

Curso Académico: ( 2024 / 2025 )

Fecha de revisión: 21-02-2025

Departamento asignado a la asignatura: Departamento de Informática

Coordinador/a: GARCIA CARBALLEIRA, FELIX

Tipo: Obligatoria Créditos ECTS : 6.0

Curso : 3 Cuatrimestre : 2

## REQUISITOS (ASIGNATURAS O MATERIAS CUYO CONOCIMIENTO SE PRESUPONE)

Sistemas Operativos (Curso 2 - cuatrimestre 2)  
Redes de ordenadores (Curso 3 - cuatrimestre 1)

## RESULTADOS DEL PROCESO DE FORMACIÓN Y APRENDIZAJE

RA2.2: La capacidad de identificar, formular y resolver problemas de ingeniería en su especialidad; elegir y aplicar de forma adecuada métodos analíticos, de cálculo y experimentales ya establecidos; reconocer la importancia de las restricciones sociales, de salud y seguridad, ambientales, económicas e industriales.

RA3.1: Capacidad para proyectar, diseñar y desarrollar productos complejos (piezas, componentes, productos acabados, etc.), procesos y sistemas de su especialidad, que cumplan con los requisitos establecidos, incluyendo tener conciencia de los aspectos sociales, de salud y seguridad, ambientales, económicos e industriales; así como seleccionar y aplicar métodos de proyecto apropiados.

RA3.2: Capacidad de proyecto utilizando algún conocimiento de vanguardia de su especialidad de ingeniería.

RA7.1: Capacidad para comunicar eficazmente información, ideas, problemas y soluciones en el ámbito de ingeniería y con la sociedad en general.

CG7: Ser capaz de exponer y discutir propuestas en el trabajo en equipo, demostrando habilidades personales y sociales que le permitan asumir responsabilidades distintas dentro de los mismos.

CGO6: Capacidad para concebir y desarrollar sistemas o arquitecturas informáticas centralizadas o distribuidas integrando hardware, software y redes de acuerdo con los conocimientos adquiridos.

CECRI11: Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.

CECRI13: Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.

CECRI14: Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.

CTE6: Capacidad de analizar y evaluar arquitecturas de computadores, incluyendo plataformas paralelas y distribuidas, así como desarrollar y optimizar software de para las mismas.

CTE7: Capacidad de diseñar e implementar software de sistema y de comunicaciones.

CTE8: Capacidad de analizar, evaluar y seleccionar las plataformas hardware y software más adecuadas para el soporte de aplicaciones empujadas y de tiempo real.

## OBJETIVOS

El objetivo de este curso es describir los principales conceptos necesarios para diseñar y desarrollar aplicaciones y sistemas distribuidos.

## DESCRIPCIÓN DE CONTENIDOS: PROGRAMA

Los descriptores asociados con la asignatura son:

Comunicación entre procesos; Middleware; RPC; almacenamiento distribuido; aplicaciones distribuidas; sincronización; tolerancia a fallos.

Temario:  
Temario:

1. Introducción y conceptos básicos
  - Conceptos básicos
  - Redes e interconexión
  - Ventajas de los sistemas distribuidos
  - Paradigmas de computación distribuida
  - Diseño de sistemas distribuidos
2. Comunicación y sincronización entre procesos
  - Mecanismos de comunicación en sistemas de memoria compartida
  - Mecanismos de comunicación en sistemas de paso de mensajes
  - Servicios POSIX de comunicación y sincronización
  - Threads en Python
3. Paso de mensajes y aplicaciones cliente-servidor
  - Modelo de comunicación mediante paso de mensajes
  - Aspectos de diseño
  - Colas de mensajes POSIX
  - Aplicaciones cliente-servidor
  - Diseño de servidores concurrentes
4. Comunicación con sockets
  - Modelo de comunicación con sockets
  - API de sockets en POSIX
  - API de sockets en Python
  - Guía de diseño de aplicaciones cliente-servidor con sockets
5. Llamadas a procedimientos remotos
  - Funcionamiento de las RPC
  - Lenguaje de definición de interfaces
  - Aplanamiento y transferencia de mensajes
  - ONC-RPC
  - Llamadas a procedimientos remotos en Python
6. Servicios web
  - Protocolo HTTP
  - SOAP
  - Desarrollo de servicios web
7. Servicios distribuidos
  - Servicio de nombres
  - Sincronización en sistemas distribuidos
  - Relojes físicos y lógicos
  - Exclusión mutua distribuida
  - Algoritmos de elección
  - Comunicación multicast
8. Sistemas de almacenamiento distribuidos
  - Estructura de un sistema de ficheros distribuido
  - Servicio de ficheros y directorios
  - Implementación de servicios de ficheros distribuidos
  - Ejemplos de sistemas de fichero distribuidos
  - Sistemas de ficheros de discos compartidos
  - Sistemas de ficheros paralelos
  - Redes de almacenamiento
9. Tolerancia a fallos en sistemas distribuidos
  - Conceptos básicos sobre tolerancia a fallos
  - Tolerancia a fallos software
  - Detectores de fallos
  - Replicación

