

FICHA TÉCNICA DE LA ASIGNATURA

Datos de la asignatura	
Nombre	Programación de Aplicaciones Telemáticas
Código	ASI30
Titulación	Grado en Ingeniería Telemática
Curso	3º
Cuatrimestre	2º
Créditos ECTS	7,5 ECTS
Carácter	Obligatorio
Departamento	Telemática y Computación
Área	Ingeniería Telemática
Universidad	Universidad Pontificia Comillas
Horario	
Profesores	David Contreras Bárcena
Descriptor	

Datos del profesorado	
Profesor	
Nombre	David Contreras Bárcena
Departamento	Telemática y Computación
Área	Ingeniería Telemática
Despacho	D-410
e-mail	davidcb@dsi.icaupcomillas.es
Teléfono	91 542 28 00 ext. 4235
Horario de Tutorías	Lunes de 12:00 a 14:00

DATOS ESPECÍFICOS DE LA ASIGNATURA

Contextualización de la asignatura

Aportación al perfil profesional de la titulación

El objetivo principal de la asignatura es proporcionar al alumno la capacidad de desarrollar aplicaciones web mediante el lenguaje de programación Java.

Alcanzar estos objetivos implica conocer los lenguajes básicos de programación del lado del cliente, como son HTML, CSS y Javascript. Además se deberá dominar criterios de diseño para la implementación óptima de la capa de lógica de negocio.

Al finalizar el curso el alumno adquirirá los conocimientos suficientes para desarrollar aplicaciones de acceso a base de datos en entornos web y a desarrollar interfaces para dispositivos móviles que interactúen con estas aplicaciones, de forma nativa y mediante el acceso HTTP.

Prerrequisitos

Ninguno

Competencias - Objetivos

Competencias Genéricas del título-curso

CGT3. Conocimiento de materias básicas y tecnologías, que le capacite para el aprendizaje de nuevos métodos y tecnologías, así como que le dote de una gran versatilidad para adaptarse a nuevas situaciones.

CGT4. Capacidad de resolver problemas con iniciativa, toma de decisiones, creatividad, y de comunicar y transmitir conocimientos, habilidades y destrezas, comprendiendo la responsabilidad ética y profesional de la actividad del ingeniero técnico de telecomunicación.

CRT7. Conocimiento y utilización de los fundamentos de la programación en redes, sistemas y servicios de telecomunicación.

Competencias de tecnología Específica Telemática

CETM4. Capacidad de describir, programar, validar y optimizar protocolos e interfaces de comunicación en los diferentes niveles de una arquitectura de redes.

Resultados de Aprendizaje¹

Aplicar las técnicas de modelado UML con Java para el análisis y diseño orientado a objetos de aplicaciones telemáticas.

RA1. Diseña y programa soluciones web basadas en el patrón MVC enfocadas a resolver problemas de software reales.

RA2. Adapta los resultados del análisis y diseño UML (diagramas de clases, casos de uso y secuencia) a soluciones web.

Diseñar, desarrollar y validar aplicaciones empresariales con Java EE.

RA3. Implementa interfaces web usables mediante HTML y CSS.

RA4. Desarrolla páginas web amigables con funcionalidad Javascript.

RA5. Realiza programas basados en JSP y Servlet entendiendo la diferencia entre ambos.

RA6. Conoce y sabe aplicar los lenguajes EL y JSTL.

Utilización de tecnologías y frameworks de desarrollo para aplicaciones web.

RA7. Comprende el funcionamiento del protocolo HTTP.

RA8. Conoce las diferencias existentes entre peticiones GET y POST.

RA9. Gestiona la información persistente en el lado del cliente: *cookies*.

RA10. Conoce las técnicas y principios que permitan mejorar el rendimiento de una aplicación web en el lado del cliente y servidor.

RA11. Diferencia los tres contextos que existen en el lado del servidor: *request*, *session* y *application*.

RA12. Sabe utilizar cuando corresponda los elementos que permiten gestionar el flujo de navegación en el servidor: *forward* vs *redirect*.

RA13. Realiza desarrollos basados en peticiones asíncronas basadas en XML y JSON.

RA14. Conoce y utiliza *frameworks* y *librerías* de ayuda al desarrollo de aplicaciones web en el lado del cliente y servidor.

RA15. Sabe en qué consisten y replica ataques de seguridad del tipo *Form Tampering*, *SQL Injection*, *XSS* y *CSRF*.

RA16. Conoce los principios básicos de la metodología de seguridad OWASP.

Diseñar y desarrollar aplicaciones para dispositivos móviles.

RA17. Desarrolla aplicaciones web basadas en los principios *Mobile First* y *1-1-3*.

RA18. Desarrolla aplicaciones móviles nativas.

