



Course guides

804254 - XJO - Networks and Online Games

Last modified: 07/04/2021

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: 2021 **ECTS Credits:** 6.0 **Languages:** Catalan, English

LECTURER

Coordinating lecturer: Abadal Cavallé, Sergi

Others: Díaz García, Jesús

PRIOR SKILLS

To be able to programme and develop computer applications
To know and be the main Internet tools and services at user level

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Specific:

CEVJ 5. Use programming languages, algorithmic patterns, data structures, visual programming tools, game engines and libraries for the development and prototyping of video games, in any genre and for any platform and mobile device.

CEVJ 13. Undertake and manage video game design and development projects, including planning, direction, execution and evaluation.

Generical:

CGFB5VJ. Interpret the structure, operation and interconnection of computer systems, and the basics of their programming.

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.

07 AAT. SELF-DIRECTED LEARNING. Detecting gaps in one's knowledge and overcoming them through critical self-appraisal. Choosing the best path for broadening one's knowledge.

04 COE N3. EFFICIENT ORAL AND WRITTEN COMMUNICATION - Level 3. Communicating clearly and efficiently in oral and written presentations. Adapting to audiences and communication aims by using suitable strategies and means.

TEACHING METHODOLOGY

The teaching methodology is divided in four parts:

- Sessions for the contents exposition at classroom
- Practical working sessions at classroom (training for the use of the tools, presentations and use case debates)
- Practical development of applications/games in C++ and Unity where the concepts learnt during the course will be applied
- Autonomous work to study and carry out exercises and activities

In the content presentation sessions, teachers will introduce the theoretical bases of the subject, concepts, methods and results, illustrating them with convenient examples to facilitate their understanding.

In the practical work sessions in the classroom, the teacher will guide the student in the application of the theoretical concepts for problem solving, based at all times on critical reasoning. Exercises will be proposed that the student solves in the classroom and outside the classroom, in order to favor the contact and use of the basic tools necessary for the resolution of problems.

The student, autonomously, must work on the material provided by the teacher and the result of the work-problem sessions in order to assimilate and fix the concepts. Teachers will provide a study and activity monitoring plan (CAMPUS).

The methodology of the subject is a combination between the face-to-face class and the realization of different complementary activities (resolution of cases, problems, etc.), either individually or in small groups, both inside and outside the classroom. In the contact hours the student receives the basic information to work on the contents of the teaching plan that guarantee the achievement of the established objectives.

In the Virtual Campus all the information relative to the subject is published (rules of evaluation, dates of delivery of the activities, etc.). Attendance at the sessions is important for the learning process, which is based on both individual or team work and classroom participation.

LEARNING OBJECTIVES OF THE SUBJECT

- To show understanding and application capacity in the on-line game development, about foundations of telematics networks, their capacities and limitations
- To be able to identify the problems related to online games caused by the network limitations and propose solutions
- To show knowledge and be able to use network game engines, for the development of online games
- To show knowledge of game servers and to know how to use them for the development and implementation of online games
- To show knowledge of protocols at the application layer, as well as of protocols in lower layers to a lesser extent

STUDY LOAD

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

Total learning time: 150 h



CONTENTS

1. Introduction

Description:

- 1.1 Presentation of the subject and evaluation criteria
- 1.2 A brief history of networks and on-line games
- 1.3 Architectures and metrics
- 1.4 On-line games seen as distributed systems

Specific objectives:

To obtain basic knowledge about the history of the Internet and on-line games, about the basic types of network architectures and the metrics used to measure their performance, as well as to understand an on-line game as a distributed system.

Related activities:

Programming with threads

Related competencies :

CGFB5VJ. Interpret the structure, operation and interconnection of computer systems, and the basics of their programming.

Full-or-part-time: 10h

Theory classes: 2h

Guided activities: 2h

Self study : 6h

2. The OSI Protocol Stack

Description:

- 2.1 Introduction. The physical and link layers
- 2.2 The network and transport layers (TCP/IP)
- 2.3 Other relevant protocols
- 2.4 Putting it all together: Wireshark

Specific objectives:

To understand the layered structure used in networks, the process of encapsulation, as well as the usefulness of it all.
To learn about the functionality of each layer of abstraction, as well as to know the classical protocols in each layer, paying special attention in those that are relevant for the programming of on-line games (IP, TCP/UDP, NAT)
To learn about the use of packet sniffers, and to employ them to observe the packet structure in real networks

Related activities:

2. Sockets and the Client-Server architecture

Related competencies :

CGFB5VJ. Interpret the structure, operation and interconnection of computer systems, and the basics of their programming.

