Facultad de Ciencias

GUIA DOCENTE DE LA ASIGNATURA

G667 - Servicios Software

Grado en Ingeniería Informática
Optativa. Curso 4

Curso Académico 2026 - 2027

1. DATOS IDENTIFICATIVOS					
Título/s	Grado en Ingeniería Informática			Tipología y curso	Optativa. Curso 4
Centro	Facultad de Ciencias				
Módulo / materia	MATERIA INGENIERÍA DEL SOFTWARE MENCIÓN EN INGENIERÍA DEL SOFTWARE				
Código y denominación	G667 - Servicios Software				
Créditos ECTS	6	Cuatrimestre		Cuatrimestral (2)	
Web	http://moodle.unican.es				
Idioma de impartición	Español	English Friendly	No	Forma de impartición	Presencial

Departamento	DPTO. INGENIERÍA INFORMÁTICA Y ELECTRÓNICA
Profesor responsable	JUAN RAMON SANTANA MARTINEZ
E-mail	juanramon.santana@unican.es
Número despacho	Facultad de Ciencias. Planta: + 1. DESPACHO PROFEROR (1070)
Otros profesores	

2. CONOCIMIENTOS PREVIOS

El alumno debe haber cursado las siguientes asignaturas, que cubren aspectos necesarios para el adecuado desarrollo de esta asignatura (los aspectos concretos aparecen indicados entre paréntesis):
- Procesos de la Ingeniería Software (Jakarta EE y pruebas de software).
- Ingeniería del Software I (Ciclo de vida del software y UML).
- Ingeniería del Software II (Maven y pruebas de software)

Se supone, además, que el alumno ha superado todas las asignaturas básicas sobre programación, bases de datos y sistemas de información.

3. COMPETENCIAS GENÉRICAS Y ESPECÍFICAS DEL PLAN DE ESTUDIOS TRABAJADAS
Competencias Genéricas
Capacidad de análisis, síntesis y evaluación.
Capacidad de resolución de problemas aplicando técnicas de ingeniería.
Capacidad para argumentar y justificar lógicamente las decisiones tomadas y las opiniones.
Capacidad de trabajo en equipo.
Capacidad de relación interpersonal.
Aprendizaje autónomo.
Creatividad.
Capacidad de organización y planificación.
Tener motivación por la calidad.
Competencias Específicas
Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la Ingeniería del Software.
Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.
Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales.
Capacidad para diseñar soluciones apropiadas en uno o más dominios de aplicación utilizando métodos de la ingeniería del software que integren aspectos éticos, sociales, legales y económicos.
Competencias Básicas
Que los estudiantes hayan demostrado 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.
Que los estudiantes sepan 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.
Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.
Competencias Transversales

3.1 RESULTADOS DE APRENDIZAJE
Desarrollar aplicaciones software empleando una arquitectura orientada a servicios.
Utilizar entornos para el desarrollo de servicios web.
Saber reflejar en las diversas etapas del desarrollo de un sistema software los aspectos relacionados con la seguridad.

4. OBJETIVOS
Presentar a los estudiantes el origen y los principios de la computación orientada a servicios y de las arquitecturas orientadas a servicios.
Presentar a los estudiantes los principios y las diferencias entre las principales tecnologías de servicios (e.g. WSDL/SOAP, REST, GraphQL).
Explicar a los estudicantes cómo desarrollar servicios y aplicaciones basadas en servicios aplicando tecnologías como WSDL/SOAP, REST o GraphQL.
Explicar a los estudiantes cómo gestionar aspectos relacionados con seguridad en el desarrollo de servicios.

5. MODALIDADES ORGANIZATIVAS Y MÉTODOS DOCENTES	
ACTIVIDADES	HORAS DE LA ASIGNATURA
ACTIVIDADES PRESENCIALES	
HORAS DE CLASE (A)	
- Teoría (TE)	15
- Prácticas en Aula (PA)	5
- Prácticas de Laboratorio Experimental (PLE)	
- Prácticas de Laboratorio en Ordenador (PLO)	40
- Prácticas Clínicas (CL)	
Subtotal horas de clase	60
ACTIVIDADES DE SEGUIMIENTO (B)	
- Tutorías (TU)	4
- Evaluacion (EV)	8
Subtotal actividades de seguimiento	12
Total actividades presenciales (A+B)	72
ACTIVIDADES NO PRESENCIALES	
Trabajo en grupo (TG)	20
Trabajo autónomo (TA)	58
Tutorías No Presenciales (TU-NP)	
Evaluacion No Presencial (EV-NP)	
Total actividades no presenciales	78
HORAS TOTALES	150

