

Facultad de Ciencias

## GUÍA DOCENTE DE LA ASIGNATURA

G659 - Ingeniería del Software II

Grado en Ingeniería Informática  
Obligatoria. Curso 3

Curso Académico 2022-2023

### 1. DATOS IDENTIFICATIVOS

Título/s	Grado en Ingeniería Informática		Tipología v Curso	Obligatoria. Curso 3
Centro	Facultad de Ciencias			
Módulo / materia	MATERIA INGENIERÍA DEL SOFTWARE Y SISTEMAS DE INFORMACIÓN MÓDULO OBLIGATORIO			
Código y denominación	G659 - Ingeniería del Software II			
Créditos ECTS	6	Cuatrimestre	Cuatrimestral (2)	
Web	<a href="https://moodle.unican.es">https://moodle.unican.es</a>			
Idioma de impartición	Español	English friendly	No	Forma de impartición Presencial

Departamento	DPTO. INGENIERÍA INFORMÁTICA Y ELECTRÓNICA			
Profesor responsable	PATRICIA LOPEZ MARTINEZ			
E-mail	<a href="mailto:patricia.lopez@unican.es">patricia.lopez@unican.es</a>			
Número despacho	Facultad de Ciencias. Planta: + 3. DESPACHO PROFESOR (3051)			
Otros profesores	JUAN MARIA RIVAS CONCEPCION			

### 2. CONOCIMIENTOS PREVIOS

Esta asignatura es la continuación natural de la asignatura Ingeniería del Software I, por lo que se requiere haberla aprobado, o al menos, cursado.

Se supone, además, que el alumno posee los conocimientos sobre programación orientada a objetos impartidos en las asignaturas Métodos de Programación y Estructuras de Datos, y sobre desarrollo de interfaces gráficas impartidos en la asignatura Interacción Persona-Computador.

### 3. COMPETENCIAS GENÉRICAS Y ESPECÍFICAS DEL PLAN DE ESTUDIOS TRABAJADAS

Competencias Genéricas
(Conocimiento) Poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio.
(Aplicación) Aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio.
(Aprendizaje) Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.
Capacidad de análisis, síntesis y evaluación.
Capacidad de resolución de problemas aplicando técnicas de ingeniería.
Capacidad de trabajo en equipo.
Aprendizaje autónomo.
Tener motivación por la calidad.
Creatividad.
Competencias Específicas
Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.
Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.
Conocimiento, administración y mantenimiento de sistemas, servicios y aplicaciones informáticas.
Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software.

#### 3.1 RESULTADOS DE APRENDIZAJE

- Conocer los principios, fundamentos y buenas prácticas de la verificación y validación del software.
- Saber usar herramientas que dan soporte al desarrollo de sistemas software.
- Implementar y mantener aplicaciones, de acuerdo a las actividades de análisis y diseño previamente realizadas.
- Conocer y comprender el concepto, los principios y mecanismos de control de la calidad del software.
- Conocer los fundamentos, principios y técnicas de la gestión de proyectos software.

#### 4. OBJETIVOS

Presentar a los estudiantes los conocimientos básicos sobre verificación y validación de sistemas software.
Explicar a los estudiantes cómo aplicar un proceso de pruebas unitarias de sistemas software de forma metodológica.
Presentar a los estudiantes conocimientos y destrezas básicas sobre gestión del mantenimiento del software.
Hacer entender a los estudiantes la importancia de la gestión de la calidad en los procesos de desarrollo y en los productos software.
Presentar a los estudiantes conocimientos básicos sobre gestión de la configuración software y enseñarles a manejar herramientas para el control de versiones.
Presentar a los estudiantes los conocimientos y destrezas básicas sobre gestión de proyectos software.

5. MODALIDADES ORGANIZATIVAS Y MÉTODOS DOCENTES	
ACTIVIDADES	HORAS DE LA ASIGNATURA
<b>ACTIVIDADES PRESENCIALES</b>	
HORAS DE CLASE (A)	
- Teoría (TE)	20
- Prácticas en Aula (PA)	10
- Prácticas de Laboratorio Experimental(PLE)	
- Prácticas de Laboratorio en Ordenador (PLO)	30
- Prácticas Clínicas (CL)	
Subtotal horas de clase	60
<b>ACTIVIDADES DE SEGUIMIENTO (B)</b>	
- Tutorías (TU)	4
- Evaluación (EV)	8
Subtotal actividades de seguimiento	12
<b>Total actividades presenciales (A+B)</b>	<b>72</b>
<b>ACTIVIDADES NO PRESENCIALES</b>	
Trabajo en grupo (TG)	20
Trabajo autónomo (TA)	58
Tutorías No Presenciales (TU-NP)	
Evaluación No Presencial (EV-NP)	
<b>Total actividades no presenciales</b>	<b>78</b>
<b>HORAS TOTALES</b>	<b>150</b>

