



UNIVERSIDAD
POLITÉCNICA
DE MADRID

PROCESO DE
COORDINACIÓN DE LAS
ENSEÑANZAS PR/CL/001



E.T.S. de Ingenieros
Informaticos

ANX-PR/CL/001-01

GUÍA DE APRENDIZAJE

ASIGNATURA

103000611 - Profundizacion En Ingenieria Del Software

PLAN DE ESTUDIOS

10AN - Master Universitario En Ingenieria Informatica

CURSO ACADÉMICO Y SEMESTRE

2024/25 - Primer semestre

Índice

Guía de Aprendizaje

1. Datos descriptivos.....	1
2. Profesorado.....	1
3. Competencias y resultados de aprendizaje.....	2
4. Descripción de la asignatura y temario.....	3
5. Cronograma.....	4
6. Actividades y criterios de evaluación.....	8
7. Recursos didácticos.....	10
8. Otra información.....	11

1. Datos descriptivos

1.1. Datos de la asignatura

Nombre de la asignatura	103000611 - Profundizacion en Ingenieria del Software
No de créditos	3 ECTS
Carácter	Obligatoria
Curso	Primer curso
Semestre	Primer semestre
Período de impartición	Septiembre-Enero
Idioma de impartición	Castellano
Titulación	10AN - Master Universitario en Ingenieria Informatica
Centro responsable de la titulación	10 - Escuela Tecnica Superior De Ingenieros Informaticos
Curso académico	2024-25

2. Profesorado

2.1. Profesorado implicado en la docencia

Nombre	Despacho	Correo electrónico	Horario de tutorías *
Oscar Dieste Tubio (Coordinador/a)	D6203	oscar.dieste@upm.es	M - 16:00 - 19:00 X - 17:00 - 19:00 V - 16:00 - 17:00
Natalia Juristo Juzgado	D5104	natalia.juristo@upm.es	Sin horario. El horario de tutorías está disponible en Moodle.

* Las horas de tutoría son orientativas y pueden sufrir modificaciones. Se deberá confirmar los horarios de tutorías

con el profesorado.

3. Competencias y resultados de aprendizaje

3.1. Competencias

CE1 - Capacidad para la integración de tecnologías, aplicaciones, servicios y sistemas propios de la Ingeniería Informática, con carácter generalista, y en contextos más amplios y multidisciplinares.

CE4 - Capacidad para modelar, diseñar, definir la arquitectura, implantar, gestionar, operar, administrar y mantener aplicaciones, redes, sistemas, servicios y contenidos informáticos.

CE8 - Capacidad para analizar las necesidades de información que se plantean en un entorno y llevar a cabo en todas sus etapas el proceso de construcción de un sistema de información.

CG10 - Conocimiento y comprensión de la informática necesaria para la creación de modelos de información, y de los sistemas y procesos complejos

CG6 - Capacidad de pensamiento creativo con el objetivo de desarrollar enfoques y métodos nuevos y originales

3.2. Resultados del aprendizaje

RA206 - Integrar procesos de desarrollo y operaciones

RA205 - Planificar y ejecutar pruebas del software

RA204 - Aplicar técnicas de mejora de calidad del software dentro del proceso de desarrollo

4. Descripción de la asignatura y temario

4.1. Descripción de la asignatura

Este curso tiene como objetivo proporcionar a los alumnos conocimientos acerca de las técnicas de desarrollo de software utilizadas en disciplinas ágiles, tales como el desarrollo dirigido por tests, test automatizados y coordinación mediante repositorios. Se hará especial énfasis en los aspectos de calidad. A partir del año académico 2019/2020, se ha introducido el concepto de DevOps (unión de desarrollo y operaciones), abordando conceptos de integración continua, entrega continua, provisión programática de recursos y monitorización.

4.2. Temario de la asignatura

1. Métodos y prácticas ágiles
2. Repositorios de código
3. Estimación temprana
4. Herramientas de integración y entrega continua
5. Frameworks de prueba de unidad
6. Pruebas del software (estructurales y funcionales)
7. Mocking
8. Diseño testable
9. Frameworks de pruebas de sistema automatizadas (e.g., selenium)
10. Calidad del código
11. IA aplicada a la programación
12. Contratos
13. Test-driven development/Acceptance test-driven development/Behavior-driven development

