

PLAN DOCENTE DE LA ASIGNATURA

Curso académico: 2026/2027

Identificación y características de la asignatura					
Código	501295				
Denominación (español)	Programación en Internet				
Denominación (inglés)	Internet Programming				
Titulaciones	Grado en Ingeniería Informática en Ingeniería del Software Grado en Ingeniería Informática en Ingeniería de Computadores				
Centro	Escuela Politécnica				
Módulo	De Tecnología Específica en Ingeniería del Software				
Materia	Programación en Internet (GII-IS) / Sistemas Ubicuos (GII-IC)				
Carácter	Obligatoria (GII-IS) / Optativa (GII-IC)	ECTS	6	Semestre	6º
Profesorado					
Nombre	Despacho	Correo-e	Página web		
Juan Manuel Murillo Rodríguez	Despacho 53 (Pabellón de Edificación)	juanmamu@unex.es	https://spilab.es/juanmamurillo.html		
Área de conocimiento	Lenguajes y Sistemas Informáticos				
Departamento	Ingeniería de Sistemas Informáticos y Telemáticos				
Profesor/a coordinador/a (si hay más de uno)	Juan Manuel Murillo Rodríguez				
Competencias					
1. Competencias básicas y generales establecidas para Grado en el Anexo I 3.2 del RD 861/2010.					
COMPETENCIAS BÁSICAS					
CB1: 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.					
CB2: 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.

CB3: Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética.

CB4: Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.

CB5: Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.

COMPETENCIAS GENERALES

CG02: Capacidad para dirigir las actividades objeto de los proyectos del ámbito de la Informática de acuerdo con los conocimientos adquiridos según lo establecido en el apartado 5 del Anexo II de la resolución antes mencionada para la tecnología específica de Ingeniería del Software / Ingeniería de Computadores.

CG03: Capacidad para diseñar, desarrollar, evaluar y asegurar la accesibilidad, ergonomía, usabilidad y seguridad de los sistemas, servicios y aplicaciones informáticas, así como de la información que gestionan.

CG08: Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.

CG09: Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.

2. La asignatura cubre, total o parcialmente, las siguientes **competencias técnicas y sus resultados de aprendizaje** especificadas en los planes de estudio:

CIS03: Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.

CIS05: Capacidad de identificar, evaluar y gestionar los riesgos potenciales asociados que pudieran presentarse.

CIS06: 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.

3. La asignatura cubre, total o parcialmente, las siguientes **competencias técnicas y sus resultados de aprendizaje** especificadas en los planes de estudio:

CIC04: Capacidad de diseñar e implementar software de sistema y de comunicaciones.

CIC06: Capacidad para comprender, aplicar y gestionar la garantía y seguridad de los sistemas informáticos.

4. La asignatura cubre en un nivel básico, total o parcialmente, las siguientes **competencias transversales y sus resultados de aprendizaje** especificadas en los planes de estudio:

CT02: Habilidades de gestión de recursos de información.

CT06: Capacidad de comunicación efectiva en inglés.

Contenidos
Breve descripción del contenido
Programación de páginas estáticas (HTML, JavaScript). Despliegue de aplicaciones (Servidores. Cliente/Servidor). Programación de aplicaciones dinámicas (SP, PHP, ASP, acceso a bases de datos). Desarrollo de aplicaciones intensivas de datos (Sesiones, AJAX). Rich Internet Applications. Tecnologías RIA: Flash, Action Script, Flex, AJAX, Silverlight, CURL, AIR. Desarrollo de aplicaciones Web empresariales. Clustering de datos y lógica de negocio. Seguridad de aplicaciones Web y Pasarelas de pago. Control de versiones. Herramientas y frameworks específicos. Tendencias en el desarrollo Web. Hoy en día, Internet es el paradigma predominante para la construcción y despliegue de aplicaciones software distribuidas. La estandarización y consolidación de los protocolos en internet ha permitido la construcción de aplicaciones compuestas por servicios distribuidos en todo el mundo que pueden utilizarse desde prácticamente cualquier plataforma hardware. El objetivo de esta asignatura es introducir al alumno en el diseño arquitectónico y en la construcción de este tipo de aplicaciones. Para ello la asignatura incluye entre otros los contenidos de programación de la parte cliente (frontend), programación de la parte servidora (backend), la lógica de negocio y la persistencia de datos.
Temario de la asignatura
Tema 0: Presentación de la asignatura Contenidos del tema 0: Presentación del profesorado. Objetivos de la asignatura. Temario. Metodología. Hitos principales en el semestre. Evaluación continua y Evaluación ordinaria
Tema 1: Introducción a la programación en Internet Contenidos del tema 1: Aplicaciones en internet. Protocolos en Internet. URI y http. Arquitectura de aplicaciones WEB. Patrones de diseño para aplicaciones web. Ingeniería WEB y metodologías de desarrollo WEB. Descripción de las actividades prácticas del tema 1: Instalación del entorno de desarrollo WEB.
Tema 2: Cliente WEB. Front-end Contenidos del tema 2: Arquitectura de clientes WEB. Buenas prácticas en el desarrollo de clientes WEB. Tecnologías de desarrollo de clientes web (HTML5, CSS, JavaScript) Descripción de las actividades prácticas del tema 2: Presentación del proyecto. Desarrollo del prototipo de cliente web.
Tema 3: Servidor WEB. Back-end Contenidos del tema 3: Lógica de negocio. Intercambio de datos (XML, JASON, YAML). Seguridad (SSL). Persistencia de datos (patrones DAO, FACTORY y DTO). Estilos arquitectónicos basados en servicios. RESTful y OpenAPI.