CEVJ 13. Undertake and manage video game design and development projects, including planning, direction, execution and evaluation.

07 AAT. SELF-DIRECTED LEARNING. Detecting gaps in one's knowledge and overcoming them through critical self-appraisal. Choosing the best path for broadening one's knowledge.

04 COE N3. EFFICIENT ORAL AND WRITTEN COMMUNICATION - Level 3. Communicating clearly and efficiently in oral and written presentations. Adapting to audiences and communication aims by using suitable strategies and means.

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: 45h

Theory classes: 10h

Guided activities: 8h

Self study : 27h



3. Impact of the network on the programming of on-line games

Description:

- 3.1 Introduction and general aspects. Object serialization and replication
- 3.2 Latency, jitter, and reliability
- 3.3 Latency hiding techniques
- 3.4 Scalability and massive multiplayer

Specific objectives:

To understand how the network impacts on the programming of on-line games through the study of problems related to the loss of information or delays

To learn about techniques used to hide the network problems, minimizing their impact so that the game performance is not affected

To learn about widespread techniques in nowadays games, be them with a few or many simultaneous players

Related activities:

- 3. Data serialization
- 4. Multiplayer game in C++
- 5. Networked games in Unity

Related competencies :

CEVJ 13. Undertake and manage video game design and development projects, including planning, direction, execution and evaluation.

CEVJ 5. Use programming languages, algorithmic patterns, data structures, visual programming tools, game engines and libraries for the development and prototyping of video games, in any genre and for any platform and mobile device.

07 AAT. SELF-DIRECTED LEARNING. Detecting gaps in one's knowledge and overcoming them through critical self-appraisal. Choosing the best path for broadening one's knowledge.

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: 75h

Theory classes: 10h

Guided activities: 20h

Self study : 45h

4. Graphs and neural networks

Description:

- 4.1 Graph theory
- 4.2 Algorithms over graphs
- 4.3 Neural network fundamentals

Specific objectives:

To learn about the fundamentals of graph theory and algorithms that operate over graphs, which are common to networks and many other domains

To understand a few concepts related to the neural networks fundamentals, as well as their relation with graphs and computer networks

Related competencies :

CGFB5VJ. Interpret the structure, operation and interconnection of computer systems, and the basics of their programming.

CEVJ 13. Undertake and manage video game design and development projects, including planning, direction, execution and evaluation.

CEVJ 5. Use programming languages, algorithmic patterns, data structures, visual programming tools, game engines and libraries for the development and prototyping of video games, in any genre and for any platform and mobile device.

07 AAT. SELF-DIRECTED LEARNING. Detecting gaps in one's knowledge and overcoming them through critical self-appraisal. Choosing the best path for broadening one's knowledge.

Full-or-part-time: 20h

Theory classes: 8h

Self study : 12h

ACTIVITIES

1. Threads

Description:

Learning how to use parallel code execution mechanisms (threads) and the necessary tools to maintain a proper synchronization amongst them, and to safely access shared memory regions in a concurrent fashion.

Specific objectives:

- Threads (mechanisms for parallel/concurrent code execution)
- Mutex objects (mechanism for safe concurrent data access)
- Condition variables (mechanism for synchronization among threads)

Material:

Instructions for the exercise and template code.

Related competencies :

CGFB5VJ. Interpret the structure, operation and interconnection of computer systems, and the basics of their programming.

CEVJ 5. Use programming languages, algorithmic patterns, data structures, visual programming tools, game engines and libraries for the development and prototyping of video games, in any genre and for any platform and mobile device.

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.

07 AAT. SELF-DIRECTED LEARNING. Detecting gaps in one's knowledge and overcoming them through critical self-appraisal. Choosing the best path for broadening one's knowledge.

Full-or-part-time: 5h

Guided activities: 2h

Self study: 3h



2. Sockets and Client-Server architecture

Description:

Learning the main mechanisms to use TCP and UDP sockets to transmit and receive data between two or more machines over the network.