5. Cronograma

5.1. Cronograma de la asignatura *

Sem	Actividad tipo 1	Actividad tipo 2	Tele-enseñanza	Actividades de evaluación
1	<p>Objetivos y metodología Duración: 00:15 LM: Actividad del tipo Lección Magistral</p> <p>Jerarquía de pruebas Duración: 00:15 LM: Actividad del tipo Lección Magistral</p> <p>Epics, historias de usuario y slicing Duración: 00:15 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Repositorios de código y buenas prácticas Duración: 00:45 LM: Actividad del tipo Lección Magistral</p>			<p>Realización de tareas con GitHub TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva No presencial Duración: 02:00</p>
2	<p>Introducción a los sistemas de build Duración: 00:10 LM: Actividad del tipo Lección Magistral</p> <p>Project build management: Maven Duración: 00:30 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Integración continua: Jenkins Duración: 00:20 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Introducción a AWS Duración: 00:30 PL: Actividad del tipo Prácticas de Laboratorio</p>			<p>Definición de una máquina EC2 en AWS. Instalar Jenkins y crear un proyecto de integración continua en Maven. TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva No presencial Duración: 04:00</p>
3	<p>Pruebas de unidad automatizadas: xUnit Duración: 00:20 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Pruebas de unidad automatizadas: Buenas prácticas Duración: 00:25 LM: Actividad del tipo Lección Magistral</p> <p>Pruebas de unidad en lenguajes sin reflexión: El caso de BoostTest/gTest Duración: 00:15 LM: Actividad del tipo Lección Magistral</p> <p>Statuschecks</p>			<p>Diseño, implementación y ejecución de pruebas automatizadas. Merge automatizado con statuschecks. Generación de reportes de calidad en Jenkins. TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva No presencial Duración: 04:00</p>

	<p>Duración: 00:30 PL: Actividad del tipo Prácticas de Laboratorio</p>			
4	<p>Otros sistemas de build: make, ant, gradle Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Pipelines/workflows Duración: 00:30 PL: Actividad del tipo Prácticas de Laboratorio</p>			
5	<p>Otros sistemas de build: cMake, Bazel Duración: 01:30 PL: Actividad del tipo Prácticas de Laboratorio</p>			<p>Definir un proyecto de integración continua en otro lenguaje de build TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva No presencial Duración: 01:00</p>
6	<p>Contratos Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Pruebas de integración Duración: 00:30 LM: Actividad del tipo Lección Magistral</p>			<p>Definición de los contratos y casos de prueba equivalentes para una clase TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva No presencial Duración: 01:00</p>
7	<p>Otros sistemas de integración continua: TravisCI, Bamboo, GitHub, CodeDeploy Duración: 00:40 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Despliegue en AWS Duración: 00:20 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Despliegue en AWS: Dockerización Duración: 00:30 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Control individual Duración: 02:00 OT: Otras actividades formativas / Evaluación</p>			<p>Control individual EP: Técnica del tipo Examen de Prácticas Evaluación Progresiva Presencial Duración: 02:00</p>
8	<p>Test doubles Duración: 00:15 LM: Actividad del tipo Lección Magistral</p> <p>Fakes, stubs, spies and mocks Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Mocking en languages sin reflexión: El caso de C++/gMock Duración: 00:15 LM: Actividad del tipo Lección Magistral</p>			<p>Diseño, implementación y ejecución de pruebas con mocking TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva No presencial Duración: 02:00</p>

9	<p>Diseño testable Duración: 01:30 PL: Actividad del tipo Prácticas de Laboratorio</p>			<p>Modificación de un proyecto para conseguir que sea testable TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva No presencial Duración: 02:00</p>
10	<p>Calidad del código. Refactorización. Duración: 01:30 PL: Actividad del tipo Prácticas de Laboratorio</p>			
11	<p>Diseño de casos de prueba: Introducción Duración: 00:30 LM: Actividad del tipo Lección Magistral</p> <p>Diseño de casos de prueba: Técnicas estructurales. Herramientas de cobertura. Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio</p>			<p>Diseño, implementación y ejecución de pruebas estructurales TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva No presencial Duración: 03:00</p>
12	<p>Diseño de casos de prueba: Técnicas funcionales Duración: 01:30 PL: Actividad del tipo Prácticas de Laboratorio</p>			<p>Diseño, implementación y ejecución de pruebas funcionales TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva No presencial Duración: 03:00</p>
13	<p>Calidad de casos de prueba: Mutación Duración: 01:30 PL: Actividad del tipo Prácticas de Laboratorio</p>			<p>Aplicación de mutation testing a un conjunto de casos de prueba TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva No presencial Duración: 02:00</p>
14	<p>Pruebas de aceptación Duración: 00:30 LM: Actividad del tipo Lección Magistral</p> <p>Pruebas de sistema (sobre interfaz) automatizadas Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio</p>			<p>Diseño e implementación de pruebas de sistema (sobre interfaz) automatizadas. Ejecución en AWS. TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva No presencial Duración: 04:00</p>
15	<p>IA generativa para calidad Duración: 01:30 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Control individual Duración: 02:00 OT: Otras actividades formativas / Evaluación</p>			<p>Control individual EP: Técnica del tipo Examen de Prácticas Evaluación Progresiva Presencial Duración: 02:00</p>
16				<p>Examen EP: Técnica del tipo Examen de Prácticas Evaluación Global Presencial Duración: 03:00</p>

17				
----	--	--	--	--

Para el cálculo de los valores totales, se estima que por cada crédito ECTS el alumno dedicará dependiendo del plan de estudios, entre 26 y 27 horas de trabajo presencial y no presencial.