- Protocolos de consenso

## ACTIVIDADES FORMATIVAS, METODOLOGÍA A UTILIZAR Y RÉGIMEN DE TUTORÍAS

- \* Clases magistrales: 1 ECTS. Tienen por objetivo alcanzar las competencias específicas cognitivas de la asignatura, así como las competencias transversales capacidad de análisis y abstracción.
- \* Clases prácticas: 1 ECTS. Tienen por objetivo iniciar el desarrollo de las competencias específicas instrumentales, así como las competencias transversales resolución de problemas y aplicación de conocimientos.
- \* Ejercicios de evaluación continua: 1,5 ECTS. Iniciados durante las clases prácticas y terminados fuera de las mismas, tienen por objetivo completar el desarrollo de las competencias específicas instrumentales e iniciar el desarrollo de las competencias específicas actitudinales, así como las competencias transversales resolución de problemas y aplicación de conocimientos.
- \* Trabajos prácticos: 2 ECTS. Desarrollados sin presencia del profesor, tienen por objetivo completar e integrar el desarrollo de todas las competencias específicas y transversales, en la resolución de dos casos prácticos donde queden bien documentados el planteamiento del problema, la elección del método de resolución, los resultados obtenidos y la interpretación de los mismos.
- \* Tutorías: TUTORÍAS. Asistencia individualizada (tutorías individuales) o en grupo (tutorías colectivas) a los estudiantes por parte del profesor.
- \* Examen final: 0,5 ECTS. Tiene por objeto incidir y complementar en el desarrollo de las capacidades específicas cognitivas y procedimentales. Refleja especialmente el aprovechamiento de las clases magistrales.

## SISTEMA DE EVALUACIÓN

<b>Peso porcentual del Examen Final:</b>	20
<b>Peso porcentual del resto de la evaluación:</b>	80

La evaluación será continua y consta de las siguientes partes:

1. Evaluación continua (80%)
  - Ejercicios y trabajos: 40%
  - Proyecto final de programación: 40%
2. Examen final (20%)

La nota mínima en el examen final será de 4.  
La nota media mínima del proyecto final será de 4.  
La notas media mínima de todos los ejercicios y trabajos será de 4.

Se considera que un alumno sigue el proceso de evaluación continua cuando:

- La nota media de todos los ejercicios y trabajos es de 4.
- La nota media del proyecto de programación final es de 4.

El examen final de la convocatoria extraordinaria incluirá los conceptos teóricos y prácticos de la asignatura.

En esta asignatura los y las estudiantes no deben utilizar herramientas de inteligencia artificial para la realización de los trabajos o ejercicios propuestos por el profesor o la profesora. En el supuesto de que la utilización de IA por el/la estudiante dé lugar a fraude académico por falsear los resultados de un examen o trabajo requerido para acreditar el rendimiento académico, se aplicará lo dispuesto en el Reglamento de la Universidad Carlos III de Madrid de desarrollo parcial de la Ley 3/2022, de 24 de febrero, de convivencia universitaria.

## BIBLIOGRAFÍA BÁSICA

- G. Coulouris, J. Dollimore, T. Kindberg, G. Blair Distributed Systems, Concepts and design. 5ª edition. 2011, Addison-Wesley.

- Maarten van Steen, Andrew S. Tanenbaum Distributed Systems, 4ed., Maarten van Steen, 2024

#### BIBLIOGRAFÍA COMPLEMENTARIA

- B. A. Forouzan Transmisión de datos y redes de comunicaciones. Cuarta edición, McGraw Hill.

- F. García, J. Carretero, A. Calderón, J. Fernández, J. M. Pérez Problemas resueltos de programación en C, Thomson.

- Jesus Carretero Pérez, Félix García Carballeira, Fernando Pérez Costoya Sistemas Operativos. Una Visión Aplicada: Tercera edición. , Amazon, 2021

- L. H. Etzkorn Introduction to Middleware: Web Services, Object Components, and Cloud Computing, CRC Press, 2017