¹ Los resultados de aprendizaje son indicadores de las competencias que nos permiten evaluar el grado de dominio que poseen los alumnos. Las competencias suelen ser más generales y abstractas. Los R.A. son indicadores observables de la competencia

BLOQUES TEMÁTICOS Y CONTENIDOS

Contenidos – Bloques Temáticos

Tema 1: JAVA SE AVANZADO

- 1.1 JDBC: acceso a base de datos.
- 1.2 Threads: programación concurrente.
- 1.3 Novedades de las últimas versiones del JDK.

Tema 2: DESARROLLO DE PÁGINAS WEB

- 2.1 HTML.
- 2.2 CSS.
- 2.3 Javascript.
- 2.4 Despliegue de contenidos web: configuración y puesta en marcha de un servidor web.

Tema 3: DESARROLLO DE APLICACIONES WEB CON JAVA

- 4.1 Aplicaciones web basadas en JSPs y Servlets.
- 4.2 Patrón MVC.
- 4.3 Cookies y Sesiones.
- 4.4 Lenguaje EL/JSTL.
- 4.5 Ajax.

Tema 4: HTTP Y TÉCNICAS PARA MEJORAR EL DESARROLLO WEB

- 3.1 Protocolo HTTP.
- 3.2 Técnicas y recomendaciones para la mejora de nuestros sitios web.

Tema 5: CONSIDERACIONES EN EL DESARROLLO WEB

- 5.1 Accesibilidad.
- 5.2 Posicionamiento Web: SEO.
- 5.3 Metodología de desarrollo web por capas:
 - Graceful Degradation.
 - Progressive Enhancement.

Tema 6: SEGURIDAD EN LA PROGRAMACIÓN WEB

- 6.1 Conceptos básicos de seguridad.
- 6.2 Tipos de ataques.
 - Form Tampering.
 - Inyecciones SQL.
 - Cross-site Scripting (XSS).
 - Cross-site Request Forgery (CSRF).
- 6.3 Desarrollo seguro de aplicaciones Web.
 - OWASP.

Tema 7: PROGRAMACIÓN EN DISPOSITIVOS MÓVILES

- 7.1 Programación web en móviles:
 - Librerías y frameworks web.
 - Responsive Web Design.
- 7.2 Programación nativa en de aplicaciones móviles:
 - Android.
 - iOS.

METODOLOGÍA DOCENTE

Aspectos metodológicos generales de la asignatura

Con el fin de conseguir el desarrollo de competencias propuesto, la materia se desarrollará teniendo en cuenta la actividad del alumno como factor prioritario. Ello implicará que tanto las sesiones presenciales como las no presenciales promoverán la implicación activa de los alumnos en las actividades de aprendizaje.

Metodología Presencial: Actividades

1. **Lección expositiva:** El profesor desarrolla el tema que previamente los alumnos han leído, explicándolo en la pizarra. Una vez desarrollados los conceptos teóricos se aplican a un caso real. Para ello el profesor y/o los propios alumnos, proponen un problema cotidiano al cual se le intenta dar solución con la participación de los alumnos. A continuación, el profesor codifica dicho programa en el ordenador con la participación de los de los alumnos. Este programa codificado por el profesor es facilitado a los alumnos a través de la web de la asignatura.
2. **Debates grupales, pruebas y resolución de ejercicios:** En estas sesiones se resolverán las dudas surgidas de las lecturas llevadas a cabo por los alumnos de los temas o contenidos propuestos por el profesor. También se realizarán debates grupales y pruebas para facilitar la comprensión de los mismos. Además, se resolverán los pequeños ejercicios propuestos por el profesor.
3. **Prácticas de laboratorio.** A lo largo del curso los alumnos se realizarán prácticas individuales de cada uno de los temas expuestos. En el laboratorio se finalizará el desarrollo de la práctica y se resolverán las dudas pertinentes.
4. **Tutorías.** Se realizarán en grupo e individualmente para resolver las dudas planteadas a los alumnos después de trabajar sobre los distintos temas. También se utilizarán para orientar al alumno en su proceso de aprendizaje.

Metodología No presencial: Actividades

1. **Estudio teórico:**
 - a. Estudio y preparación de los temas o conceptos que vayan a ser expuestos por el profesor.
 - b. Estudio individual y personal por parte del alumno de los conceptos ya expuestos en las lecciones expositivas.
2. **Casos prácticos:** Preparación y comienzo del desarrollo de las prácticas de laboratorio propuestas semanalmente por el profesor.
3. **Proyecto:** Desarrollo de la práctica final de curso sobre la cual se trabajará incrementalmente todo el curso. Se realizarán entregas parciales al profesor.

