Course guide
804232 - FIS2VJ - Physics II

Last modified: 14/07/2022

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: 2022 **ECTS Credits:** 6.0 **Languages:** Catalan, English

LECTURER

Coordinating lecturer: De La Torre Sangrà, David

Others: De La Torre Sangrà, David

PRIOR SKILLS

Knowledge about Physics and coding in C++

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Generical:
CGFC1VJ. Design, develop, select and evaluate applications and computer systems from 0 for video games, ensuring their reliability, security and quality, in accordance with ethical principles and current legislation and regulations.
CGFB2VJ. Interpret and master the basic concepts of the general laws of mechanics, thermodynamics, fields and waves and electromagnetism; and their application for solving engineering problems.
CGFB1VJ. Solve mathematical problems that may arise in engineering. Apply knowledge of linear algebra; geometry; integral and differential calculus; numerical methods; statistics.

Transversal:
05 TEQ N1. TEAMWORK - Level 1. Working in a team and making positive contributions once the aims and group and individual responsibilities have been defined. Reaching joint decisions on the strategy to be followed.

TEACHING METHODOLOGY

The weekly theory lectures consist in 2 hours (1 session of 2 h).
- Exposition about physics concepts, examples of application to videogames, live gameplays.
- Exercises of development and analysis of physics engines.
- Development of the projects.

The weekly practice laboratory consist in 2 hours (1 session of 2 h).
- Training exercises targeting the course's APIs (Box2D, Bullet).
- Development of the projects.

Activities schedule may change, depending on the difficulty of the exercises and the corresponding contents.
The supporting material to be used will be available at the virtual campus site.

LEARNING OBJECTIVES OF THE SUBJECT

- Knowledge on how to create games based on physics simulations in both 2D and 3D.
- To be able to apply physical models to video games and simulations in both 2D and 3D.
- To understand the structure of the Box2D and Bullet libraries.

STUDY LOAD

Type	Hours	Percentage
Hours large group	34,0	22.67
Self study	90,0	60.00
Guided activities	10,0	6.67
Hours medium group	16,0	10.67

Total learning time: 150 h

CONTENTS

Physics Review

Description:
Review of the physical concepts given in Physics 1 and basics calculus tools:
- Overview on vector and differential calculus.
- Coordinate systems. Relative position of physical bodies in the 3D space and collisions.
- Kinematics 1D, 2D and 3D.
- Dynamics: motion under forces, non-frictional and frictional systems.
- Momentum balance and collisions 1D and 2D (purely elastic, inelastic, breakage).

Full-or-part-time: 8h
Theory classes: 2h
Practical classes: 2h
Self study : 4h

Rigid Body Dynamics

Description:
Description of kinematics and dynamics of the rigid body:
- Review on matrix calculus.
- Momentum conservation. Angular momentum.
- Center of mass. Inertia.
- Rotational movement 2D and 3D: Pure translation and pure rotation.
- Rotational dynamics 2D and 3D: forces and torques.
- Rigid body transformation: 2D and 3D displacement and rotation, deformation.

Full-or-part-time: 8h
Theory classes: 2h
Practical classes: 2h
Self study : 4h

Integrator and Framerate

Description:
Numerical integration methods:
- Implicit Euler.
- Symplectic Euler.
- Velocity-Verlet & Störmer-Verlet.
- High order Runge-Kutta.

Framerate control methods:
- Types: fixed, variable, hybrid.
- Sub-stepping and advanced methods.
- Multiplayer syncronicity.

Related activities:
Project Theory

Full-or-part-time: 16h
Theory classes: 4h
Practical classes: 4h
Self study : 8h

Collisions

Description:
- Elastic vs dampening.
- Collision solving methods.
- Raycasting.

Related activities:
Project Theory

Full-or-part-time: 8h
Theory classes: 2h
Practical classes: 2h
Self study : 4h

Physics Forces

Description:
Main physics forces applied to videogames:
- Gravity: constant, scaled, lineal, universal law of gravitation.
- Aerodynamics: lift, drag.
- Hydrodynamics: flotación, drag, sustentación.
- Oscillations: harmonic motion, springs.

Related activities:
Project Theory

Full-or-part-time: 32h
Theory classes: 8h
Practical classes: 8h
Self study : 16h

Complex Systems

Description:
- Ropes/cables.
- Cloth/textiles.
- Soft bodies.
- Fluids.
- Ragdolls.

Full-or-part-time: 6h
Theory classes: 2h
Self study : 4h

Virtual Reality

Description:
- Physics in VR/AR videogames.
- User-game interface.
- Motion sickness.

Full-or-part-time: 6h
Theory classes: 2h
Self study : 4h

Box2D integration

Description:
- Analyzing the Box2D API.
- Integration plan.
- Binding creation in C++.
- Collision detection.
- Physics simulation.

