



x

Guía docente de la asignatura

Asignatura	ARQUITECTURAS Y TECNOLOGÍAS PARA EL DESARROLLO DE APLICACIONES DISTRIBUIDAS		
Materia	COMPLEMENTOS DE TELEMÁTICA para Graduados en Tecnologías Específicas de Telecomunicación, Mención en Sistemas Electrónicos		
Módulo			
Titulación	MASTER EN INGENIERÍA DE TELECOMUNICACIÓN		
Plan	544	Código	53800
Periodo de impartición	1 ^{er} . CUATRIMESTRE	Tipo/Carácter	OPTATIVA*
Nivel/Ciclo	MASTER	Curso	COMPLEMENTOS DE FORMACIÓN
Créditos ECTS	6 ECTS		
Lengua en que se imparte	CASTELLANO		
Profesor/es responsable/s	MÍRIAM ANTÓN RODRÍGUEZ MARÍA ÁNGELES PÉREZ JUÁREZ		
Datos de contacto (E-mail, teléfono...)	TELÉFONO: 983 423000 ext. 3716 / ext. 3709 E-MAIL: mirant@tel.uva.es, mperez@tel.uva.es		
Horario de tutorías	Véase www.uva.es → Másteres → Ingeniería de Telecomunicación → Tutorías		
Departamento	TEORÍA DE LA SEÑAL Y COMUNICACIONES E INGENIERÍA TELEMÁTICA		

(*) Esta asignatura es optativa a nivel de título pero es obligatoria para los alumnos que acceden al máster desde el Grado en Ingeniería de Tecnologías Específicas de Telecomunicación, Mención en Sistemas Electrónicos.



1. Situación / Sentido de la Asignatura

1.1 Contextualización

Cuando, durante su trabajo en el Laboratorio Europeo de Física de Partículas (CERN), al científico británico Tim Berners-Lee le hacen el encargo de desarrollar un método eficiente y rápido para intercambiar la documentación generada en el seno de la comunidad científica, ni el mismo podía sospechar que su trabajo para dar solución a dicho problema acabaría convirtiéndose en la base de lo que hoy conocemos como el World Wide Web, una telaraña mundial en constante crecimiento y evolución para la cual se desarrollan constantemente nuevas aplicaciones.

Igualmente es importante notar el hecho de que, en la actualidad, el desarrollo de aplicaciones, incluidas las aplicaciones web, suele estar asociado a una distribución de elementos o responsabilidades desde un punto de vista lógico, hecho que, para el caso de muchas aplicaciones, incluidas las aplicaciones web, está además asociado a una distribución física de elementos o responsabilidades. Esta distribución lógica y/o física que podemos considerar muy frecuente e incluso obligatoria en el desarrollo de las aplicaciones software actuales, requiere de arquitecturas y tecnologías específicas que se presentarán en esta asignatura.

Los conocimientos adquiridos en la presente asignatura son, además de interesantes, ampliamente demandados por la industria del software puesto que, en la actualidad, buena parte de las aplicaciones que se desarrollan son aplicaciones web y/o distribuidas, motivo por el cual esta asignatura cuenta con un importante mercado laboral, tanto dentro de nuestras fronteras como fuera de ellas, y, tanto real como potencial, puesto que constantemente aparecen nuevas tecnologías o mejoras en las ya existentes, que abren la puerta a nuevas posibilidades hasta entonces no contempladas o a formas más eficientes de conseguir un mismo resultado.

1.2 Relación con otras materias

Esta asignatura está enmarcada dentro de los Complementos de Telemática para los graduados en Tecnologías Específicas de Telecomunicación mención en Sistemas Electrónicos, que constan de tres asignaturas todas ellas a impartirse durante el 1er cuatrimestre:

- Arquitecturas y tecnologías para el desarrollo de aplicaciones distribuidas
- Operación, administración y mantenimiento de redes telemáticas
- Teletráfico y calidad de servicio

Esta materia se apoya en las competencias generales y específicas fomentadas en el “Bloque de materias básicas de telecomunicaciones” del Grado en Tecnologías Específicas de Telecomunicación – mención en Sistemas Electrónicos – que facilitan la adquisición de competencias específicas comunes a los Complementos de Telemática relacionadas con el desarrollo de aplicaciones telemáticas, aunque no está específicamente vinculada o condicionada por ninguna de las asignaturas que componen dicho bloque. Igualmente, esta



Guía docente de la asignatura

asignatura se apoya en la asignatura “Programación” de la materia “Informática” del “Bloque de Materias Instrumentales”.