6. Actividades y criterios de evaluación

6.1. Actividades de evaluación de la asignatura

6.1.1. Evaluación (progresiva)

Sem.	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
1	Realización de tareas con GitHub	TG: Técnica del tipo Trabajo en Grupo	No Presencial	02:00	5%	3 / 10	CE4 CE8
2	Definición de una máquina EC2 en AWS. Instalar Jenkins y crear un proyecto de integración continua en Maven.	TG: Técnica del tipo Trabajo en Grupo	No Presencial	04:00	6%	3 / 10	CE1 CE4 CE8
3	Diseño, implementación y ejecución de pruebas automatizadas. Merge automatizado con statuschecks. Generación de reportes de calidad en Jenkins.	TG: Técnica del tipo Trabajo en Grupo	No Presencial	04:00	10%	3 / 10	CE1 CE4 CE8
5	Definir un proyecto de integración continua en otro lenguaje de build	TG: Técnica del tipo Trabajo en Grupo	No Presencial	01:00	4%	3 / 10	
6	Definición de los contratos y casos de prueba equivalentes para una clase	TG: Técnica del tipo Trabajo en Grupo	No Presencial	01:00	3%	3 / 10	CE4 CE8
7	Control individual	EP: Técnica del tipo Examen de Prácticas	Presencial	02:00	15%	3 / 10	CG6 CG10 CE1 CE4 CE8
8	Diseño, implementación y ejecución de pruebas con mocking	TG: Técnica del tipo Trabajo en Grupo	No Presencial	02:00	7%	3 / 10	CE4 CE8
9	Modificación de un proyecto para conseguir que sea testable	TG: Técnica del tipo Trabajo en Grupo	No Presencial	02:00	3%	3 / 10	CE4 CE8

11	Diseño, implementación y ejecución de pruebas estructurales	TG: Técnica del tipo Trabajo en Grupo	No Presencial	03:00	10%	3 / 10	CE4 CE8
12	Diseño, implementación y ejecución de pruebas funcionales	TG: Técnica del tipo Trabajo en Grupo	No Presencial	03:00	8%	3 / 10	CE1 CE4
13	Aplicación de mutation testing a un conjunto de casos de prueba	TG: Técnica del tipo Trabajo en Grupo	No Presencial	02:00	6%	3 / 10	CE4 CE8
14	Diseño e implementación de pruebas de sistema (sobre interfaz) automatizadas. Ejecución en AWS.	TG: Técnica del tipo Trabajo en Grupo	No Presencial	04:00	8%	3 / 10	CE4 CE8
15	Control individual	EP: Técnica del tipo Examen de Prácticas	Presencial	02:00	15%	3 / 10	CG6 CG10 CE1 CE4 CE8

6.1.2. Prueba evaluación global

Sem	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
16	Examen	EP: Técnica del tipo Examen de Prácticas	Presencial	03:00	100%	5 / 10	CG6 CG10 CE1 CE4 CE8

6.1.3. Evaluación convocatoria extraordinaria

Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
Examen final extraordinario	EP: Técnica del tipo Examen de Prácticas	Presencial	03:00	100%	5 / 10	CG6 CG10 CE1 CE4 CE8

6.2. Criterios de evaluación

Evaluación progresiva

- La calificación dependerá de (1) la calidad en la presentación del trabajo realizado y (2) el grado de corrección del ejercicio.
- Todos los ejercicios prácticos son recuperables mediante re-entrega. No se penaliza la re-entrega,
- Los controles individuales son recuperables mediante un examen que coincidirá en fecha con la evaluación global de Enero.
- La calificación final se calculará como una media ponderada en función de los pesos indicados para las actividades de evaluación.

Evaluación global (Enero)

- Los alumnos realizarán un examen final que comprenderá todos los contenidos (teóricos y prácticos de la asignatura). Los materiales para su preparación estarán disponibles en Moodle.

Evaluación global (Julio)

- Se aplican las mismas normas que en la Evaluación global (Enero).

7. Recursos didácticos

7.1. Recursos didácticos de la asignatura

Nombre	Tipo	Observaciones
Moodle	Recursos web	Todos los materiales de la asignatura estarán disponibles en Moodle.

8. Otra información

8.1. Otra información sobre la asignatura