



DATOS IDENTIFICATIVOS

Programación concurrente y distribuida

Asignatura	Programación concurrente y distribuida			
Código	V05G300V01641			
Titulación	Grado en Ingeniería de Tecnologías de Telecomunicación			
Descriptores	Creditos ECTS	Seleccione	Curso	Cuatrimestre
	6	OP	3	2c
Lengua Impartición	Castellano			
Departamento	Ingeniería telemática			
Coordinador/a	García Duque, Jorge			
Profesorado	García Duque, Jorge			
Correo-e	jgd@det.uvigo.es			
Web	http://fatic.uvigo.es			
Descripción general	El objetivo que se persigue con esta asignatura es que el alumno conozca los fundamentos de la sincronización y comunicación entre procesos tanto en sistemas centralizados como distribuidos.			

Competencias

Código	
B3	CG3 Conocimiento de materias básicas y tecnologías que capaciten al alumnado para el aprendizaje de nuevos métodos y tecnologías, así como que le dote de una gran versatilidad para adaptarse a nuevas situaciones.
B4	CG4 Capacidad para resolver problemas con iniciativa, para la toma de decisiones, la creatividad, y para comunicar y transmitir conocimientos, habilidades y destrezas, comprendiendo la responsabilidad ética y profesional de la actividad del Ingeniero Técnico de Telecomunicación.
B9	CG9 Capacidad para trabajar en un grupo multidisciplinar y en un entorno multilingüe y de comunicar, tanto por escrito como de forma oral, conocimientos, procedimientos, resultados e ideas relacionadas con las telecomunicaciones y la electrónica.
C33	CE33/TEL7 Capacidad de programación de servicios y aplicaciones telemáticas, en red y distribuidas.
D2	CT2 Concebir la Ingeniería en un marco de desarrollo sostenible.
D3	CT3 Tomar conciencia de la necesidad de una formación y mejora continua de calidad, mostrando una actitud flexible, abierta y ética ante opiniones o situaciones diversas, en particular en materia de no discriminación por sexo, raza o religión, respeto a los derechos fundamentales, accesibilidad, etc.
D4	CT4 Favorecer el trabajo cooperativo, las capacidades de comunicación, organización, planificación y aceptación de responsabilidades en un ambiente de trabajo multilingüe y multidisciplinar, que favorezca la educación para la igualdad, para la paz y para el respeto de los derechos fundamentales.

Resultados de aprendizaje

Resultados previstos en la materia	Resultados de Formación y Aprendizaje	
Capacidad para diseñar y construir sistemas *concurrentes y distribuidos.	B4 B9	C33
Comprensión de los principales conceptos teóricos de los sistemas *concurrentes y distribuidos.	B3	D2 D3 D4
Conocimiento de las principales herramientas y entornos para el desarrollo de sistemas *concurrentes y distribuidos	B4 B9	C33
Adquisición de las competencias específicas del título (en función del proyecto realizado por el alumno en las prácticas).		

Contenidos

Tema	
Introducción a la Programación Concurrente	<input type="checkbox"/> Conceptos de concurrencia, paralelismo y multitarea. <input type="checkbox"/> Entrelazamiento de instrucciones atómicas. <input type="checkbox"/> Grafos de precedencia.
El problema de la sección crítica	<input type="checkbox"/> Definición del problema. <input type="checkbox"/> Espera Activa. <input type="checkbox"/> Inanición <input type="checkbox"/> Interbloqueo. <input type="checkbox"/> El algoritmo de Decker. <input type="checkbox"/> El algoritmo de Peterson
Herramientas de Sincronización	<input type="checkbox"/> Semáforos. <input type="checkbox"/> El problema del productor-consumidor. <input type="checkbox"/> El problema de los filósofos. <input type="checkbox"/> Monitores. <input type="checkbox"/> Variables de Condición. <input type="checkbox"/> El problema de los lectores-escritores.
Gestión de Interbloqueo	<input type="checkbox"/> Introducción y definición de interbloqueo. <input type="checkbox"/> Condiciones necesarias. <input type="checkbox"/> Estrategias de Prevención. <input type="checkbox"/> Estrategias de Evasión. <input type="checkbox"/> Detección y Recuperación
Comunicación entre procesos	<input type="checkbox"/> Paso de Mensajes. <input type="checkbox"/> Llamada a Procedimiento Remoto (RPC).
Programación Distribuida	<input type="checkbox"/> Introducción a los Sistemas Distribuidos. <input type="checkbox"/> Exclusión Mutua Distribuida: o Algoritmo Ricart-Agrawala. o Algoritmos de paso de testigo. <input type="checkbox"/> Consenso Distribuido: o Fallos de parada. o Fallos bizantinos.