Related activities:
Project Box2D

Full-or-part-time: 15h
Theory classes: 6h
Self study : 9h

Integration of Bullet

Description:
- Analyzing the API of Bullet 3D.
- Integration plan.
- Binding creation in C++.
- Collision detection.
- Physics simulation.

Related activities:
Project Bullet

Full-or-part-time: 15h
Theory classes: 6h
Self study : 9h

ACTIVITIES

Theory Project (2D Physics)

Description:
The objective is to create a physics engine from scratch and apply it to a functional videogame.
- The students shall use all the concepts explained in theory class and use them to build their own physics engine.
- Said physics engine shall then be used in the context of a simple (but functional) videogame. Relative freedom is granted to choose the game's theme and content.
- The main objective of the project is to be able to analyse and see how a physics engine works, so being able to configure and "play" with the engine's parameters will be a must.

Activities:
- Defining goals and limitations.
- Creating a environment for simulations.
- Coding the interactive elements.
- Victory conditions.

Specific objectives:
- Understand physics principles: Newton's laws, forces vs impulses, inertia, linear vs angular dynamics, gravity, aerodynamics, hydrodynamics, harmonic motion, collisions.
- Understand how to apply the above to videogames: accuracy vs performance, simplifications.

Material:
- The students will have a barebones code available at the start of the project. That code already deals with pre-processing, the main game loop and the graphics engine. The students shall only implement the physics engine and whichever game logic is required.
- The theory classes will provide with all the required concepts and algorithms to build a functional physics engine.
- Bibliography (books, video resources, etc.) will also be available as a support material.

Delivery:
The delivery of this task shall include:
- A working executable release of the game.
- The source code of the game.
- Documentation (Readme, Manual, Guide, etc.)

Related competencies :
CGFB2VJ. Interpret and master the basic concepts of the general laws of mechanics, thermodynamics, fields and waves and electromagnetism; and their application for solving engineering problems.
CGFC1VJ. Design, develop, select and evaluate applications and computer systems from 0 for video games, ensuring their reliability, security and quality, in accordance with ethical principles and current legislation and regulations.
CGFB1VJ. Solve mathematical problems that may arise in engineering. Apply knowledge of linear algebra; geometry; integral and differential calculus; numerical methods; statistics.
05 TEQ N1. TEAMWORK - Level 1. Working in a team and making positive contributions once the aims and group and individual responsibilities have been defined. Reaching joint decisions on the strategy to be followed.

Full-or-part-time: 11h
Guided activities: 3h
Self study: 8h

Project Box2D (2D Physics)

Description:
The objective is to learn to use the Box2D physics library.
- The students shall use Box2D as a physics engine to create a Pinball game.

Activities:
- Defining goals and limitations.
- Creating a environment for simulations.
- Coding the interactive elements.
- Victory conditions.

Specific objectives:
- Learn how to work with Box2D physics engine.
- Understand physics principles: Newton's laws, forces vs impulses, inertia, linear vs angular dynamics, gravity, aerodynamics, hydrodynamics, harmonic motion, collisions.
- Understand how to apply the above to videogames: accuracy vs performance, simplifications.

Material:
- The students will have a barebones code available at the start of the project. That code already deals with pre-processing, the main game loop and the graphics engine. The students shall only implement the physics engine and whichever game logic is required.
- The practice (and to a lesser extent, theory) classes will provide with all the required concepts and algorithms to configure and work with the Box2D physics engine.
- The main documentation of Box2D will be the primary source of technical references.
- Bibliography (books, video resources, etc.) will also be available as a support material.

Delivery:
The delivery of this task shall include:
- A working executable release of the game.
- The source code of the game.
- Documentation (Readme, Manual, Guide, etc.)

Related competencies :
CGFC1VJ. Design, develop, select and evaluate applications and computer systems from 0 for video games, ensuring their reliability, security and quality, in accordance with ethical principles and current legislation and regulations.
CGFB2VJ. Interpret and master the basic concepts of the general laws of mechanics, thermodynamics, fields and waves and electromagnetism; and their application for solving engineering problems.
CGFB1VJ. Solve mathematical problems that may arise in engineering. Apply knowledge of linear algebra; geometry; integral and differential calculus; numerical methods; statistics.
05 TEQ N1. TEAMWORK - Level 1. Working in a team and making positive contributions once the aims and group and individual responsibilities have been defined. Reaching joint decisions on the strategy to be followed.

Full-or-part-time: 11h
Guided activities: 3h
Self study: 8h

Project Bullet (3D Physics)

Description:
The objective is to learn to use the Bullet physics library.
- The students shall use Bullet as a physics engine to create a Racing car game (or alike).

Activities:
- Definition of the goals and limitation of racing games.
- Creating the environment for the simulations.
- Car creation.
- Victory conditions.

