DEGREE CURRICULUM

PROGRAMMING AND COMMUNICATIONS III

Coordination: ONRUBIA PALACIOS, JORDI RICARD

Academic year 2023-24

PROGRAMMING AND COMMUNICATIONS III 2023-24

Subject's general information

<table>
<tr><td>Subject name</td><td>PROGRAMMING AND COMMUNICATIONS III</td></tr>
<tr><td>Code</td><td>102135</td></tr>
<tr><td>Semester</td><td>2nd Q(SEMESTER) CONTINUED EVALUATION</td></tr>
<tr><td>Typology</td><td>
<table>
<tr><th>Degree</th><th>Course</th><th>Character</th><th>Modality</th></tr>
<tr><td>Bachelor's Degree in Automation and Industrial Electronic Engineering</td><td>4</td><td>OPTIONAL</td><td>Attendance-based</td></tr>
</table>
</td></tr>
<tr><td>Course number of credits (ECTS)</td><td>6</td></tr>
<tr><td>Type of activity, credits, and groups</td><td>
<table>
<tr><th>Activity type</th><td>PRAULA</td><td>TEORIA</td></tr>
<tr><th>Number of credits</th><td>3</td><td>3</td></tr>
<tr><th>Number of groups</th><td>1</td><td>1</td></tr>
</table>
</td></tr>
<tr><td>Coordination</td><td>ONRUBIA PALACIOS, JORDI RICARD</td></tr>
<tr><td>Department</td><td>COMPUTER ENGINEERING AND DIGITAL DESIGN</td></tr>
<tr><td>Teaching load distribution between lectures and independent student work</td><td>6 ECTS = 25x6 = 150
- 60 hours of on-class activities
- 90 hours of autonomous activities</td></tr>
<tr><td>Important information on data processing</td><td>Consult this link for more information.</td></tr>
<tr><td>Language</td><td>English</td></tr>
<tr><td>Distribution of credits</td><td>ONRUBIA PALACIOS, JORDI RICARD (6)</td></tr>
</table>

PROGRAMMING AND COMMUNICATIONS III 2023-24

Teaching staff	E-mail addresses	Credits taught by teacher	Office and hour of attention
ONRUBIA PALACIOS, JORDI RICARD	jordi.onrubia@udl.cat	6	Agreed by e-mail

Learning objectives

Expected learning outcomes related to the strategic transversal competences:

- The student is able to learn and work with technical documentation in English on programming languages on Linux environment (Competences UdL2, EPS4).
- The student is able to work in teams to carry out the development of a computer system composed of different subsystems in which there to apply knowledge from different fields (Competence EPS9).
- The student is able to prepare presentations in English to show aspects main programs developed for other equipment engineers can understand their solutions. (Competences UdL2, UdL3).
- The student is able to generalize basic algorithmic schemes to apply them in different contexts and problems from the ones initially seen (Competence EPS4).

Expected learning outcomes linked to specific competencies:

- The student is able to understand the basic features of the functioning of operating systems multitasking, multiuser based on the Linux kernel (Competences GEEIA3 and GEEIA34).
- The student is able to integrate the knowledge of circuits, sensors and processes industrial on knowledge of computer programming for address the full development of small automated systems monitoring / control software based processes on Raspberry Pi computers or microcomputers (Competences GEEIA3, GEEIA28 and GEEIA34).

Competences

Strategic competences UdL:

- UdL2. Knowledge of an foreign language.
- UdL3. Knowledge of ICT.

Transversal competences EPS:

- EPS4. Have the learning abilities needed to start superior studies or improve academic learning with a certain autonomous degree.
- EPS9. Ability to work in teams, both as an interdisciplinary and multidisciplinary.

Specific competences GEEIA:

- GEEIA3 Basic knowledge on using and programming computers, operating systems, databases and software with applications in engineering.
- GEEIA28. Applied knowledge of industrial computing and communications.
- GEEIA-EPS34. Knowledge of the fundamentals of computer systems and applications.

PROGRAMMING AND COMMUNICATIONS III 2023-24

Subject contents

- Introduction to computer networks:
  - OSI / ISO layer models
  - TCP / IP model.
  - Introduction to IP.
  - Introduction to TCP.
  - Introduction to HTTP.
  - Python sockets
- Introduction to Docker
- Introduction to Databases
  - SQL
  - DBeaver
- Web Application Development
  - REST
  - Requests
  - FastAPI
- Deployment
  - External Services
  - Raspberry
- Databases extension
  - Timeseries
    - Timescale DB
  - NoSQL
    - MongoDB
- Visualisation
  - Grafana

Methodology

Lectures.

