

ANX-PR/CL/001-02
GUÍA DE APRENDIZAJE

ASIGNATURA

Analisis y diseño de software

CURSO ACADÉMICO - SEMESTRE

2014-15 - Segundo semestre

FECHA DE PUBLICACIÓN

Enero - 2015

Datos Descriptivos

Nombre de la Asignatura	Análisis y diseño de software
Titulación	09TT - Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación
Centro responsable de la titulación	E.T.S. de Ingenieros de Telecomunicacion
Semestre/s de impartición	Cuarto semestre
Módulo	Comun rama
Materia	Redes y servicios
Carácter	Obligatoria
Código UPM	95000024

Datos Generales

Créditos	4.5	Curso	2
Curso Académico	2014-15	Período de impartición	Febrero-Junio
Idioma de impartición	Castellano	Otros idiomas de impartición	

Requisitos Previos Obligatorios

Asignaturas Superadas

El plan de estudios Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación no tiene definidas asignaturas previas superadas para esta asignatura.

Otros Requisitos

El plan de estudios Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación no tiene definidos otros requisitos para esta asignatura.

Conocimientos Previos

Asignaturas Previas Recomendadas

Fundamentos de los sistemas telemáticos

Programación

Otros Conocimientos Previos Recomendados

El coordinador de la asignatura no ha definido otros conocimientos previos recomendados.

Competencias

CECT1 - Capacidad para aprender de manera autónoma nuevos conocimientos y técnicas adecuados para la concepción, el desarrollo o la explotación de sistemas y servicios de telecomunicación

CECT2 - Capacidad de utilizar aplicaciones de comunicación e informáticas (ofimáticas, bases de datos, cálculo avanzado, gestión de proyectos, visualización, etc.) para apoyar el desarrollo y explotación de redes, servicios y aplicaciones de telecomunicación y electrónica.

CECT7 - Conocimiento y utilización de los fundamentos de la programación en redes, sistemas y servicios de telecomunicación

CG2 - 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

CG9 - Uso de Tecnologías de la Información y de las Comunicaciones

Resultados de Aprendizaje

RA171 - Saber usar un entorno de programación para implementar, documentar, probar, empaquetar y desplegar programas.

RA172 - Conocer los principios básicos del análisis y diseño de algoritmos y aplicarlos a algoritmos representativos.

RA173 - Conocer los principios del análisis de complejidad de algoritmos y aplicarlos a algoritmos representativos.

RA174 - Conocer los conceptos básicos de la programación concurrente y los mecanismos de sincronización fundamentales.

RA175 - Conocer y comprender los problemas que pueden aparecer en programas concurrentes y las estrategias básicas para evitarlos

RA176 - Conocer un proceso de desarrollo de software para el desarrollo de aplicaciones telemáticas.

RA177 - Conocer una arquitectura de sistemas telemáticos representativa.

RA178 - Conocer, comprender y aplicar patrones de diseño en el proceso de desarrollo de software.

RA179 - Conocer, comprender y aplicar técnicas para el desarrollo de interfaces gráficas.

Profesorado

Profesorado

Nombre	Despacho	e-mail	Tutorías
Puente Alfaro, Juan Antonio De La (Coordinador/a)	B-318	juan.de.la.puente@upm.es	
Alonso Muñoz, Alejandro Antonio	B-319	alejandro.alonso@upm.es	
Alvarez Rodriguez, Angel	C-218	angel.alvarez@upm.es	
Huecas Fernandez-Toribio, Gabriel	B-212	gabriel.huecas@upm.es	
Iglesias Fernandez, Carlos Angel	C-211	carlosangel.iglesias@upm.es	
Mañas Argemi, Jose Antonio	C-219	joseantonio.manas@upm.es	

Nota.- Las horas de tutoría son orientativas y pueden sufrir modificaciones. Se deberá confirmar los horarios de tutorías con el profesorado.

Descripción de la Asignatura

La asignatura Análisis y diseño de software de segundo curso tiene como objetivo profundizar en el conocimiento de los conceptos, métodos y herramientas necesarios para desarrollar software de comunicaciones, a partir de los conocimientos adquiridos en las asignaturas Fundamentos de los Sistemas Telemáticos y Programación de primer curso.