Planificación

	Horas en clase	Horas fuera de clase	Horas totales
Talleres	5	30	35
Prácticas en aulas de informática	13	26	39
Lección magistral	20	46	66
Examen de preguntas objetivas	1	0	1
Práctica de laboratorio	1	0	1
Trabajo	2	6	8

*Los datos que aparecen en la tabla de planificación son de carácter orientativo, considerando la heterogeneidad de alumnado

Metodologías

	Descripción
Talleres	<p>Cada grupo de alumnos abordará el diseño e implementación de un proyecto software de complejidad media. Dicha tarea se realizará en diferentes pasos sucesivos, que serán discutidos y validados en cada una de las sesiones presenciales.</p> <p>Esta metodología de trabajo tiene como objetivo proporcionar una adecuada realimentación para, si es oportuno, mejorar las soluciones planteadas.</p> <p>Esta metodología está orientada a adquirir las competencias CG4 y CG9</p>
Prácticas en aulas de informática	<p>Los alumnos resolverán bajo la supervisión del profesorado los problemas prácticos que se planteen en cada sesión de laboratorio.</p> <p>Esta metodología está orientada a adquirir las competencias CE33/TEL7</p>
Lección magistral	<p>Exposición de las ideas, conceptos, técnicas y algoritmos de cada lección del temario.</p> <p>Esta metodología está orientada a adquirir las competencias CG3</p>

Atención personalizada

Metodologías	Descripción
Lección magistral	Mediante tutorías
Talleres	Parte de las sesiones se dedican a resolver cuestiones individuales con cada alumno mediante preguntas individualizadas tanto por parte del profesor como del alumno
Prácticas en aulas de informática	De manera completa para los alumnos que hacen las prácticas de manera individual, y mediante la resolución de cuestiones individuales con cada alumno mediante preguntas individualizadas tanto por parte del profesor como del alumno

Evaluación						
	Descripción	Calificación	Resultados de Formación y Aprendizaje			
Examen de preguntas objetivas	Prueba de contenidos teóricos de cada uno de los temas expuestos en las sesiones magistrales.	50	B3 B4	C33	D2	
Práctica de laboratorio	Evaluación del trabajo realizado en cada una de las sesiones de laboratorio	20	B3 B4	C33	D2 D3	
	Para la evaluación individualizada de cada alumno se realizarán cuestiones personalizadas en cada una de las sesiones.					
Trabajo	En la última sesión presencial del taller los alumnos entregarán y expondrán a sus compañeros el diseño y la solución suscitados para el sistema software objetivo del proyecto. Dicta solución será expuesta a debate entre los alumnos y los profesores.	30	B9	C33	D3 D4	
	Para la evaluación individualizada de cada alumno se realizarán cuestiones personalizadas en cada una de las sesiones.					

Otros comentarios sobre la Evaluación

La materia puede superarse mediante Evaluación Continua segundo los criterios que se indican más adelante, teniendo abierta la posibilidad de optar por la Evaluación No Continua en cualquier momento hasta el comienzo del examen final a celebrar el día fijado a tal efecto en el calendario oficial de la EET.

Todos aquellos alumnos que opten por la evaluación continua se considerarán presentados se se evalúan de la parte del trabajo en Talleres.

Evaluación Continua:

La nota final resultará de la suma de las notas correspondientes al tres componentes siguientes:

1. Cuatro pruebas de tipo Test para evaluar los contenidos impartidos en las clases magistrales. Cada prueba tendrá lugar en una de las sesiones magistrales, excepto la última que se realizará en una de las sesiones del Taller.

Puntuación: Hasta 1,25 puntos cada prueba.

2. Seis Pruebas Prácticas que se realizarán al finalizar cada una de las sesiones de laboratorio y que consistirán en la **validación de los resultados obtenidos durante la dicha sesión.

Puntuación: Hasta 1/3 puntos. cada prueba.

3. Presentación del Proyecto propuesto como trabajo en las sesiones del Taller.

Puntuación: Hasta 3 puntos.

Para aprobar la materia por Evaluación Continua se tendrán que dar el tres condiciones siguientes:

- (i) obtener una calificación igual o superior a 2 puntos en el conjunto de los tests.;
- (ii) calificación superior a 0 puntos en, por lo menos, cuatro del seis pruebas prácticas; y
- (iii) asistir a todas las sesiones presenciales de taller y obtener más de 0 puntos en la presentación del proyecto.

En caso de no cumplir alguna de dicha condición, la nota final del alumno será limitada a un máximo de 4 puntos.

Evaluación No Continua:

Mediante un examen sobre 10 puntos fijado en el calendario oficial de la EET.

Convocatoria de Fin de Curso y Convocatoria Extraordinaria (fin de grado):

Se registrá por el indicado para la evaluación No Continua.

Fuentes de información

Bibliografía Básica

M. Ben-Ari, **Principles of Concurrent And Distributed Programming**, Second Edition,

Bibliografía Complementaria

George Coulouris, Jean Dollimore, Tim Kindberg and Gordon Blair, **Distributed Systems Concepts and Design**, Fifth Edition,

William Stallings, **Operating Systems: Internals and Design Principles, 6/E**, Eight Edition,

Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, **Operating system concepts**, Ninth Edition,

Lea, Douglas, **Programación concurrente en Java : principios y patrones de diseño**, Second Edition,

Recomendaciones

Asignaturas que se recomienda cursar simultáneamente

Arquitecturas y servicios telemáticos/V05G300V01645

Sistemas de información/V05G300V01644

Asignaturas que se recomienda haber cursado previamente

Informática: Arquitectura de ordenadores/V05G300V01103

Programación I/V05G300V01205

Programación II/V05G300V01302

Sistemas operativos/V05G300V01541