6. ORGANIZACIÓN DOCENTE

	CONTENIDOS	TE	PA	PLE	PLO	CL	TU	EV	TG	TA	TU-NP	EV-NP	Semana
1	TECNOLOGÍAS DE SOPORTE A LOS SERVICIOS WEB - Repaso de HTTP, URI, XML y JSON.	1,00	0,00	0,00	6,00	0,00	0,50	1,00	2,00	6,00	0,00	0,00	1-2
2	INTRODUCCIÓN A LA ORIENTACIÓN A SERVICIOS - Origen, terminología y conceptos básicos de la orientación a servicios. - Paradigma SOC y arquitecturas SOA.	1,00	0,00	0,00	0,00	0,00	0,25	0,50	1,00	4,00	0,00	0,00	2
3	DESPLIEGUE DE SERVICIOS EN LA NUBE - Uso de servicios cloud para el despliegue de VM en la nube - Despliegue de servicios a través de tecnologías como Docker	1,00	1,00	0,00	2,00	0,00	0,50	0,50	1,00	4,00	0,00	0,00	3
4	TECNOLOGÍAS DE SERVICIOS WEB Desarrollo de servicios, despliegue y pruebas de servicios web basados en las siguientes tecnologías como: - WSDL/SOAP - REST - GRAPHQL - gRPC	10,00	2,00	0,00	28,00	0,00	2,00	4,50	12,00	36,00	0,00	0,00	4-13
5	SEGURIDAD EN SERVICIOS WEB - Conceptos básicos de seguridad para servicios web - JWT, OAuth2.0	2,00	2,00	0,00	4,00	0,00	0,75	1,50	4,00	8,00	0,00	0,00	13-14
TOTAL DE HORAS		15,00	5,00	0,00	40,00	0,00	4,00	8,00	20,00	58,00	0,00	0,00	
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 escrito	Examen escrito	Sí	Sí	50,00

Calif. mínima	4,00
Duración	3 horas.
Fecha realización	Fecha asignada en la convocatoria ordinaria de exámenes.
Condiciones recuperación	Examen similar en la convocatoria extraordinaria.
Observaciones	El examen consistirá en una serie de cuestiones y ejercicios cortos.

Entrega de prácticas	Trabajo	No	Sí	50,00

Calif. mínima	4,50
Duración	Cada práctica tendrá una duración de entre dos y cuatro sesiones de laboratorio.
Fecha realización	A lo largo del cuatrimestre.
Condiciones recuperación	Entrega de prácticas similares en convocatoria extraordinaria.
Observaciones	Los alumnos realizarán un conjunto de prácticas de ordenador donde se apliquen los conceptos expuestos en las clases teóricas. La calificación final se obtendrá como la media ponderada de las prácticas entregadas. El porcentaje de ponderación de cada práctica se establecerá al inicio del curso. Las prácticas entregadas fuera de fecha recibirán una calificación de 0. 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. En el caso de alumnos que realicen la entrega 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.

TOTAL	100,00

Observaciones

De acuerdo a la normativa de la UC, la calificación de aquellos alumnos que sólo superen la calificación mínima de 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 sólo deberán recuperar aquella parte en la que no se alcanzó la calificación mínima en la 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.

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 regirán 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
Gustavo Alonso et al., "Web Services: Concepts, Architectures and Applications", Springer, 2004.
Roy T. Fielding and N.Richard, "Principled Design of the Modern Web Architecture", ACM Transactions on Internet Technology 2 (2): 115–150, 2002.
Leonard Richardson and Sam Ruby, "RESTful Web Services", O'Reilly, 2007.
Prisma, "The Fullstack Tutorial for GraphQL", https://www.howtographql.com/
Cloud Native Computing Foundation (CNCF), "gRPC docs", https://grpc.io/docs/
Docker, "Docker Docs", https://docs.docker.com/
Complementaria
Eclipse Foundation, "The Jakarta EE Tutorial".

9. SOFTWARE

PROGRAMA / APLICACIÓN	CENTRO	PLANTA	SALA	HORARIO
Eclipse IDE for Java Developers / Visual Studio Code				
Apache Maven				
Spring Boot				

10. COMPETENCIAS LINGÜÍSTICAS

☑ Comprensión escrita ☐ Comprensión oral

☐ Expresión escrita ☐ Expresión oral

☐ Asignatura íntegramente desarrollada en inglés

Observaciones