ANX-PR/CL/001-01

LEARNING GUIDE

SUBJECT

593000410 - Advanced Virtual Instrumentation Systems

DEGREE PROGRAMME

59AF - Master Univ. Ing. Sistemas Y Servicios Para La Sociedad De La Informacion

ACADEMIC YEAR & SEMESTER

2019/20 - Semester 2

Index

Learning guide

1. Description

1.1. Subject details

Name of the subject	593000410 - Advanced Virtual Instrumentation Systems
No of credits	5 ECTS
Type	Optional
Academic year ot the programme	First year
Semester of tuition	Semester 2
Tuition period	February-June
Tuition languages	English
Degree programme	59AF - Master Univ. Ing. Sistemas Y Servicios Para La Sociedad De La Informacion
Centre	59 - Escuela Tecnica Superior de Ingenieria y Sistemas de Telecomunicacion
Academic year	2019-20

2. Faculty

2.1. Faculty members with subject teaching role

Name and surname	Office/Room	Email	Tutoring hours *
Francisco Javier Jimenez Martinez (Subject coordinator)	A4202	franciscojavier.jimenez@upm.es	Sin horario.

* The tutoring schedule is indicative and subject to possible changes. Please check tutoring times with the faculty member in charge.

3. Prior knowledge recommended to take the subject

3.1. Recommended (passed) subjects

The subject - recommended (passed), are not defined.

3.2. Other recommended learning outcomes

- Engineering Programming (at least basic level in some programming language)

- Electronic instrumentation: sensors and transducers, signal conditioning, A / D and D / A conversion, structure of data acquisition systems

4. Skills and learning outcomes *

4.1. Skills to be learned

CB10 - To have the learning abilities to continue studying in a mostly self-guided or autonomous manner.

CB6 - To have knowledge that provides the basis or the opportunity of being original to develop and/or to apply ideas, usually in a research context.

CB7 - To be capable of applying the students' acquired knowledge, as well as their problem solving abilities, to new or not well-known environments in broader (or multidisciplinary) contexts that are in the framework of their expertise area.

CE.1 - To be capable of analyzing, interpreting and applying standards related to the ICT.

CE.7 - To be capable of proposing, organizing and executing research works in the framework of the Information Society engineering.

CESI.5 - To be capable of implementing data acquisition systems and automated testing systems by using advanced instrumentation tools.

CGEN.3 - To be capable of elaborating, planning strategically, leading, coordinating and managing, both technically and economically, projects in the framework of the Information Society engineering, according to ethical, quality and environmental criteria.

CGEN.4 - To be capable of planning, calculating and designing systems and services for the Information Society.

4.2. Learning outcomes

RA13 - Ability to specify hardware and software technologies of virtual instrumentation systems for the development of data acquisition systems and test benches

RA18 - Use FPGA-based technologies in the development of data acquisition applications and test beds

RA16 - Develop advanced data acquisition applications including programming of synchronization and hardware trigger functions

RA14 - Apply software development methodologies in the development of virtual instrumentation systems

RA15 - Manage software development tools for virtual instrumentation systems

RA17 - Develop data acquisition applications and test beds using real-time technologies

* The Learning Guides should reflect the Skills and Learning Outcomes in the same way as indicated in the Degree Verification Memory. For this reason, they have not been translated into English and appear in Spanish.

5. Brief description of the subject and syllabus

5.1. Brief description of the subject

This course introduces the student in the handling of Virtual Instrumentation technologies gradually starting from scratch. The first block deals with the development of simple data acquisition applications with a graphic programming tool widely used in the sector. In a second block, the use of more advanced techniques or technologies is explored, such as the control of low-level acquisition, real-time systems, and the programming of FPGAs from high-level graphic languages.

5.2. Syllabus

1. Introduction to Virtual Instrumentation
2. LabVIEW Programming
3. Applications design
4. Introduction to Data Acquisition
5. Advanced Data Acquisition
6. Real Time Systems: LabVIEW RT
7. FPGAs in Virtual Instrumentation

6. Schedule

6.1. Subject schedule*

Week	Face-to-face classroom activities	Face-to-face laboratory activities	Other face-to-face activities	Assessment activities
1		**Topic 1** Duration: 02:00 **Topic 2** Duration: 00:30 **Topic 2** Duration: 00:30		
2		**Topic 2** Duration: 01:30 **Topic 2** Duration: 01:30		
3		**Topic 3** Duration: 01:00 **Topic 3** Duration: 02:00		
4		**Topic 3** Duration: 03:00		**Assignment 1** Continuous assessment Duration: 00:20
5		**Topic 4** Duration: 01:00 **Topic 4** Duration: 02:00		
6		**Topic 4** Duration: 03:00		
7		**Assignment 2** Duration: 03:00		

8		**Assignment 2** Duration: 02:40		
9		**Topic 5** Duration: 01:00 **Topic 5** Duration: 02:00		**Assignment 2** Continuous assessment Duration: 00:20
10		**Topic 6** Duration: 02:00 **Topic 6** Duration: 01:00		
11		**Topic 6** Duration: 03:00		
12		**Toipic 6** Duration: 02:00 **Topic 6** Duration: 01:00		**RT Evaluation** Continuous assessment Duration: 00:20
13		**Topic 7** Duration: 03:00		
14		**Topic 7** Duration: 01:00 **Topic 7** Duration: 02:00		
15		**Topic 7** Duration: 03:00		**FPGA Evaluation** Continuous assessment Duration: 00:20
16				
17				**One written/practical exam (individually) about Assigmments that previously (the week before exam) must be developed by the student in the laboratory** Final examination Duration: 03:00

The independent study hours are training activities during which students should spend time on individual study or individual assignments.

