



POLITÉCNICA

INTERNATIONAL
CAMPUS OF
EXCELLENCE

COORDINATION PROCESS OF
LEARNING ACTIVITIES
PR/CL/001



E.T.S. de Ingenieros
Informáticos

ANX-PR/CL/001-01

LEARNING GUIDE

SUBJECT

103000385 - Software Verification And Validation

DEGREE PROGRAMME

10AK - Master Universitario En Software Y Sistemas

ACADEMIC YEAR & SEMESTER

2023/24 - Semester 1

Index

Learning guide

1. Description.....	1
2. Faculty.....	1
3. Prior knowledge recommended to take the subject.....	2
4. Skills and learning outcomes	2
5. Brief description of the subject and syllabus.....	4
6. Schedule.....	5
7. Activities and assessment criteria.....	7
8. Teaching resources.....	11

1. Description

1.1. Subject details

Name of the subject	103000385 - Software Verification And Validation
No of credits	4 ECTS
Type	Optional
Academic year of the programme	First year
Semester of tuition	Semester 1
Tuition period	September-January
Tuition languages	English
Degree programme	10AK - Master Universitario en Software y Sistemas
Centre	10 - Escuela Tecnica Superior De Ingenieros Informaticos
Academic year	2023-24

2. Faculty

2.1. Faculty members with subject teaching role

Name and surname	Office/Room	Email	Tutoring hours *
Sira Vegas Hernandez (Subject coordinator)	5105	sira.vegas@upm.es	M - 13:00 - 16:00 Th - 14:00 - 17:00
Natalia Juristo Juzgado	5104	natalia.juristo@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

- Programación
- Lenguajes de programación C y JAVA

4. Skills and learning outcomes *

4.1. Skills to be learned

CEM1 - Identificar, a partir del estado de la cuestión, la presencia de problemas de investigación relacionados con la concepción, la construcción, el uso y la evaluación de sistemas sociotécnicos complejos que hagan un uso intensivo de software

CEM4 - Analizar y evaluar los diferentes paradigmas y enfoques de ingeniería de construcción y gestión de sistemas basados en software.

CEM5 - Aportar soluciones a aquellos problemas abiertos relacionados con el ámbito de aplicación y los métodos, técnicas y herramientas de Verificación y Validación de Software

CG12 - Comprensión amplia de las técnicas y métodos aplicables en una especialización concreta, así como de sus límites

CG13 - Apreciación de los límites del conocimiento actual y de la aplicación práctica de la tecnología más reciente.

CG14 - Conocimiento y comprensión de la informática necesaria para la creación de modelos de información, y de los sistemas y procesos complejos

CG4 - Que los estudiantes posean las habilidades de aprendizaje que les permitan continuar estudiando de un modo que habrá de ser en gran medida autodirigido o autónomo.

CG7 - Especificación y realización de tareas informáticas complejas, poco definidas o no familiares

CG8 - Planteamiento y resolución de problemas también en áreas nuevas y emergentes de su disciplina

CG9 - Aplicación de los métodos de resolución de problemas más recientes o innovadores y que puedan implicar el uso de otras disciplinas

CG120 - Adquirir conocimientos científicos avanzados del campo de la informática que le permitan generar nuevas ideas dentro de una línea de investigación.

CG123 - Capacidad de leer y comprender publicaciones dentro de su ámbito de estudio/investigación, así como su catalogación y valor científico

4.2. Learning outcomes

RA42 - Seleccionar la técnica de verificación/validación de software más adecuada para un proyecto determinado

RA41 - Aplicar efectivamente las técnicas de verificación y validación de software

* 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

No hay descripción de la asignatura.