6. ORGANIZACIÓN DOCENTE													
CONTENIDOS		TE	PA	PLE	PLO	CL	TU	EV	TG	TA	TU-NP	EV-NP	Semana
1	GESTIÓN DE LA CONFIGURACIÓN DE SISTEMAS SOFTWARE - Terminología, conceptos básicos y áreas de la GC. - Herramientas de control de versiones. - Herramientas de construcción automatizada de software.	4,00	1,00	0,00	6,00	0,00	0,50	0,50	2,00	8,00	0,00	0,00	1-2
2	CONSTRUCCIÓN DE SISTEMAS SOFTWARE - Fase de construcción dentro del ciclo de vida del software. - Conceptos y principios básicos de la construcción de software.	2,00	1,00	0,00	4,00	0,00	0,50	2,00	5,00	10,00	0,00	0,00	3-4
3	PRUEBA DE SISTEMAS SOFTWARE - Fase de Verificación y Validación dentro del ciclo de vida del software. - Niveles y tipos de pruebas. - Técnicas de diseño de casos de prueba para pruebas unitarias. - Frameworks para la implementación de pruebas unitarias.	6,00	4,00	0,00	6,00	0,00	1,00	2,00	5,00	12,00	0,00	0,00	5-8
4	MANTENIMIENTO DE SISTEMAS SOFTWARE - Fase de mantenimiento dentro del ciclo de vida del software. - Tipos de mantenimiento. - Gestión del mantenimiento. - Técnicas y herramientas de soporte al mantenimiento.	2,00	1,00	0,00	4,00	0,00	0,50	1,00	3,00	8,00	0,00	0,00	9-10
5	CALIDAD DE SISTEMAS Y PROCESOS SOFTWARE - Concepto y factores de calidad del software. - Calidad de producto y calidad del proceso.	2,00	1,00	0,00	4,00	0,00	0,50	1,00	2,00	8,00	0,00	0,00	11-12
6	GESTIÓN DE PROYECTOS - Introducción a la gestión de proyectos software. - Integración y alcance. - Gestión de costes. - Gestión de tiempo. - Gestión de riesgos.	4,00	2,00	0,00	6,00	0,00	1,00	1,50	3,00	12,00	0,00	0,00	13-15
<b>TOTAL DE HORAS</b>		<b>20,00</b>	<b>10,00</b>	<b>0,00</b>	<b>30,00</b>	<b>0,00</b>	<b>4,00</b>	<b>8,00</b>	<b>20,00</b>	<b>58,00</b>	<b>0,00</b>	<b>0,00</b>	

Esta organización tiene carácter orientativo.

TE	Horas de teoría
PA	Horas de prácticas en aula
PLE	Horas de prácticas de laboratorio experimental
PLO	Horas de prácticas de laboratorio en ordenador
CL	Horas de prácticas clínicas
TU	Horas de tutoría
EV	Horas de evaluación
TG	Horas de trabajo en grupo
TA	Horas de trabajo autónomo
TU-NP	Tutorías No Presenciales
EV-NP	Evaluación No Presencial

## 7. MÉTODOS DE LA EVALUACIÓN

Descripción	Tipología	Eval. Final	Recuper.	%
Examen final	Examen escrito	Sí	Sí	40,00
Calif. mínima	4,50			
Duración	3 horas.			
Fecha realización	En la fecha indicada en el periodo ordinario de exámenes.			
Condiciones recuperación	Examen similar en la convocatoria extraordinaria.			
Observaciones	<p>El examen constará de un conjunto de cuestiones y/o ejercicios cortos.</p> <p>En la realización del examen se permite el uso de material escrito (apuntes, ejercicios, etc.) pero en ningún caso se permitirá la utilización de dispositivos electrónicos.</p>			
Entrega de prácticas	Trabajo	No	Sí	60,00
Calif. mínima	4,50			
Duración	Cada práctica tendrá una duración de entre una y tres sesiones de laboratorio.			
Fecha realización	A lo largo del cuatrimestre.			
Condiciones recuperación	Entrega y presentación de prácticas similares en convocatoria extraordinaria.			
Observaciones	<p>Los alumnos deberán realizar una serie de prácticas con ordenador donde se apliquen los conceptos expuestos en las clases teóricas.</p> <p>La calificación final se obtendrá como la media ponderada de las prácticas. El porcentaje de ponderación de cada práctica se establecerá al inicio del curso. Las prácticas entregadas fuera de fecha recibirán una calificación de 0.</p> <p>La recuperación de las prácticas consistirá en la entrega y presentación al profesor, en la convocatoria extraordinaria, de prácticas similares a las no entregadas o que obtuvieron una calificación menor a 5 en la convocatoria ordinaria.</p> <p>En el caso de alumnos que realicen entregas sin haber asistido asiduamente a las sesiones de prácticas en el laboratorio, además de la entrega a través de Moodle, deberán realizar una presentación de las prácticas al profesor, sujeta a preguntas, que sirva para confirmar la autoría de las mismas.</p>			
<b>TOTAL</b>				<b>100,00</b>
Observaciones				