Por otra parte, y dado que la presente asignatura es una asignatura que aborda el desarrollo de aplicaciones software, sí es importante, aunque no requisito indispensable, el haber cursado y asimilado los conceptos presentados en la asignatura de “Ingeniería de Sistemas Software” de la materia “Fundamentos de Sistemas Software” del “Bloque de Materias Básicas de Telecomunicaciones”, que supone una aproximación a la disciplina que se ocupa del software, entendiendo el mismo como un producto de ingeniería que requiere planificación, análisis, diseño, implementación, pruebas y mantenimiento, puesto que el enfoque para trabajar sobre el tipo de aplicaciones abordadas en la presente asignatura deberá ser ese mismo. Además, para el bloque de “Tecnologías para Aplicaciones Distribuidas” de la presente asignatura, es aconsejable conocer el paradigma de Orientación a Objetos, así como las nociones del lenguaje de programación Java presentadas en dicha asignatura. Igualmente, le resultará de utilidad al alumno el haber cursado y asimilado los conceptos presentados en la asignatura “Programación” de la materia “Informática” del “Bloque de Materias Instrumentales” ya mencionada en el párrafo anterior.

1.3 Prerrequisitos

No existen condiciones previas excluyentes para cursar esta asignatura, aunque sí recomendaciones lógicas que el alumno debería tener en cuenta. En concreto, es recomendable haber cursado con anterioridad la asignatura “Programación” de la materia “Informática” del “Bloque de Materias Instrumentales” que se imparte en el 1er cuatrimestre del 1er curso del Grado en Tecnologías Específicas de Telecomunicación – mención en Sistemas Electrónicos –. Así como la asignatura de “Ingeniería de Sistemas Software” de la materia “Fundamentos de Sistemas Software” del “Bloque de Materias Básicas de Telecomunicaciones” que se imparte en el 2º cuatrimestre del 2º curso del Grado en Tecnologías Específicas de Telecomunicación – mención en Sistemas Electrónicos –. En dichas asignaturas se proporcionarán los conceptos básicos relativos al desarrollo de aplicaciones software que facilitarán al alumno la comprensión de otros mostrados en la presente materia.

En especial, debe tenerse en cuenta que para el bloque de “Tecnologías para Aplicaciones Distribuidas” de la presente asignatura, es aconsejable conocer el paradigma de Orientación a Objetos, así como las nociones del lenguaje de programación Java presentadas en la asignatura de “Ingeniería de Sistemas Software”.

2. Competencias

2.1 Generales

- Capacidad para resolver problemas con iniciativa, creatividad y razonamiento crítico.
- Capacidad para diseñar y llevar a cabo experimentos, así como analizar e interpretar datos.
- Capacidad para elaborar informes basados en el análisis crítico de la bibliografía técnica y de la realidad en el campo de su especialidad.



Guía docente de la asignatura

- Capacidad para trabajar en un grupo multidisciplinar y multilingüe, responsabilizándose de la dirección de actividades objeto de los proyectos del ámbito de su especialidad y consiguiendo resultados eficaces.
- Capacidad para desarrollar metodologías y destrezas de aprendizaje autónomo eficiente para la adaptación y actualización de nuevos conocimientos y avances científicos.
- Capacidad de organización, planificación y gestión del tiempo.
- Capacidad para comunicar, tanto por escrito como de forma oral, conocimientos, procedimientos, resultados e ideas relacionadas con las telecomunicaciones y la electrónica.
- Capacidad para trabajar en cualquier contexto, individual o en grupo, de aprendizaje o profesional, local o internacional, desde el respeto a los derechos fundamentales, de igualdad de sexo, raza o religión y los principios de accesibilidad universal, así como la cultura de paz.