Temario

1. Introducción
 - 1.1. Diseño de programas, depuración y pruebas.
2. Diseño de algoritmos
 - 2.1. Algoritmos recursivos
 - 2.2. Complejidad de los algoritmos: conceptos básicos y familias de algoritmos.
 - 2.3. Algoritmos para construir diccionarios. Análisis de complejidad
 - 2.4. Optimización: memoria frente a tiempo de ejecución.
3. Programación concurrente.
 - 3.1. Programas secuenciales y concurrentes. Hebras (threads).
 - 3.2. Interacción entre hebras. Variables comunes, exclusión mutua y comunicación sincronizada.
 - 3.3. Propiedades de programas concurrentes. Interbloqueos.
4. Interfaces y aplicaciones móviles
 - 4.1. Introducción a las aplicaciones móviles.
 - 4.2. Diseño de interfaces.
 - 4.3. Programación basada en eventos.
 - 4.4. Desarrollo de aplicaciones con Android.

Cronograma

Horas totales: 49 horas

Horas presenciales: 49 horas (41.9%)

Peso total de actividades de evaluación continua: 90%

Peso total de actividades de evaluación sólo prueba final: 100%

Semana	Actividad Presencial en Aula	Actividad Presencial en Laboratorio	Otra Actividad Presencial	Actividades Evaluación
Semana 1	Tema 1. Introducción Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 2	Tema 2.1 - Algoritmos recursivos Duración: 01:00 LM: Actividad del tipo Lección Magistral	Laboratorio 1 - Entorno de desarrollo, depuración y pruebas. Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
Semana 3	Tema 2.2 - Análisis de complejidad Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 4	Tema 2.3 - Diccionarios Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 5	Tema 2.4 - Optimización: memoria y tiempo de ejecución. Duración: 01:00 LM: Actividad del tipo Lección Magistral	Laboratorio 2 - Análisis de complejidad. Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
Semana 6	Tema 3.1 - Programas secuenciales y concurrentes Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 7	Tema 3.2 - Sincronización y monitores. Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 8	Tema 3.3 - Propiedades de programas concurrentes Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 9				
Semana 10	Tema 4.1 - Introducción a las aplicaciones móviles. Duración: 01:00 LM: Actividad del tipo Lección Magistral	Laboratorio 3 - Programación concurrente. Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
Semana 11	Tema 4.2 - Desarrollo de aplicaciones móviles Duración: 03:00 LM: Actividad del tipo Lección Magistral			Examen parcial 1 Duración: 02:00 EX: Técnica del tipo Examen Escrito Evaluación continua Actividad presencial

Semana 12	Tema 4.3 -Programación basada en eventos. Duración: 01:00 LM: Actividad del tipo Lección Magistral	Laboratorio 4 - Desarrollo de aplicaciones con Android. Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
Semana 13	Tema 4.4 - Desarrollo de aplicaciones con Android. Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 14	Tema 4.4 - Desarrollo de aplicaciones con Android. Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 15	Tema 4.4 - Desarrollo de aplicaciones con Android. Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 16	Tema 4. Consultas y aclaraciones. Duración: 01:00 LM: Actividad del tipo Lección Magistral	Laboratorio 5 - Consultas y apoyo para la realización del proyecto 2. Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
Semana 17				Examen parcial 2 Duración: 02:00 EX: Técnica del tipo Examen Escrito Evaluación continua Actividad presencial Examen final Duración: 03:00 EX: Técnica del tipo Examen Escrito Evaluación sólo prueba final Actividad presencial

Nota.- El cronograma sigue una planificación teórica de la asignatura que puede sufrir modificaciones durante el curso.

Nota 2.- Para poder calcular correctamente la dedicación de un alumno, la duración de las actividades que se repiten en el tiempo (por ejemplo, subgrupos de prácticas") únicamente se indican la primera vez que se definen.