Depending on the programme study plan, total values will be calculated according to the ECTS credit unit as 26/27 hours of student face-to-face contact and independent study time.

* The subject schedule is based on a previous theorical planning of the subject plan and might go to through

experience some unexpected changes along throughout the academic year.

7. Activities and assessment criteria

7.1. Assessment activities

7.1.1. Continuous assessment

Week	Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
4	Assignment 1		Face-to-face	00:20	20%	5 / 10	CB10
9	Assignment 2		Face-to-face	00:20	40%	5 / 10	CB10 CB7 CESI.5 CGEN.4
12	RT Evaluation		Face-to-face	00:20	25%	5 / 10	CB10 CE.7 CB6 CB7 CESI.5 CGEN.4 CE.1
15	FPGA Evaluation		Face-to-face	00:20	15%	5 / 10	CB10 CE.7 CB6 CB7 CGEN.3 CESI.5 CGEN.4 CE.1

7.1.2. Final examination

Week	Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
17	One written/practical exam (individually) about Assigmments that previously (the week before exam) must be developed by the student in the laboratory		Face-to-face	03:00	100%	5 / 10	CB10 CE.7 CB6 CB7 CGEN.3 CESI.5 CGEN.4 CE.1

7.1.3. Referred (re-sit) examination

Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
One written/practical exam (individually) about Assigmments that previously (the week before exam) must be developed by the student in the laboratory		Face-to-face	03:00	100%	5 / 10	CB10 CE.7 CB6 CB7 CGEN.3 CESI.5 CGEN.4 CE.1

7.2. Assessment criteria

1. Evaluation of the guided practices. The following will be valued: the clarity, organization and documentation of the developed code
2. Knowledge and skill in the contents that are presented. Clarity and detail in the answers to the questions made by the teacher during the evaluations.
3. Degree of self-learning in the use of tools. The teacher will raise problems that the student should try to solve within a fixed period.

In the evaluation of the design, the grade obtained by each student will be based on the assessment obtained in the following aspects:

- **Functionality and improvements made:** Covers up to 35% of the design note and tries to verify whether the program performed meets the functionality stated in the statement and if the students have included improvements that have not been specified in the statement.

- **Programming style:** Covers up to 25% of the design note. The students must make an application with a Design Pattern defined so that the software made is as efficient as possible, scalable and maintainable.
- **Documentation of the code made:** Covers up to 20% of the design note. All VIs must be properly documented, and the programming style must be such that all VIs are perfectly legible.
- **User Interface:** Covers up to 15% of the note. The students should design a user interface that is friendly and simple.

- **Appreciation of the teacher on the knowledge of LabVIEW and the software developed by each of the students:** Covers 5% of the grade. The teacher after the development of the classes and the meeting with the students on the date and time prefixed before will assess the knowledge acquired in this part of the subject. If the teacher detects that any of the students does not know the programming language and / or does not know in depth the design that has been delivered, it will suppose the failure in the design independently of the evaluation obtained in points 1 to 4.

FINAL PROOF OF THE SUBJECT:

Students who choose to take a final test must meet the following conditions:

1. Having properly processed the application for the final test of the subject
2. Have been evaluated in at least 50% of the directed practices and have attended at least 50% of the face-to-face classes of the subject.

The final test will consist of:

1. Exercises similar to those carried out in class that cover the subjects taught in the whole subject. Teachers will ask explicit questions for each exercise.
2. The realization of a design similar to the one made in the subject, evaluating the aspects described in the design evaluation.
3. Said test will be carried out on the same day and in morning and afternoon sessions.

8. Teaching resources

8.1. Teaching resources for the subject

Name	Type	Notes
A Software Engineering Approach to LabVIEW. Jon Conway/Steve Watts. Prentice Hall.	Bibliography	
LabVIEW Graphical Programming and LabVIEW Power Programming by Gary W. Johnson, McGraw-Hill	Bibliography	

PCI6024, NI MyRIO, LabVIEW+FPGA+RealTime	Equipment	
www.ni.com	Web resource	
www.info-labview.org	Web resource	
Class Slides	Bibliography	Class slides to follow teachers explanations. Available in Moodle

9. Other information

9.1. Other information about the subject

5 ECTS

49 h Class attendance (exams included) ~ 3-4 h / week

71 h work not attendded ~ 4-5 h / week

120 h total ~ 8 h / week