2.2 Específicas

- Capacidad de construir, explotar y gestionar las redes, servicios, procesos y aplicaciones de telecomunicaciones, entendidas éstas como sistemas de captación, transporte, representación, procesado, almacenamiento, gestión y presentación de información multimedia, desde el punto de vista de los servicios telemáticos.
- Capacidad de diseñar arquitecturas de redes y servicios telemáticos.
- Capacidad de programación de servicios y aplicaciones telemáticas, en red y distribuidas.

3. Objetivos

Al finalizar la asignatura el alumno deberá ser capaz de:

- Diseñar e implementar interfaces web que sean capaces de interactuar con sus usuarios utilizando para ello de forma adecuada un conjunto de herramientas y soluciones tecnológicas como son los lenguajes de marcado, las hojas de estilo o los lenguajes de script orientados a eventos, en función de los requisitos del problema bajo estudio.
- Diseñar e implementar y utilizar bases de datos relacionales.
- Diseñar e implementar aplicaciones web complejas que incluyan la interacción de su interfaz con una base de datos utilizando para ello de forma adecuada un conjunto de lenguajes como son el Lenguaje de Consultas Estructurado (SQL) o PHP en función de los requisitos del problema bajo estudio.
- Comprender la problemática específica del desarrollo de aplicaciones distribuidas, así como el concepto de middleware y su importancia en el ámbito de las aplicaciones distribuidas.
- Conocer de forma comparada los distintos tipos de middleware, así como las principales soluciones existentes en el ámbito del middleware orientado a objetos, a componentes y a servicios.
- Diseñar e implementar aplicaciones distribuidas utilizando para ello de forma adecuada un conjunto de herramientas y soluciones tecnológicas, en función de los requisitos del problema bajo estudio.
- Aprender de forma autónoma:
 - Localizar y asimilar una determinada información a partir de su referencia.
 - Buscar información técnica relevante para una tarea especificada.



Guía docente de la asignatura

- Autoevaluar el trabajo propio realizado, identificando sus errores y aspectos a mejorar.
- Trabajar en equipo:
 - Intercambiar información a través del correo electrónico, foros de discusión y otras herramientas de comunicación síncrona y/o asíncrona.
 - Explicar a un equipo el resultado de una tarea realizada y asegurarse de que todos los demás lo han comprendido.
 - Identificar adecuadamente las tareas a realizar por el equipo, repartir equitativamente las tareas, establecer plazos de entrega, e integrar el trabajo realizado por los diferentes miembros del equipo.
 - Identificar los aspectos positivos relativos al funcionamiento del equipo. Identificar y abordar los aspectos a mejorar así como los conflictos en el funcionamiento del equipo.
- Realizar documentación de carácter técnico.

4. Tabla de dedicación del estudiante a la asignatura

ACTIVIDADES PRESENCIALES	HORAS	ACTIVIDADES NO PRESENCIALES	HORAS
Clases teórico-prácticas (T/M)	15	Estudio y trabajo autónomo individual	40
Clases prácticas de aula (A)		Estudio y trabajo autónomo grupal	50
Laboratorios (L)	45		
Prácticas externas, clínicas o de campo			
Seminarios (S)			
Tutorías grupales (TG)			
Evaluación (fuera del periodo oficial de exámenes)			
Total presencial	60	Total no presencial	90

5. Bloques temáticos

Bloque 1: Tecnologías para Aplicaciones Web

Carga de trabajo en créditos ECTS:

a. Contextualización y justificación

Cuando, durante su trabajo en el Laboratorio Europeo de Física de Partículas (CERN), al científico británico Tim Berners-Lee le hacen el encargo de desarrollar un método eficiente y rápido para intercambiar la documentación generada en el seno de la comunidad científica, ni el mismo podía sospechar que su trabajo para dar solución a dicho problema acabaría convirtiéndose en la base de lo que hoy conocemos como el World Wide Web, una telaraña mundial en constante crecimiento y evolución para la cual se desarrollan constantemente nuevas aplicaciones lo cual constituye el objeto del presente bloque temático.



b. Objetivos de aprendizaje

Al finalizar este bloque temático, el alumno deberá ser capaz de:

- Diseñar e implementar interfaces web que sean capaces de interactuar con sus usuarios utilizando para ello de forma adecuada un conjunto de herramientas y soluciones tecnológicas como son los lenguajes de marcado, las hojas de estilo o los lenguajes de script orientados a eventos, en función de los requisitos del problema bajo estudio.
- Diseñar e implementar y utilizar bases de datos relacionales.
- Diseñar e implementar aplicaciones web complejas que incluyan la interacción de su interfaz con una base de datos utilizando para ello de forma adecuada un conjunto de lenguajes como son el Lenguaje de Consultas Estructurado (SQL) o PHP en función de los requisitos del problema bajo estudio.