Actividades de Evaluación

Semana	Descripción	Duración	Tipo evaluación	Técnica evaluativa	Presencial	Peso	Nota mínima	Competencias evaluadas
11	Examen parcial 1	02:00	Evaluación continua	EX: Técnica del tipo Examen Escrito	Sí	50%	4 / 10	CG2, CECT1, CECT2, CG9, CECT7
17	Examen parcial 2	02:00	Evaluación continua	EX: Técnica del tipo Examen Escrito	Sí	40%	4 / 10	CECT1, CECT2, CECT7, CG2, CG9
17	Examen final	03:00	Evaluación sólo prueba final	EX: Técnica del tipo Examen Escrito	Sí	100%	5 / 10	CECT1, CECT2, CECT7, CG2, CG9

Criterios de Evaluación

En convocatoria ordinaria, los alumnos serán evaluados normalmente mediante evaluación continua. No obstante, en cumplimiento de la Normativa de Evaluación de la Universidad Politécnica de Madrid, los alumnos que lo deseen serán evaluados mediante un único examen final, siempre y cuando lo comuniquen al Director del Departamento de Ingeniería de Sistemas Telemáticos mediante solicitud presentada en el registro de la Escuela Técnica Superior de Ingenieros de Telecomunicación antes del día 10 de abril de 2015. Esta opción supone la renuncia a la evaluación continua.

En la convocatoria extraordinaria la evaluación se realizará mediante un único examen final.

La **evaluación continua** se realizará de la siguiente manera: la asignatura se aprobará cuando se obtenga una calificación mayor o igual a 5 puntos sobre un total de 10, según las normas que se indican en este apartado, siempre que se hayan realizado todas las actividades de laboratorio y los dos proyectos de programación que forman parte del trabajo de la asignatura.

La nota final se obtendrá mediante la suma ponderada de las calificaciones correspondientes a las diferentes actividades de evaluación, con los siguientes pesos:

- Examen parcial 1 (temas1-3 y proyecto 1): 50%
- Examen parcial 2 (tema 4 y proyecto 2): 40 %
- Laboratorios y actividades adicionales en clase, a criterio del profesor: 10 %

En cualquier caso será necesario superar el umbral de 4 puntos sobre 10 en cada uno de estos bloques de evaluación para aprobar la asignatura. Los proyectos de programación se evaluarán mediante ejercicios escritos en los dos exámenes parciales.

Los alumnos que opten por la evaluación mediante un único examen final deberán realizar también los proyectos de programación propuestos durante el curso, que se evaluarán mediante ejercicios en el examen final. El examen final de la convocatoria extraordinaria también incluirá ejercicios sobre los trabajos prácticos.

Todos los exámenes y trabajos que se realicen deben ser fruto del trabajo personal del alumno, aunque se fomentará la discusión y el trabajo en grupo para ayudar a entender mejor los problemas que se intentan resolver. La copia de exámenes o trabajos prácticos supondrá el suspenso de la asignatura de forma automática, tanto para quien copia como para quien se deja copiar.

En el caso de que en el desarrollo de las pruebas de evaluación se aprecie el incumplimiento de los deberes como estudiante universitario, el coordinador de la asignatura podrá ponerlo en conocimiento del Director o Decano del Centro, que de acuerdo con lo establecido en el artículo 74 (n) de los Estatutos de la UPM tiene competencias para "Proponer la iniciación del procedimiento disciplinario a cualquier miembro de la Escuela o Facultad, por propia iniciativa o a instancia de la Comisión de Gobierno" al Rector, en los términos previstos en los estatutos y normas de aplicación.

Recursos Didácticos

Descripción	Tipo	Observaciones
Portal de la asignatura	Recursos web	Portal de la asignatura en moodle con información, copias de transparencias y enlaces para entregar las actividades prácticas.
Libros de referencia	Bibliografía	K. Sierra, Head first Java M.A. Weiss, Data structures & Problem Solving using Java Wong, Java Threads M. Gargenta, Learning Android
Laboratorios docentes del DIT	Equipamiento	Laboratorios A-127 y B-123
Información complementaria	Recursos web	Tutoriales, herramientas y almacenes de software accesibles a través del portal de la asignatura.

Otra Información

Además de la asistencia a las clases en el aula y en el laboratorio, los alumnos deben completar las prácticas de laboratorio y realizar dos proyectos de programación, que deberán entregar en las fechas que se señalen. Se darán instrucciones para la realización de estas actividades en el portal de la asignatura.