Y DESARROLLO DE VIDEOJUEGOS. 2013 (BOE 28/03/2014)		
Module:	PROGRAMACIÓN DE VIDEOJUEGOS		
Subject type:	OBLIGATORIA		
Year:	3	Teaching period:	Segundo Cuatrimestre
Credits:	6	Total hours:	150
Classroom activities:	65	Individual study:	85
Main teaching language:	Inglés	Secondary teaching language:	Castellano
Lecturer:	BLASCO LATORRE, DANIEL (T)	Email:	dblasco@usj.es

PRESENTATION:

The subject covers an introduction to simulation and animation techniques from a computer science perspective, applied to real time interactive products, like videogames and educational / training environments.

The goals include learning fundamentals on mathematical / physical methods in real time application programming, as well as understanding the implications of the derived behaviours in gameplay mechanics and user experience.

PROFESSIONAL COMPETENCES ACQUIRED IN THE SUBJECT:

General programme competences	G07	Ability to handle different complex knowledge models through a process of abstraction and its application to approach and solve problems.
Specific programme competences	E01	Ability to solve mathematical problems inherent to engineering. Ability to apply knowledge about: algebra; geometry; differential and integral calculus; optimisation and numerical methods
	E02	Ability to understand and master the concepts of the general laws of classical mechanics, fields, waves and electromagnetism and their application for solving video game development problems.
	E21	Ability to execute the art of video games, create characters and settings.
	E26	Ability to perform the design and creation of animated characters and their application in the development of video games.
	E29	Ability to understand and apply the techniques of visualisation, animation, simulation and interaction on models
	E37	Ability to design and create sounds and sound environments and their application in game development
Learning outcomes	R01	Create an animation character by skeleton in a video game.
	R02	Capture a movement and use it as the basis for the animation of a character in a video game.
	R03	Explain the fundamental physical principles to simulate in a video game.
	R04	Apply the basic techniques of physical simulation.
	R05	Use libraries of animation and physical simulation in video games.
	R06	Use animation software for sound.

PRE-REQUISITES:

This subject doesn't have pre-requisites. The students should have object oriented programming knowledge, experience with Unity Engine and C#, as well as basic physics, geometry, calculus and algebra knowledge.

SUBJECT PROGRAMME:

Subject contents:

1 - Introduction
1.1 - Methodology Initial Overview
1.2 - Mathematics and Physics Review
1.3 - Basic Behaviour Use
2 - Rigid-Body Dynamics
2.1 - Motion Basics
2.2 - Force Management
2.3 - Object Interaction
3 - Applied Mechanics
3.1 - Vehicle Modeling Notions
3.2 - Projectiles
3.3 - Character Animation and Motion Capture
4 - Beyond Basics Appendix
4.1 - Non-Rigid Bodies
4.2 - Inverse Kinematics
4.3 - Audio Simulation

Subject planning could be modified due unforeseen circumstances (group performance, availability of resources, changes to academic calendar etc.) and should not, therefore, be considered to be definitive.

TEACHING AND LEARNING METHODOLOGIES AND ACTIVITIES:

Teaching and learning methodologies and activities applied:

Theory/ Practice Sessions:

During these sessions, the contents featured in the subject will be exposed using resources like whiteboards, slideshows, etc., to show examples and illustrate properly the different sections. Additionally, active involvement will be encouraged through theoretical or real life case discussion. These sessions will be supported by different exercises.

Individual/ Team Exercises:

A part of the overall score will depend on individual exercises dealing with the different sections studied. These exercises will involve programming or tool usage and they are meant not to be independent, but interrelated, as new content is presented/ added in the course. Each exercise will consist of a set of instructions and certain results to be delivered before a specific date. Apart from individual work, a group activity will be developed forming teams and under similar conditions.

Tests/ Exams:

A written test will act as a theory/ practice assessment method, covering the content in the subject. The main purpose of this test is evaluating the knowledge acquired and underlying the processes and cases studied and explored in both lectures and exercises.

Tutorials:

The students will take part, on demand, in tutorials to be conducted on Thursdays at 11:00 AM, but schedules may vary according to particular necessities or circumstances. The main goal pursued is to clear up doubts, and help students strengthen the knowledge and skills to be acquired. Just like with other subjects, the PDU is a useful communication tool to ask for/ share information on the course.

Student work load:

Teaching mode	Teaching methods	Estimated hours

Classroom activities	Master classes	15
	Practical work, exercises, problem-solving etc.	12
	Workshops	12
	Laboratory practice	16
	Assessment activities	4
	Extra-curricular activities (visits, conferences, etc.)	6
Individual study	Tutorials	4
	Individual study	22
	Individual coursework preparation	22
	Project work	23
	Research work	4
	Compulsory reading	5
	Recommended reading	5
Total hours:		150

ASSESSMENT SCHEME:

Calculation of final mark:

Written tests:	25 %
Individual coursework:	25 %
Group coursework:	30 %
Final exam:	20 %
TOTAL	100 %

*Las observaciones específicas sobre el sistema de evaluación serán comunicadas por escrito a los alumnos al inicio de la materia.

BIBLIOGRAPHY AND DOCUMENTATION:

Basic bibliography:

BOURG, David M., BYWALEC, Bryan. Physics for Game Developers. Second Edition. Sebastopol, CA: O'Reilly Media, Inc., 2013.

Recommended bibliography:

GREGORY, Jason. Game Engine Architecture. Second Edition. Boca Raton, FL: A K Peters / CRC Press, 2014.

Recommended websites:

Unity Technologies <https://unity3d.com/>