De acuerdo a la normativa de la UC, la calificación de aquellos alumnos que sólo aprueben una de las partes (examen o entrega de prácticas) en la convocatoria ordinaria se calculará como el mínimo de 4,9 y la nota media obtenida.

En la convocatoria extraordinaria estos alumnos deberán recuperar únicamente aquella parte (o partes) en la que no se hubiera obtenido la calificación mínima en convocatoria ordinaria.

En el caso de la entrega de prácticas en convocatoria extraordinaria, estos alumnos, además de la entrega a través de Moodle, deberán realizar una presentación de las prácticas al profesor, sujeta a preguntas, que sirva para confirmar la autoría de las mismas.

A lo largo del curso se propondrán un conjunto de ejercicios que podrán ser entregados de manera voluntaria, pudiendo suponer hasta 1 punto de incremento en la nota final de la asignatura (siempre que la calificación de la asignatura sea mayor o igual a 5).

Todas las actividades de evaluación indicadas son adaptables a un escenario de evaluación no presencial. En caso de producirse este escenario, se indicaría a los alumnos las instrucciones concretas de adaptación de cada actividad (mecanismos de entrega o de realización de la prueba, plataformas de soporte, etc.) con la suficiente antelación.

#### Criterios de evaluación para estudiantes a tiempo parcial

Los alumnos matriculados a tiempo parcial se registrarán por el mismo sistema que los alumnos matriculados a tiempo completo, estando todas las actividades evaluables adaptadas a un régimen de asistencia semipresencial, a excepción de las pruebas escritas. No obstante, al celebrarse dichas pruebas en días bien identificados desde el comienzo del cuatrimestre, es de esperar que el alumno en régimen semipresencial pueda acudir a dichas pruebas.

En el caso de las prácticas de laboratorio, además de la entrega a través de Moodle, estos alumnos deberán realizar una presentación de las prácticas al profesor, sujeta a preguntas, que sirva para confirmar la autoría de las mismas.

## 8. BIBLIOGRAFÍA Y MATERIALES DIDÁCTICOS

### BÁSICA

Ian Sommerville, "Ingeniería del Software", 9ª Edición, Addison-Wesley, 2011.

Glenford J. Myers, Corey Sandler and Tom Badgett, "The Art of Software Testing", 3ª Edición, Wiley, 2011.

M. Piattini, F. García, I. García-Rodríguez de Guzmán, F. Pino, "Calidad de Sistemas de Información", 4ª Ed., Ra-Ma, 2018.

### Complementaria

IEEE Computer Society, "SWEBOOK - Guide to the Software Engineering Body of Knowledge v3", 2014.

Project Management Institute (PMI), "PMBOK - Project Management Body of Knowledge", 6ª edición, 2017.

Jon Loeliger and Matthew McCullough, "Version Control with Git", O'Reilly Media, 2012.

Martin Fowler, "Refactoring: improving the design of existing code", Addison Wesley, 2009.

9. SOFTWARE				
PROGRAMA / APLICACIÓN	CENTRO	PLANTA	SALA	HORARIO
Magic Draw Personal Edition 18.0	Facultad de Ciencias			
Eclipse for Java Developers	Facultad de Ciencias			
Apache Maven	Facultad de Ciencias			
Git	Facultad de Ciencias			
SonarQube	Facultad de Ciencias			
ProjectLibre	Facultad de Ciencias			

10. COMPETENCIAS LINGÜÍSTICAS	
<input checked="" type="checkbox"/> Comprensión escrita	<input type="checkbox"/> Comprensión oral
<input type="checkbox"/> Expresión escrita	<input type="checkbox"/> Expresión oral
<input type="checkbox"/> Asignatura íntegramente desarrollada en inglés	
<b>Observaciones</b>	