Además se habrá trabajado sobre los objetivos horizontales como Aprendizaje autónomo, Autoevaluación del trabajo propio realizado, Trabajo en equipo y Realización de documentación de carácter técnico. [Véanse los objetivos de la asignatura].

c. Contenidos

TEMA 1: Tecnologías para el desarrollo de Aplicaciones Web: Lado Cliente

- 1.1 Metalenguajes y Lenguajes de Marcado: HTML, XHTML y XML
 - 1.1.1 ¿Para qué sirven los Metalenguajes y los Lenguajes de Marcado?
 - 1.1.2 Sintaxis
 - 1.1.3 Ejemplos
- 1.2 Hojas de Estilo en Cascada
 - 1.2.1 ¿Para qué sirven las Hojas de Estilo en Cascada?
 - 1.2.2 Sintaxis
 - 1.2.3 Ejemplos
- 1.3 JavaScript
 - 1.3.1 ¿Para qué sirve JavaScript?
 - 1.3.2 Sintaxis
 - 1.3.3 Ejemplos

TEMA 2: Tecnologías para el desarrollo de Aplicaciones Web: Lado Servidor

- 2.1 Diseño de Bases de Datos Relacionales
 - 2.1.1 ¿Qué son las Bases de Datos Relacionales?
 - 2.1.2 Proceso de Normalización
 - 2.1.3 Ejemplos
- 2.2 Lenguaje de Consultas Estructurado (SQL)
 - 2.2.1 ¿Para qué sirve el Lenguaje de Consultas Estructurado?
 - 2.2.2 Sintaxis



Guía docente de la asignatura

2.2.3 Ejemplos

2.3 PHP

2.3.1 ¿Para qué sirve PHP?

2.3.2 Sintaxis

2.3.3 Ejemplos

PRÁCTICAS

- El trabajo en el laboratorio se desarrollará en torno a un proyecto consistente en el desarrollo de una aplicación web. También se desarrollará un pequeño trabajo de documentación que permitirá a los alumnos aproximarse a los conceptos de Usabilidad y Accesibilidad Web. El objetivo final será el que sus proyectos de desarrollo web sean usables y accesibles.
- Las entregas concretas y su planificación se detallará en el Anexo I mencionado en la guía, que se facilitará al comienzo de la asignatura.

d. Métodos docentes

- Clase magistral.
- Aprendizaje colaborativo.
- Método de proyectos.

e. Plan de trabajo

Véase el Anexo I.

f. Evaluación

La evaluación de la adquisición de competencias se basará en:

- El código realizado por los alumnos sobre el proyecto de desarrollo de aplicaciones web en el que se trabaje en la asignatura.
- El trabajo de documentación sobre los conceptos de Usabilidad y Accesibilidad Web y el resultado de la aplicación de las pautas recopiladas sobre sus proyectos de desarrollo web.
- Las prueba escrita de evaluación al finalizar la asignatura.

g. Bibliografía básica

- Documentación básica de los diferentes temas de la asignatura proporcionada por las profesoras de la asignatura.
- Otros documentos complementarios también proporcionados por los profesores de la asignatura: enunciados de prácticas de laboratorio, artículos de congresos o revistas, etc., algunos de ellos extraídos de la bibliografía propuesta como complementaria.