Specific objectives:
- Learn how to work with Bullet physics engine.
- Understand physics principles: Newton's laws, forces vs impulses, inertia, linear vs angular dynamics, gravity, aerodynamics, hydrodynamics, harmonic motion, collisions.
- Understand how to apply the above to videogames: accuracy vs performance, simplifications.

Material:
- The students will have a barebones code available at the start of the project. That code already deals with pre-processing, the main game loop and the graphics engine. The students shall only implement the physics engine and whichever game logic is required.
- The practice (and to a lesser extent, theory) classes will provide with all the required concepts and algorithms to configure and work with the Bullet physics engine.
- The main documentation of Bullet will be the primary source of technical references.
- Bibliography (books, video resources, etc.) will also be available as a support material.

Delivery:
The delivery of this task shall include:
- A working executable release of the game.
- The source code of the game.
- Documentation (Readme, Manual, Guide, etc.)

Related competencies :
CGFB1VJ. Solve mathematical problems that may arise in engineering. Apply knowledge of linear algebra; geometry; integral and differential calculus; numerical methods; statistics.
CGFC1VJ. Design, develop, select and evaluate applications and computer systems from 0 for video games, ensuring their reliability, security and quality, in accordance with ethical principles and current legislation and regulations.
CGFB2VJ. Interpret and master the basic concepts of the general laws of mechanics, thermodynamics, fields and waves and electromagnetism; and their application for solving engineering problems.
05 TEQ N1. TEAMWORK - Level 1. Working in a team and making positive contributions once the aims and group and individual responsibilities have been defined. Reaching joint decisions on the strategy to be followed.

Full-or-part-time: 16h
Guided activities: 4h
Self study: 12h

GRADING SYSTEM

The qualification of the subject will be obtained following a system of continuous evaluation. The weight of each part is as follows:
- Project Theory: 25%
- Project Box2D: 15%
- Project Bullet: 30%
- Final Exam: 20%
- Participation and attitude 10%

The approved one is obtained when obtaining a note of 5 in the final grade weighted according to the previous criterion. If you do not present an exam or practical exercise, you will get a score of 0.
If the subject is not passed, there is the possibility of presenting a re-evaluation exam, during which only the theoretical part will be reassessed. In case of passing the course, the maximum final mark will be 5.

EXAMINATION RULES.

In-class exercises:
During the lectures, students will develop exercises to be discussed and solved in the same lecture. These exercises act as training to do the projects.

Projects:
The projects will be done in groups and they will be delivered before the established deadline. The delivery includes de developed code (C++, Matlab, Python, etc.), a functional release of the game, and a technical report if necessary.

BIBLIOGRAPHY

Basic:
- Bourg, David M. Physics for game developers . 2nd ed. Beijing: O'Reilly, 2013. ISBN 978-1449392512.
- Palmer, Grant. Physics For Game Programmers. 1st ed. Apress, 2005. ISBN 978-1590594728.
- Dickinson, Chris. Learning Game Physics with Bullet Physics and OpenGL. 1st ed. Packt Publishing Ltd., 2013. ISBN 978-1-78328-187-9.
- Eberly, David H. Game Physics. 2nd ed. Morgan Kaufmann, 2010. ISBN 978-0123749031.
- Parberry, I. Introduction to game physics with Box2D. 1st ed. Boca Raton: CRC Press, 2013. ISBN 9781466565760.
- Millington, Ian. Game Physics Engine Development. 2nd ed. CRC Press, 2017. ISBN 1138403121.

Complementary:
- Sanglard, Fabien. Game Engine Black Book: Doom. Version 1.1. Independently published, 2019. ISBN 978-1099819773.
- van den Bergen, Gino. Game Physics Pearls. 1st ed. CRC Press, 2010. ISBN 978-1-56881-474-2.
- Szauer, Gabor. Game Physics Cookbook. 1st ed. Packt Publishing, 2017. ISBN 978-1787123663.
- Emperore, K; Sherry, D. Unreal Engine Physics Essentials. 1st ed. Packt Publishing, 2015. ISBN 978-1-78439-490-5.
- Sanglard, Fabien. Game Engine Black Book: Wolfenstein 3D. Version 2.1. Independently published, 2019. ISBN 978-1070515847.
- Feronato, Emanuele. Box2D for Flash Games. 1st ed. Packt Publishing, 2012. ISBN 978-1849519625.
- Harbour, Jonathan S. Multi-Threaded Game Engine Design. 1st ed. Course Technology PTR, 2010. ISBN 1435454170.
- Ericson, Christer. Real-Time Collision Detection. 1st ed. Morgan Kaufmann, 2005. ISBN 978-0080474144.

RESOURCES

Hyperlink:
- GDC (Game Developer's Conference). https://www.youtube.com/c/Gdconf- SIGGRAPH (Association for Computing Machinery's (ACM) Special Interest Group on Computer Graphics and Interactive Techniques). https://www.youtube.com/user/ACMSIGGRAPH