Specific objectives:

- Sockets TCP (connection oriented)
- Sockets UDP (non-connection oriented)
- Non-blocking usage of sockets functions (non-blocking mode, select).

Material:

Instructions for the exercise and code template.

Related competencies :

CGFB5VJ. Interpret the structure, operation and interconnection of computer systems, and the basics of their programming.
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.

07 AAT. SELF-DIRECTED LEARNING. Detecting gaps in one's knowledge and overcoming them through critical self-appraisal. Choosing the best path for broadening one's knowledge.

Full-or-part-time: 20h

Guided activities: 8h

Self study: 12h

3. Data serialization

Description:

Here we will see the importance of data serialization to transmit objects in memory that may be potentially complex, as these objects could not be represented contiguously in memory, and can contain pointers to other objects, thus making impossible copying them directly from memory at once.

Specific objectives:

- Sequential and ordered data access.
- MemoryStream objects.

Material:

Instructions for the exercise and code template.

Related competencies :

CGFB5VJ. Interpret the structure, operation and interconnection of computer systems, and the basics of their programming.
CEVJ 5. Use programming languages, algorithmic patterns, data structures, visual programming tools, game engines and libraries for the development and prototyping of video games, in any genre and for any platform and mobile device.

07 AAT. SELF-DIRECTED LEARNING. Detecting gaps in one's knowledge and overcoming them through critical self-appraisal. Choosing the best path for broadening one's knowledge.

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: 5h

Guided activities: 2h

Self study: 3h



4. Multiplayer videogame in C++

Description:

We will put in practica all concepts learnt so far (threads, sockets and data serialization) to develop a multiplayer videogame in C++.

Specific objectives:

- . Client-server architecture.
- Authoritative server.
- Game state replication.
- Messages among hosts.

Material:

Instructions for the exercise and code template.

Related competencies :

CGFB5VJ. Interpret the structure, operation and interconnection of computer systems, and the basics of their programming.
CEVJ 13. Undertake and manage video game design and development projects, including planning, direction, execution and evaluation.

CEVJ 5. Use programming languages, algorithmic patterns, data structures, visual programming tools, game engines and libraries for the development and prototyping of video games, in any genre and for any platform and mobile device.

07 AAT. SELF-DIRECTED LEARNING. Detecting gaps in one's knowledge and overcoming them through critical self-appraisal. Choosing the best path for broadening one's knowledge.

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: 20h

Guided activities: 8h

Self study: 12h

5. Networked games in Unity

Description:

In this part of the subject we will learn how to develop a multiplayer game at a higher level using the tools provided by the Unity games engine. Instead of programming all the data serialization manually, we will use the existing pieces of this engine in order to send commands and game state among several PCs over the network.

Specific objectives:

- Main architecture of multiplayer games in Unity
- Main networking components
- Data synchronization mechanisms (commands, syncvars and hooks, client RCPs, etc).

Material:

Instructions for the exercise and code template.

Full-or-part-time: 25h

Guided activities: 10h

Self study: 15h

GRADING SYSTEM

Lab sessions: 40%

Partial exam: 20%

Final exam: 30%

Contribution and learning attitude of the student: 10%

In case the subject is not passed through the continuous evaluation there is the option to perform a re-evaluation exam of the theoretical part, corresponding to 50% of the subject's mark.



EXAMINATION RULES.

The evaluation activities are individual

For the practices it is allowed to use all learning material and their submission must be done on the indicated deadline with no option for extension

Exams will be done with no access to any learning material

BIBLIOGRAPHY

Basic:

- Tanenbaum, A.S.; Wetherall, D.J. Computer networks. 5th ed. Boston: Pearson, 2011. ISBN 9780132126953.
- Glazer, Josh; Madhav, Sanjay . Multiplayer game programming: Architecting networked games. Addison-Wesley Professional, 2015. ISBN 9780134034331.

RESOURCES

Other resources:

Additional readings that will be published in the Campus Virtual during the course, including:

- The history of Internet
- I shot you first: The Networking of HALO
- 1500 Archers on a 28.8: Network Programming in Age of Empires and Beyond
- The TRIBES Engine Networking Model