h. Bibliografía complementaria

- R. Wagner, *JavaScript*, Sams.net. 3ª edición. 2000
- A. Silberschatz, H.F. Korth, *Fundamentos de Bases de Datos*, McGraw Hill. 4ª edición. 2002
- A. Lucas Gómez, *Diseño y Gestión de Bases de Datos*, Paraninfo. 1993
- J. R. Groff, *SQL: Manual de Referencia*, McGraw Hill. 1ª edición. 2003
- L. Ullman, *MySQL: Guía de Aprendizaje*, Prentice Hall. 1ª edición. 2003
- L. M. Cabezas Granado, *PHP: El Lenguaje para los Profesionales de la Web*, Anaya Multimedia. 2004
- L. Ullman, *PHP Advanced and Object-Oriented Programming*, Peachpit Press. 3ª edición. 2012
- L. Welling, *Desarrollo Web con PHP y MySQL*, Anaya Multimedia. 2009
- L. M. Cabezas Granado, *Manual imprescindible de PHP5*, Anaya Multimedia. 2004
- F. Charte Okeda, *Proyectos Profesionales PHP5*, Anaya Multimedia. 2004
- F.J. Gil Rubio, S. Alonso Villaverde, J.A. Tejedor Cerbel, A. Yagüe Panadero, *Creación de sitios web con PHP5*, McGrawHill. 2006

Debe también tenerse en cuenta que, sobre los temas tratados en la asignatura, existen multitud de libros y manuales, a mayores de los recomendados, algunos de los cuales están disponibles para su consulta en la biblioteca del Centro. Estos materiales pueden proporcionar explicaciones alternativas (que quizá le resulten más claras al alumno) y más ejemplos. También es posible encontrar tutoriales, cursos, artículos, etc., sobre los temas abordados en la asignatura en Internet. Asimismo una web de interés para este bloque temático es <http://www.w3.org>.

i. Recursos necesarios

Serán necesarios los siguientes recursos, todos ellos facilitados por el Centro y/o por los profesores de la asignatura:

- Aula con proyector multimedia y pizarra para las clases magistrales.
- Entorno de trabajo en la plataforma Moodle ubicada en el Campus Virtual de la Universidad de Valladolid u otra plataforma virtual alternativa.
- Laboratorio de prácticas, con al menos un ordenador para cada dos alumnos, para las sesiones de laboratorio. Cada ordenador contará con los entornos de desarrollo y aplicaciones necesarias para las prácticas planteadas.
- Documentación de apoyo.
- Acceso al material bibliográfico complementario recomendado, al menos a parte del mismo, en la biblioteca del Centro o mediante otras vías como Internet.

Bloque 2: Tecnologías para Aplicaciones Distribuidas

Carga de trabajo en créditos ECTS:

a. Contextualización y justificación



En este bloque se introduce el concepto de *middleware* o software intermediario como mecanismo para simplificar el desarrollo de las aplicaciones distribuidas que se caracterizan por estar formadas por partes que se ejecutan en distintos ordenadores interconectados. El uso del *middleware* permite que el desarrollador de la aplicación no tenga que incluir código relacionado con las tareas de bajo nivel necesarias para el intercambio de información a través de la red y la gestión de los posibles errores surgidos durante esta comunicación. De esta forma, las aplicaciones distribuidas pueden solicitar la ejecución de código en un ordenador remoto de forma similar a como se solicitaría la ejecución de una función local, con lo que se consigue que la complejidad de desarrollo de una aplicación distribuida sea muy parecida al de una aplicación centralizada.

En este concepto de *middleware* se basan las tecnologías más ampliamente utilizadas en la actualidad para el desarrollo de aplicaciones distribuidas. En este bloque se describirán los principales tipos de *middleware* existentes, así como las principales soluciones tecnológicas vinculadas a los mismos.

b. Objetivos de aprendizaje

Al finalizar este bloque temático, el alumno deberá ser capaz de:

- Comprender la problemática específica del desarrollo de aplicaciones distribuidas, así como el concepto de *middleware* y su importancia en el ámbito de las aplicaciones distribuidas.
- Conocer de forma comparada los distintos tipos de *middleware*, así como las principales soluciones existentes en el ámbito del *middleware* orientado a objetos, a componentes y a servicios.
- Diseñar e implementar aplicaciones distribuidas utilizando para ello de forma adecuada un conjunto de herramientas y soluciones tecnológicas, en función de los requisitos del problema bajo estudio.