5.2. Syllabus

1. Introduction

- 1.1. Introduction to V&V
- 1.2. V&V and the software development process
- 1.3. V&V and the software development products

2. Dynamic evaluation: Software testing

- 2.1. Introduction to software testing
- 2.2. Testing levels
- 2.3. The testing process
- 2.4. Software verification and validation plan
- 2.5. Testing tools

6. Schedule

6.1. Subject schedule*

Week	Classroom activities	Laboratory activities	Distant / On-line	Assessment activities
1	Introducción Duration: 02:00 Lecture			
2	Introducción Duration: 02:00 Lecture			
3	Evaluación dinámica Duration: 02:00 Lecture			
4	Evaluación dinámica Duration: 02:00 Problem-solving class			
5	Evaluación dinámica: Seguimiento del white box exercise Duration: 02:00 Additional activities			White box exercise Individual work Continuous assessment Not Presential Duration: 00:00
6	Evaluación dinámica Duration: 02:00 Lecture			
7	Evaluación dinámica Duration: 02:00 Problem-solving class			
8	Evaluación dinámica: Seguimiento del black box exercise Duration: 02:00 Additional activities			Black box exercise Individual work Continuous assessment Not Presential Duration: 00:00
9	Evaluación dinámica Duration: 02:00 Lecture			
10	Evaluación dinámica Duration: 02:00 Problem-solving class			
11	Seguimiento de la práctica Duration: 02:00 Cooperative activities			
12	Seguimiento de la práctica Duration: 02:00 Cooperative activities			Assignment: testing a software system (part 1) Group work Continuous assessment Not Presential Duration: 00:00

13	<p>Seguimiento de la práctica Duration: 02:00 Cooperative activities</p>			
14	<p>Seguimiento de la práctica Duration: 02:00 Cooperative activities</p>			<p>Assignment: testing a software system (part 2) Group work Continuous assessment Not Presential Duration: 00:00</p>
15	<p>Plan de Pruebas Duration: 02:00 Lecture</p>			
16	<p>Plan de Pruebas Duration: 02:00 Cooperative activities</p>			
17				<p>Attendance Other assessment Continuous assessment Presential Duration: 00:00</p> <p>White box exercise (resubmission) Individual work Final examination Not Presential Duration: 00:00</p> <p>Black box exercise (resubmission) Individual work Final examination Not Presential Duration: 00:00</p> <p>Assignment: testing a software system (resubmission of part 1) Individual work Final examination Not Presential Duration: 00:00</p> <p>Assignment: testing a software system (resubmission of part 2) Group work Final examination Not Presential Duration: 00:00</p>

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 schedule is based on an a priori planning of the subject; it might be modified during the academic year, especially considering the COVID19 evolution.

7. Activities and assessment criteria

7.1. Assessment activities

7.1.1. Assessment

Week	Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
5	White box exercise	Individual work	No Presential	00:00	15%	3 / 10	CEM1 CEM5 CG7 CG12 CG14 CEM4 CGI23 CG4
8	Black box exercise	Individual work	No Presential	00:00	15%	3 / 10	CG4 CEM1 CEM5 CG7 CG12 CG14 CEM4 CGI23
12	Assignment: testing a software system (part 1)	Group work	No Presential	00:00	30%	3 / 10	CG4 CG8 CG9 CEM1 CEM5 CG13 CGI20 CEM4
14	Assignment: testing a software system (part 2)	Group work	No Presential	00:00	30%	3 / 10	CG8 CG9 CEM1 CEM5 CG13 CGI20 CEM4
17	Attendance	Other assessment	Face-to-face	00:00	10%	8 / 10	CG4 CG7 CG12 CG13 CG14

7.1.2. Global examination

Week	Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
17	White box exercise (resubmission)	Individual work	No Presential	00:00	15%	5 / 10	CG4 CG8 CG9 CEM1 CEM5 CG13 CGI20 CEM4
17	Black box exercise (resubmission)	Individual work	No Presential	00:00	15%	5 / 10	CG4 CG8 CG9 CEM1 CEM5 CG13 CGI20 CEM4
17	Assignment: testing a software system (resubmission of part 1)	Individual work	No Presential	00:00	30%	5 / 10	CG4 CG8 CG9 CEM1 CEM5 CG13 CGI20 CEM4
17	Assignment: testing a software system (resubmission of part 2)	Group work	No Presential	00:00	30%	5 / 10	CG8 CG9 CEM1 CEM5 CG13 CGI20 CEM4