El objetivo principal del trabajo no presencial es llegar a entender y comprender los conceptos teóricos de la asignatura, así como ser capaz de poner en práctica estos conocimientos para resolver los diferentes tipos de problemas.

RESUMEN HORAS DE TRABAJO DEL ALUMNO			
HORAS PRESENCIALES			
Lección magistral	Debates y ejercicios	Prácticas laboratorio	Pruebas
33	9	28	5
HORAS NO PRESENCIALES			
Trabajo autónomo sobre contenidos teóricos	Trabajo autónomo sobre contenidos prácticos	Realización de trabajos colaborativos	Estudio
13	60	5	34
CRÉDITOS ECTS:			7,5 (187 horas)

EVALUACIÓN Y CRITERIOS DE CALIFICACIÓN

Actividades de evaluación	Criterios de evaluación	PESO
Realización de exámenes: <ul style="list-style-type: none"> Examen final. 	<ul style="list-style-type: none"> Comprensión de conceptos mediante pruebas abiertas, enfocado a casos prácticos. Aplicación de los conceptos adquiridos en forma de un desarrollo de software. Presentación y comunicación escrita. 	60%
Realización de pruebas de seguimiento, debates y ejercicios.	<ul style="list-style-type: none"> Comprensión de conceptos mediante pruebas objetivas y one minute papers. Aplicación de conceptos al desarrollo de pequeños programas temáticos sobre los nuevos conceptos mediante pruebas prácticas abiertas. Análisis e interpretación de los resultados obtenidos en la resolución de problemas mediante debates. 	10%
Prácticas: <ul style="list-style-type: none"> Casos a desarrollar en la clase de prácticas. Proyecto: práctica Final 	<ul style="list-style-type: none"> Comprensión de conceptos. Aplicación de conceptos al desarrollo de casos (pequeños programas temáticos) sobre los nuevos conceptos planteados. Aplicación de todos los conocimientos adquiridos durante el curso en forma de un desarrollo de un proyecto de software incremental, mediante entregas periódicas. Realización de pruebas en el laboratorio para comprobar la destreza en un entorno real de desarrollo. 	30%

Calificaciones.

Calificaciones

La calificación en la **convocatoria ordinaria** de la asignatura se obtendrá como:

- Un 60% la calificación del examen final.
- Un 10% será la calificación de las pruebas de seguimiento.
- Un 30% será la calificación de las prácticas. La práctica final supondrá un 20% y las prácticas realizadas en clase un 10%.

La calificación en la **convocatoria extraordinaria**:

- Un 60% la calificación del examen final.
- Un 10% será la calificación de las pruebas de seguimiento.
- Un 30% será la calificación de las prácticas. La práctica final supondrá un 20% y las prácticas realizadas en clase un 10%.

Para aprobar la asignatura los alumnos tienen que tener al menos 5 puntos sobre 10 en el examen final de la asignatura, tanto en la convocatoria ordinaria como en la extraordinaria.

PLAN DE TRABAJO Y CRONOGRAMA

Actividades No presenciales	Fecha de realización	Fecha de entrega
<ul style="list-style-type: none">• Lectura y estudio de los contenidos teóricos en el libro de texto	Después de cada clase	
<ul style="list-style-type: none">• Resolución de los problemas propuestos o debates	Semanalmente	
<ul style="list-style-type: none">• Entregas parciales de la práctica final		Semanas 4, 9 y 13
<ul style="list-style-type: none">• Pruebas que se realizarán durante las horas de clase	Semanas 3, 6 y 11	
<ul style="list-style-type: none">• Preparación del examen final	Junio	
<ul style="list-style-type: none">• Entrega de la práctica final		Día del examen

BIBLIOGRAFÍA Y RECURSOS

Bibliografía Básica

- Web específica de la asignatura desarrollada por el profesor con documentación, ejemplos y enunciados de prácticas en la plataforma Moodle.
- Professional Java for Web Applications. Nicholas S. Williams.

Bibliografía Complementaria

- Deepak Alur, Dan Malks, John Crupi. “Core J2EE Patterns: Best Practices and Design Strategies (2nd Edition)”. Prentice Hall. 2003.
- Eric Jendrock, Ian Evans. “The Java EE 6 Tutorial: Basic Concepts (4th Edition)”. Java Series. 2010.
- David Heffelfinger. “Java EE 6 Development with NetBeans 7”. Packt Publishing. 2011.
- Web oficial de Java EE: <http://docs.oracle.com/javaee/6/tutorial/doc/>

FICHA RESUMEN

Ver páginas siguientes².

² En la ficha resumen se encuentra una planificación detallada de la asignatura. Esta planificación tiene un carácter orientativo y las fechas podrán irse adaptando de forma dinámica a medida que avance el curso.

