Operating systems design

Academic Year: (2020 / 2021) **Review date: 10/07/2020 17:14:07**

Department assigned to the subject:

Coordinating teacher: GARCIA GUZMAN, JAVIER

Type: Compulsory ECTS Credits : 6.0

Year : 3 Semester : 2

REQUIREMENTS (SUBJECTS THAT ARE ASSUMED TO BE KNOWN)

* Programming
* Software Development
* Operating Systems

OBJECTIVES

The goal of this course is to introduce students into the organization, structure and internal vision of the operating systems necessary for Internet of Things systems. Students will learn the services that integrate sensor and actuator based systems and understand the influence that design decisions have on the behavior of an Internet of Things System. In order to archive this goal, the student have to acquire several generic skills, knowledge, capacities and attitudes.

General/transversal competences:
- Analysis and synthesis capacities (PO a)
- Abilities to organize and to plan (PO a)
- Problem resolution abilities (PO c)
- Capacity to apply theoretical concepts (PO a, c)

Specific competences:
- Cognitive (knowledge) (PO a)
  1. Know the different types of devices and operating systems of an Internet of Things System.
  2. Know different types of use and applications of IoT technology in today's society.
  3. Know the structure of an Embedded Operating System that allows the operation of sensors and actuators to be controlled.
  4. Know the criteria for selecting the Embedded Operating Systems necessary to control sensors and actuators.
  5. Know the programming interfaces (APIs) that embedded systems provide to customize the operation of sensors and actuators.
  6. Know the fundamentals for programming IoT devices through the APIs provided by the Embedded Operating Systems.
  7. Know operating systems and base platforms to manage clouds of IoT devices.
  8. Know the fundamentals of operating systems to package and virtualize microservices to manage clouds of IoT devices.

- Procedimental/Instrumental (Know how) (PO b, e, j, k, g)
  1. Configure the Hardware and Operating System that controls the operation of sensors and actuators
  2. Program the basic operation of IoT devices using the APIs provided by the Operating System.
  3. Program basic microservices for IoT device cloud management
  4. Virtualize microservices to manage IoT device clouds

- Attitudinal (To be) (PO c, d, i)
1. Critical attitude towards the internal architecture of current IoT systems.
2. Concern for the quality of the components of an IoT system.
3. Motivation for archiving better solutions.

4. Self-learing capacities.

General and Transversal Competences
* To use in an efficient way electronic tools for writing technical report, project memos and reports about computer science, including high quality presentations (CG9)
* Basic knowledge about the usage and the programming of computers, operating systems, data bases, and computer applications with engineering applications (CGB4)

Computer Science's related competences
* Knowledge of characteristics, functionality and structure of operating systems, and to design and implement software based on its services (CECRI10)

DESCRIPTION OF CONTENTS: PROGRAMME

1. IoT Systems Architecture
2. Sensors and Actuators
3. Embedded Operating Systems for IoT devices
4. Fundamentals of programming IoT Devices
5. IoT Edge to Cloud Protocols
6. Microservices for IoT Devices Cloud Management
7. Platforms for IoT Device Cloud Management
8. Packaging and deployment of microservices for IoT

LEARNING ACTIVITIES AND METHODOLOGY

- Lectures (PO a)
- Practical lessons (PO a, b, c, e, k)
- Guided Exercises, Challenges and Exams (PO a, b, c, e)
- Student's work

Relationship between outcomes and evaluable activities:
* PO a, b, e, j: Exams
* PO c, k: labs, and guided labs
* PO d, g: labs
* PO i: guided labs

* CG9: Guided Exercises and Challenges
* CGB4: Guided Exercises and Challenges
* CECRI10: Guided Exercises and Challenges

Soft-skills: work in a group

ASSESSMENT SYSTEM

% end-of-term-examination/test:	30
% of continuous assessment (assigments, laboratory, practicals…):	70

ORDINARY CALL
The continuous evaluation is made up of: Notebooks of Practical Exercises, Challenges and, exceptionally, partial theory exams.

Regarding continuous evaluation:
+ All practical exercise books and challenges must be delivered
  * The average mark of the practices and challenges must be greater than or equal to 5 out of 10.
+ Exceptionally, the final exam of the subject may be replaced by partial exams
  * The average grade of the partial exams must be greater than or equal to 5 out of 10.

The final exam:
+ Includes all the content and aspects of the subject.
+ The final exam has a minimum grade of 5 out of 10.

The grade of the subject (by following the continuous evaluation) is calculated as follows:
Final grade = Final exam grade * 0.30 + Practical exercises notebooks * 0.40 + Final challenge * 0.3

% end-of-term-examination/test:	30
% of continuous assessment (assigments, laboratory, practicals…):	70

Alternatively, the final exam will represent 60% of the grade and the Final Challenge 40%, for all those students who do not join the previous continuous assessment system.
The minimum grade of each of the parts must be equal to or greater than 5 out of 10.

EXTRAORDINARY CALL
The final exam will represent 60% of the grade and the Final Challenge 40%.
The minimum grade of each of the parts must be equal to or greater than 5 out of 10.

CLARIFICATIONS
1) Cheating programming assignments, lab assignments, etc. will be prosecuted. Not only the continuous evaluation is lost but further punishment actions will be taken. Both, copying from other group or copying from Internet works are considered cheating.
2) For all group works, the related skills and abilities must be fulfilled by all members. All members share the responsibility for the submitted work.
3) Depending on the total number of students, the academic calendar, number of groups, etc., some adaptations will be necessary in order to unwrap the course. Aula Global will be used to introduce the supplementary norms to the ones introduced here.
4) In order to obtain the best possible grade, not only academic aspects will be considered but also how the student has been collaborating in the general learning process of the course.

BASIC BIBLIOGRAPHY

- Anand Tamboli Build Your Own IoT Platform: Develop a Fully Flexible and Scalable Internet of Things Platform in 24 Hours, Apress, 2019

- Bob Familiar Microservices, IoT, and Azure: Leveraging DevOps and Microservice Architecture to Deliver SaaS Solutions, Apress, 2015

- Gabriel N. Schenker, Hideto Saito, Hui-Chuan Chloe Lee, Ke-Jou Carol Hsu Getting Started with Containerization, Packt Publishing, 2019

- Perry Lea Internet of Things for Architects, Packt Publishing, 2018

- Peter Waher Mastering Internet of Things, Packt Publishing, 2018

- Qusay F. Hassan Internet of Things A to Z, Wiley-IEEE Press, 2018

- Richard Blum, Christine Bresnahan Sams Teach Yourself Python Programming for Raspberry Pi in 24 Hours, Second Edition, Sams, 2015