Descripción de las actividades prácticas del tema 3: Desarrollo de la lógica de negocio del Proyecto en el servidor web.

Tema 4: Contenedores y despliegue

Contenidos del tema 4: Tecnología de contenedores, generación imágenes de contenedores. Despliegue

Descripción de las actividades prácticas del tema 4: Generación y despliegue de imágenes de contenedores Docker.

Actividades formativas

Horas de trabajo del alumno/a por tema		Horas Gran grupo	Actividades prácticas				Actividad de seguimiento	No presencial
Tema	Total	GG	CH	L	O	S	TP	EP
0	2	2	0	0	0	0	0	0
1	20	4	0	4	0	0	0	12
2	33	6	0	8	0	0	1	18
3	61	12	0	12	0	0	1	36
4	21	4	0	4	0	0	1	12
Evaluación	13	2	0	2	0	0	0	9
TOTAL	150	30	0	30	0	0	3	87

GG: Grupo Grande (85 estudiantes).

CH: Actividades de prácticas clínicas hospitalarias (7 estudiantes)

L: Actividades de laboratorio o prácticas de campo (15 estudiantes)

O: Actividades en sala de ordenadores o laboratorio de idiomas (20 estudiantes)

S: Actividades de seminario o de problemas en clase (40 estudiantes).

TP: Tutorías Programadas (seguimiento docente, tipo tutorías ECTS).

EP: Estudio personal, trabajos individuales o en grupo, y lectura de bibliografía.

Metodologías docentes

Atendiendo a los paradigmas de "learning by doing" y "problem-based learning", para alcanzar los objetivos de aprendizaje de la asignatura se plantearán las siguientes actividades formativas:

Presenciales en grupo grande

- clase de explicación de conceptos
- clase de ejercicios y problemas
- resolución de ejercicios y problemas
- desarrollo de problemas en común
- presentación de problemas resueltos
- resolución de ejercicios de test

Presenciales en el laboratorio

- ejercicios guiados
- ejemplos de software desarrollado
- implementación de ejercicios de programación
- detección de errores de programas
- uso de librerías externas
- realización del proyecto de programación
- uso de estructuras de datos
- detección de errores
- ejecución de pruebas de código
- uso del aula virtual

No presenciales

- estudio de temas
- trabajo con el entorno de trabajo utilizado
- búsqueda de información (libros, Internet, etc.)
- reuniones de grupos
- realización de proyecto de programación
- realización de documentación externa e interna del proyecto de programación
- uso de foros de la asignatura
- resolución de problemas planteados en sesiones teóricas y prácticas

Resultados de aprendizaje

Al completar el módulo específico, el estudiante:

- Tiene los conocimientos y capacidades necesarios para dar solución a problemas de integración y despliegue de aplicaciones Web en función de las estrategias, estándares y tecnologías disponibles.
- Tiene capacidad para diseñar soluciones en internet apropiadas en uno o más dominios de aplicación utilizando métodos de la ingeniería del software que integren aspectos éticos, sociales, legales y económicos.
- Conoce las tecnologías básicas usadas actualmente para la programación Web.
- Tiene una visión de las tendencias actuales en programación Web.
- Tiene capacidad para identificar, evaluar y gestionar los riesgos potenciales asociados al desarrollo y uso de aplicaciones web.
- Tiene capacidad para comprender, aplicar y gestionar la garantía y seguridad de los sistemas web.
- Tiene capacidad para diseñar e implementar software de sistema y de comunicaciones para la web.
- Conoce y aplica en actividades avanzadas las competencias transversales fundamentales de la profesión.