7.1.3. Referred (re-sit) examination

Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
White box exercise (second resubmission)	Individual work	Face-to-face	00:00	15%	5 / 10	CG4 CG8 CG9 CEM1 CEM5 CG13 CGI20 CEM4

Black box exercise (second resubmission)	Individual work	Face-to-face	00:00	15%	5 / 10	CG4 CG8 CG9 CEM1 CEM5 CG13 CGI20 CEM4
Assignment: testing a software system (second resubmission of part 1)	Individual work	Face-to-face	00:00	30%	5 / 10	CG4 CG8 CG9 CEM1 CEM5 CG13 CGI20 CEM4
Assignment: testing a software system (second resubmission of part 2)	Group work	Face-to-face	00:00	30%	5 / 10	CG4 CG8 CG9 CEM1 CEM5 CG13 CGI20 CEM4

7.2. Assessment criteria

Progressive evaluation period:

The score of the course is calculated regarding the performance of the student in the different tasks that (s)he has been assigned:

Exercise applying white box techniques to a program (15% of the score).

Exercise applying black box techniques to a program (15% of the score).

- Assignment performing testing on a software system (60% of the score). This assignment will be divided into two parts, submitted separately.

It will also be taken into consideration for the score of the course attendance to the lectures (10% of the score). A minimum of 80% of attendance is required to pass this evaluation. **This task is unrecoverable.** Students that have a justification for not being able to fulfill this criterion (e.g. conciliation issues, health problems, etc.) will be offered an alternative to pass it.

Global evaluation:

When the overall score obtained by the student in the progressive evaluation period is smaller than 5, the student will have to re-submit:

- All exercises/assignments that do not reach the minimum score required.
- From those exercises/assignments that do reach the minimum required, but have a score smaller than 5, the student will choose which ones (s)he wants to re-submit.
- In no case exercises/assignments that have a score equal or greater than 5 will be re-submitted.
- The score for the attendance criterion will be taken from the score obtained during the progressive evaluation period. In case the student has not reached the minimum score to pass this criterion during the progressive evaluation period, the global evaluation will be scored out of 9 instead of 10.

Note that during global evaluation, the student can re-submit those exercises/assignments that have been submitted during the progressive evaluation period. It is not possible to submit exercises/assignments for which there is not a submission in the progressive evaluation period.

A minimum score of 5 is needed to pass the course.

Extraordinary evaluation:

When the overall score obtained by the student in the global evaluation period is smaller than 5, the student will have to re-submit (or submit in case (s)he has not done it before):

- All exercises/assignments that do not reach the minimum score required.
- From those exercises/assignments that do reach the minimum required, but have a score smaller than 5, the student will choose which ones (s)he wants to re-submit.
- In no case exercises/assignments that have a score equal or greater than 5 will be re-submitted.
- The score for the attendance criterion will be taken from the score obtained during the progressive evaluation period. In case the student has not reached the minimum score to pass this criterion during the progressive evaluation period, the global evaluation will be scored out of 9 instead of 10.

A minimum score of 5 is needed to pass the course.

8. Teaching resources

8.1. Teaching resources for the subject

Name	Type	Notes
B. Beizer. "Software Testing Techniques" 2ª Edición. 1990	Bibliography	
G. J. Myers. "The Art of Software Testing" 2ª Edición. Wiley. 2004.	Bibliography	
P.C. Jorgensen. Software Testing. A Craftsman?s Approach. CRC Press, 1995.	Bibliography	
C. Kaner, J. Falk, H.Q. Nguyen. Testing Computer Software. Wiley, 1999.	Bibliography	
W.E. Perry. Effective methods for software testing. Tercera edición. Wiley. 2006	Bibliography	
S.L. Pfleeger. Ingeniería de software: teoría y práctica. Segunda edición. Prentice Hall. 2002	Bibliography	
IEEE V&V standards	Bibliography	
Sitio Moodle de la asignatura	Web resource	