Exercices.

Development of small mini projects and small parts of the whole project.

Development plan

PROGRAMMING AND COMMUNICATIONS III 2023-24

Week	Description	Face-to-Face Activity	Autonomous Activity	Hours (F and A)
1	Presentation and introduction to communication with microcomputers	Lectures and programming laboratory	**Solve Exercises**	**4 2**
2	Internet	Lectures and programming laboratory	**Solve Exercises**	4 6
3	IP Sockets	Lectures and programming laboratory	**Solve Exercises**	4 6
4	Sockets exercices and Project 1 presentation	Lectures and programming laboratory	Work on programming assignment/s **Solve Exercises**	4 6
5	Project 1 - Doubts and guidance	Lectures and programming laboratory	Work on programming assignment/s	**4 6**
6	SQL, ORM and database tools	Lectures and programming laboratory	**Solve Exercises**	4 6
7	SQL-ORM related exercices and Project 2 presentation	Lectures and programming laboratory	Work on programming assignment/s **Solve Exercises**	4 8
8	Project 2 - Doubts and guidance	Lectures and programming laboratory	Work on programming assignment/s	4 8
9	REST-Request- FastAPI	Lectures and programming laboratory	Work on programming assignment/s	4 8
10	Project 3 - Presentation doubts and guidance	Lectures and programming laboratory	Work on programming assignment/s	4 6
11	Project 4 - Presentation doubts and guidance	Lectures and programming laboratory	Work on programming assignment/s	4 6
12	Docker	Lectures and programming laboratory	Work on programming assignment/s	4 6
13	Project 5 - Presentation doubts and guidance	Lectures and programming laboratory	Work on programming assignment/s	4 10
14	Deployment	Lectures and programming laboratory	Work on programming assignment/s	4 8
15	Database Extension	Lectures and programming laboratory	Work on programming assignment/s	4 8
16	Visualisation	Lectures and programming laboratory	Work on programming assignment/s	- 6

17	General Project - Doubts and guidance	Lectures and programming laboratory	Work on programming assignment/s	2 4
18	General Project - Doubts and guidance	Lectures and programming laboratory	Work on programming assignment/s	
19	General Project - Doubts and guidance	Lectures and programming laboratory	Work on programming assignment/s	

This planification can be shortened in case of issues (holidays, sickness, ...)

Evaluation

Acr	Evaluation Activity	Weight	Minimum Grade	In Group	
P1	Programming assignment (1)	20%	3	YES	YES
P2	Programming assignment (2)	20%	3	YES	YES
P3	Programming assignment (3)	15%	3	YES	YES
P4	Programming assignment (4)	25%	3	YES	YES
P5	Programming assignment (5)	10%	3	YES	YES
Pr	Exercises	10%	NO	NO	NO

FinalGrade = 0,2*P1 + 0,2*P2 + 0,15*P3 + 0,25*P4 + 0,10*P5 + 0,10*PR
If any grade is lesser than the minimum, the maximum grade that can be obtained from the sum of all the grades is 4

Each programming assignment will have its own document with the requirements explained by steps.

All assignments will be evaluated based on the programming of the required elements expressed in each Programming Assignment document, and the proper documentation of the implementation.

The grade of each assignment will be given by the completeness of each task described inside the document and the cleanliness of the code.

The documentation must be clear and describe in an understandable way the implementation given by the student. It is required to specify the sources of the different tools and solutions from external sources.

The last day to present all the assignments will be the last day of the last week of exams.

The last day to re-present the failed assignments will be the last day of the recovery week.

Assignments can only be presented if they have a grade lesser than 5.

The maximum grade that can be obtained in a recovered assignment is 8.

Bibliography

Documentation and examples in the virtual campus.

http://appinventor.mit.edu/explore/get-started

http://appinventor.mit.edu/explore/ai2/tutorials

Think Python - Allen B. Downey (http://www.greenteapress.com/thinkpython/)

Learn Python the Hard Way - Zed Shaw (http://learnpythonthehardway.org/)