Además se habrá trabajado sobre los objetivos horizontales como Aprendizaje autónomo, Autoevaluación del trabajo propio realizado, Trabajo en equipo y Realización de documentación de carácter técnico. [Véanse los objetivos de la asignatura].

c. Contenidos

TEMA 1: Introducción a los sistemas distribuidos y el middleware

- 1.1 Conceptos básicos: sistemas centralizados vs sistemas distribuidos
- 1.2 Requisitos para los sistemas distribuidos
- 1.3 Tipos de software distribuido:
 - 1.3.1 Aplicaciones en red
 - 1.3.2 Aplicaciones basadas en middleware

TEMA 2: Tipos de Middleware: Middleware Orientado a Objetos, Middleware Orientado a Componentes y Middleware Orientado a Servicios (Web)

- 2.1 Middleware Orientado a Objetos
 - 2.1.1 ¿Qué es el Middleware Orientado a Objetos?



Guía docente de la asignatura

- 2.1.2 Principales soluciones tecnológicas basadas en el Middleware Orientado a Objetos
- 2.2 Middleware Orientado a Componentes
 - 2.2.1 ¿Qué es el Middleware Orientado a Componentes?
 - 2.2.2 Principales soluciones tecnológicas basadas en el Middleware Orientado a Componentes
- 2.3 Middleware Orientado a Servicios
 - 2.3.1 ¿Qué es el Middleware Orientado a Servicios?
 - 2.3.2 Principales soluciones tecnológicas basadas en el Middleware Orientado a Servicios

Los contenidos de este bloque se completarán con un seminario dedicado a repasar los principales conceptos del paradigma de Programación Orientación a Objetos si resultara necesario.

PRÁCTICAS

- El trabajo en el laboratorio consistirá en la puesta en la puesta en práctica de diferentes tecnologías para el desarrollo de aplicaciones distribuidas.
- Las entregas concretas y su planificación se detallará en el Anexo I mencionado en la guía, que se facilitará al comienzo de la asignatura.

d. Métodos docentes

- Clase magistral.
- Aprendizaje colaborativo.
- Método de proyectos.

e. Plan de trabajo

Véase el Anexo I.

f. Evaluación

La evaluación de la adquisición de competencias se basará en:

- El código realizado por los alumnos sobre el proyecto de desarrollo de aplicaciones distribuidas en el que se trabaje en la asignatura.
- Las prueba escrita de evaluación al finalizar la asignatura.

g. Bibliografía básica

- Documentación básica de los diferentes temas de la asignatura proporcionada por las profesoras de la asignatura.
- Otros documentos complementarios también proporcionados por los profesores de la asignatura: enunciados de prácticas de laboratorio, artículos de congresos o revistas, etc., algunos de ellos extraídos de la bibliografía propuesta como complementaria.



h. Bibliografía complementaria

- H. Schildt, *Fundamentos de Java*, McGraw Hill. 2007
- G. F. Coulouris, J. Dollimore, T. Kindberg, *Distributed systems: concepts and design*, Addison Wesley. 5ª edición. 2012
- W. Emmerich, *Engineering distributed objects*. John Wiley & Sons. 2001.
- J. Pritchard, *COM and CORBA side by side: architectures, strategies, and implementations*, Addison-Wesley. 1999
- Z. Tari, O. Bukhres, *Fundamentals of distributed object systems: the CORBA perspective*, John Wiley & Sons. 2001
- C. Szyperski, *Component software: beyond object oriented programming*. Addison Wesley, 1999.
- S. Allamaraju et al, *Programación Java Server con J2EE*, Anaya multimedia. 2001
- Object Managment Group: Tutorial sobre CCM.
http://www.omg.org/news/meetings/workshops/RT_2003_Manual/Tutorials/T3_CCM_Wang-Rodrigues.pdf
- Microsoft: *.NET Framework Conceptual Overview*. <http://msdn.microsoft.com/en-us/library/zw4w595w.aspx>.
- G. Alonso, F. Casati, H. Kuno, V. Machiraju, *Web Services: Concepts, Architectures and Applications*, Springer. 2004