Sistemas de evaluación

La asignatura ofrece **2 itinerarios de evaluación diferentes:**

Itinerario A: Evaluación Continua. El itinerario de evaluación continua requiere la asistencia del alumno a todas las sesiones de Grupo Grande, Laboratorio y Tutoría Programada de la asignatura. Además, el alumno ha de realizar con éxito los 4 trabajos prácticos (P1 a P4) propuestos a lo largo del curso y que se evaluarán de forma independiente. La calificación de cada uno de ellos deberá ser superior a 5. La calificación de los trabajos prácticos se obtiene como una media ponderada de la calificación obtenida en cada trabajo según la siguiente fórmula:

$$\text{Calificación Trabajos} = (10\% \text{ P1} + 30\% \text{ P2} + 35\% \text{ P3} + 25\% \text{ P4})$$

Si la calificación en algún trabajo fue menor a 5, la calificación del alumno calificación en los trabajos será calculada como la inferior menor que 5 obtenida en los trabajos no superados.

El trabajo del alumno realizado en las memorias técnicas de los trabajos prácticos así como el trabajo de defensa realizado en las tutorías programadas formará también parte de la calificación final del alumno. El alumno deberá obtener una calificación mínima de 5 para poder superar este apartado de su evaluación.

Finalmente, el alumno habrá de realizar un examen final que tratará sobre el contenido teórico de la asignatura. Se exige una calificación mínima de 5 para poder superar este examen.

Siempre y cuando se hayan alcanzado las calificaciones mínimas de 5 en los tres apartados descritos, la calificación final del alumno será obtenida mediante la siguiente fórmula:

$$\text{Calificación Final} = 70\% \text{ Calificación Trabajos} + 20\% \text{ Calificación de memorias y defensa en tutorías programadas} + 10\% \text{ Calificación Examen Teórico}$$

En otro caso, su calificación será la de la mínima nota por debajo de 5 obtenida en los apartados.

Todos aquellos alumnos cuya calificación final sea inferior a 5 podrán evaluarse mediante el modelo de evaluación mediante examen en cualquiera de las convocatorias.

Itinerario B: Evaluación mediante Examen. Para los alumnos que no hayan superado la evaluación continua. En este caso el alumno ha de realizar la prueba de examen del itinerario A. Además, habrá de superar otra prueba de examen relativa a los contenidos prácticos de la asignatura. Para superar la asignatura será necesario obtener una calificación superior a 5 en cada uno de los dos apartados. La evaluación en este caso se compone a partir de la nota conseguida en ambas pruebas mediante la siguiente fórmula:

$$\text{Calificación Final} = 70\% \text{ Calificación examen práctico} + 30\% \text{ Calific. Examen Teórico}$$

En caso de no obtener una calificación superior a 5 en alguno de los dos apartados, la calificación final del alumno será la de la máxima nota por debajo de 5 obtenida.

Durante las primeras 3 semanas de curso el alumno podrá optar por cualquiera de los modelos de evaluación detallados anteriormente.

Bibliografía (básica y complementaria)

Bibliografía básica

1. Head First HTML and CSS, 2nd Edition. Elisabeth Robson, Eric Freeman. O'Reilly Media. 2012
2. Head First Servlets and JSP, 2nd Edition. Bryan Basham, Kathy Sierra & Bert Bates. O'Reilly Media. 2008
3. RESTful Java with JAX-RS 2.0 2nd Edition. Bill Burke. O'Reilly Media. 2013

4. Head First JavaScript Programming. Eric T. Freeman, Elisabeth Robson. O'Reilly Media. 2014
5. AngularJS: Up and Running: Enhanced Productivity with Structured Web Apps. Shyam Seshadri, Brad Green. O'Reilly Media. 2014

Bibliografía complementaria

6. Head First HTML5 Programming: Building Web Apps with JavaScript. Eric Freeman. O'Reilly, 2010
7. HTML5 & CSS3: Develop with Tomorrow's Standards Today. Brian P. Hogan. Pragmatic Programmers, 2010
8. Head First Mobile Web. Lyza Danger Gardner, Jason Grigsby. O'Reilly, 2010
9. Web Applications Design Patterns. Pawan Vora. Morgan Kaufmann Publishers/Elsevier, 2009.
10. JavaScript: The Good Parts. Unearthing the Excellence in JavaScript. Douglas Crockford. O'Reilly Media / Yahoo Press, 2008.
11. Internet and World Wide Web How To Program, 5/E. Paul Deitel, Harvey Deitel & Abbey Deitel. Prentice Hall, 2012.

Otros recursos y materiales docentes complementarios