Debe también tenerse en cuenta que, sobre los temas tratados en la asignatura, existen multitud de libros y manuales, a mayores de los recomendados, algunos de los cuales están disponibles para su consulta en la biblioteca del Centro. Estos materiales pueden proporcionar explicaciones alternativas (que quizá le resulten más claras al alumno) y más ejemplos. También es posible encontrar tutoriales, cursos, artículos, etc., sobre los temas abordados en la asignatura en Internet. Asimismo una web de interés para este bloque temático es <http://www.omg.org/> o <http://www.w3.org/2002/ws>.

i. Recursos necesarios

Serán necesarios los siguientes recursos, todos ellos facilitados por el Centro y/o por los profesores de la asignatura:

- Aula con proyector multimedia y pizarra para las clases magistrales participativas.
- Entorno de trabajo en la plataforma Moodle ubicada en el Campus Virtual de la Universidad de Valladolid u otra plataforma virtual alternativa.
- Laboratorio de prácticas, con al menos un ordenador para cada dos alumnos, para las sesiones de laboratorio. Cada ordenador contará con los entornos de desarrollo y aplicaciones necesarias para las prácticas planteadas.
- Documentación de apoyo.
- Acceso al material bibliográfico complementario recomendado, al menos a parte del mismo, en la biblioteca del Centro o mediante otras vías como Internet.



6. Temporalización (por bloques temáticos)

BLOQUE TEMÁTICO	CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
Bloque 1: Tecnologías para Aplicaciones Web	3 ECTS	Semanas 1 a 7
Bloque 2: Tecnologías para Aplicaciones Distribuidas	3 ECTS	Semanas 8 a 15

7. Sistema de calificaciones – Tabla resumen

INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Entregables de equipo relativos al proyecto de desarrollo de aplicaciones web a realizar.	35%	Es condición necesaria (pero no suficiente) para superar la asignatura, el alcanzar una calificación igual o superior al 50% en esta parte. Si no se realiza alguno de los entregables establecidos, la calificación final en la asignatura será de No Presentado (N.P.).
Entregables de equipo relativos al proyecto de desarrollo de aplicaciones distribuidas a realizar.	35%	Es condición necesaria (pero no suficiente) para superar la asignatura, el alcanzar una calificación igual o superior al 50% en esta parte. Si no se realiza alguno de los entregables establecidos, la calificación final en la asignatura será de No Presentado (N.P.).
Prueba escrita final con dos partes correspondientes a los dos bloques temáticos de la asignatura.	30%	Para superar la asignatura será necesario demostrar que se ha adquirido un grado de conocimiento suficiente en los contenidos tratados en la asignatura mediante una prueba escrita que se realizará al final del cuatrimestre. Es condición necesaria (pero no suficiente) para superar la asignatura alcanzar una calificación igual o superior al 50% en cada parte del examen para que se haga la media ponderada con las otras calificaciones obtenidas por el alumno en la asignatura. En caso de no realizarse esta prueba escrita la calificación final en la asignatura será de No Presentado (N.P.).

Además deberá tenerse en cuenta que:

- La nota final para superar la asignatura en la convocatoria ordinaria deberá ser de 5.0 sobre 10.0.
- Si un alumno no alcanza los requisitos mínimos descritos en la tabla anterior, su calificación final en la asignatura será el mínimo entre el valor calculado según la ponderación descrita en la tabla y 4.5.



En el caso de la convocatoria extraordinaria:

- Se mantiene la calificación obtenida en el primer y en el segundo instrumento de la tabla en ese mismo curso académico siempre que la calificación obtenida sea de, al menos, el 50% en el instrumento considerado. No obstante, incluso cumpliéndose dicho requisito, el alumno podrá mejorar el proyecto práctico en cuestión, de desarrollo de aplicaciones web o de desarrollo de aplicaciones distribuidas, realizado, para incrementar su nota en ese instrumento.
- Se mantiene la calificación obtenida en el tercer instrumento de la tabla en ese mismo curso académico siempre que la calificación obtenida sea de, al menos, el 50% en cada parte del examen. No obstante, incluso cumpliéndose dicho requisito, el alumno podrá optar por realizar la prueba escrita de la convocatoria extraordinaria para mejorar su nota en este instrumento.

8. Consideraciones finales

- El Anexo I mencionado en la guía, donde se describe la planificación detallada, se entregará al comienzo de la asignatura.