

Plan 512 GRADO EN INGENIERÍA DE TECNOLOGÍAS ESPECÍFICAS DE TELECOMUNICACIÓN

Asignatura 46618 INGENIERIA DE SISTEMAS DE SOFTWARE

Tipo de asignatura (básica, obligatoria u optativa)

Obligatoria

Créditos ECTS

6 ECTS

Competencias que contribuye a desarrollar

Generales

- GB1. Capacidad de razonamiento, análisis y síntesis.
- GB2. Capacidad para relacionar conceptos y adquirir una visión integrada, evitando enfoques fragmentarios.
- GB3. Capacidad de toma de decisiones en la resolución de problemas básicos de ingeniería de telecomunicación, así como identificación y formulación de los mismos.
- GB4 Capacidad para trabajar en grupo, participando de forma activa, colaborando con sus compañeros y trabajando de forma orientada al resultado conjunto, y en un entorno multilingüe.
- GB5. Conocimiento de materias básicas, científicas y tecnologías, que le capacite para el aprendizaje de nuevos métodos y tecnologías.
- GB6. Conocimientos para la realización de mediciones, cálculos, valoraciones, tasaciones, peritaciones, estudios, informes, planificación de tareas y otros trabajos análogos en el ámbito de la ingeniería técnica de Telecomunicación.
- GBE1. Capacidad para manejar especificaciones, reglamentos y normas de obligado cumplimiento.
- GBE3. Capacidad para resolver problemas con iniciativa, creatividad y razonamiento crítico.
- GC1. Capacidad de organización, planificación y gestión del tiempo.
- GC2. Capacidad para comunicar, tanto por escrito como de forma oral, conocimientos, procedimientos, resultados e ideas relacionadas con las telecomunicaciones y la electrónica.
- GC3. 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.

Específicas

- T1. 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.
- T2. Capacidad de utilizar aplicaciones 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.
- T6. Capacidad de concebir, desplegar, organizar y gestionar redes, sistemas, servicios e infraestructuras de telecomunicación en contextos residenciales (hogar, ciudad y comunidades digitales), empresariales o institucionales responsabilizándose de su puesta en marcha y mejora continua, así como conocer su impacto económico y social.
- T7. Conocimiento y utilización de los fundamentos de la programación en redes, sistemas y servicios de telecomunicación.

Objetivos/Resultados de aprendizaje

Al finalizar la asignatura el alumno deberá ser capaz de:

- Describir el Proceso Unificado de Desarrollo de Software, así como otros modelos de procesos más clásicos.
- Producir la documentación relativa a diversos entregables vinculados a las diferentes Fases y Actividades del Proceso Unificado de Desarrollo de Software, como por ejemplo el documento de Especificación de Requisitos del Software (SRS, Software Requirements Specification), relativos a un sistema software determinado, con el rigor y nivel de calidad adecuados, puesto que el contenido de dichos documentos deberá ser asumido y entendido tanto por el cliente/usuario como por el ingeniero de software. Asimismo, optimizar el esfuerzo de producir dichos entregables y

---

enfatar su utilidad para las siguientes Fases y Actividades.

- Modelar un sistema software concreto haciendo uso del Lenguaje de Modelado Unificado como notación fundamental vinculada a la metodología utilizada en la Ingeniería del Software, a partir de la comprensión de un dominio/negocio concreto y de la captura de Requisitos de un sistema software real mediante la comunicación con un agente especialista en dicho dominio/negocio en un tiempo limitado.
  - Construir diferentes artefactos UML como el Diagrama de Casos de Uso con la ayuda de herramientas informáticas de apoyo a la Ingeniería Software, es decir, con las denominadas herramientas CASE, relativos a un sistema software determinado.
  - Conocer los principales conceptos y propiedades vinculados al paradigma de la Programación Orientada a Objetos como paradigma de programación vinculado a la Ingeniería de Software moderna.
  - Codificar un sistema software concreto haciendo uso de un lenguaje vinculado a la programación orientada a objetos como es el lenguaje de programación Java, tomando como base entregables de la actividad de Diseño, que capturen tanto la vista estática como los aspectos dinámicos del sistema, previamente generados.
  - Aprender la rentabilidad de trabajar haciendo uso de la metodología propuesta por la Ingeniería de Software frente a un enfoque no ingenieril y caótico que se centre en la elaboración de código, prescindiendo de las actividades de Análisis y Diseño.
  - Aprender de forma autónoma siendo capaz de buscar información técnica relevante para una tarea especificada.
  - Autoevaluar el trabajo propio realizado, identificando sus errores y aspectos a mejorar.
  - Trabajar en equipo:
    - o Intercambiar información a través del correo electrónico, foros de discusión y otras herramientas de comunicación síncrona y/o asíncrona.
    - o Explicar a un equipo el resultado de una tarea realizada y asegurarse de que todos los demás lo han comprendido.
    - o 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.
    - o 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 informes de carácter técnico.
- 

## Contenidos

### TEMA 1: Introducción a la Ingeniería de Software

- 1.1 Conceptos generales
- 1.2 El proceso del software
- 1.3 El Proceso Unificado de Desarrollo de Software
- 1.4 El Lenguaje de Modelado Unificado (UML)
- 1.5 Herramientas CASE

### TEMA 2: Visión, Requisitos y Glosario

- 2.1 Visión: Descripción inicial del Sistema y Prototipo de la Interfaz de Usuario
- 2.2 Especificación de Requisitos
- 2.3 Glosario del proyecto

### TEMA 3: Los Casos de Uso

- 3.1 Motivación de los Casos de Uso
- 3.2 Notación en UML de los Casos de Uso
- 3.3 Buenas prácticas en la elaboración de los Casos de Uso

### TEMA 4: Modelos de Dominio

- 4.1 Introducción a la Orientación a Objetos
- 4.2 Notación en UML de los Diagramas de Clases
- 4.3 Motivación de los Modelos de Dominio
- 4.4 Buenas prácticas en la elaboración de los Modelos de Dominio

### TEMA 5: Los Diagramas de Secuencia del Sistema y los Contratos

- 5.1 Diagramas de Secuencia del Sistema
- 5.2 Notación en UML de los Diagramas de Secuencia del Sistema
- 5.3 Buenas prácticas en la elaboración de los Diagramas de Secuencia del Sistema
- 5.4 Contratos de las Operaciones del Sistema
- 5.5 Buenas prácticas en la elaboración de los Contratos de las Operaciones del Sistema

### TEMA 6: Los Diagramas de Interacción

- 6.1 Diagramas de Interacción
- 6.2 Notación en UML de los Diagramas de Interacción
- 6.3 Buenas prácticas en la elaboración de los Diagramas de Interacción

### TEMA 7: El Diseño

- 7.1 Consideraciones en el Diseño
  - 7.2 Patrones de Diseño
-

## TEMA 8: La Implementación

### 8.1 Consideraciones en la Implementación

### 8.2 Introducción al lenguaje de programación Java

## TEMA 9: Aspectos complementarios de Ingeniería de Software

### 9.1 Documentación de diseños mediante ingeniería inversa

### 9.2 Introducción a los métodos ágiles

Nota: Los contenidos de este tema pueden adaptarse dinámicamente en función del tiempo disponible y de la posibilidad de contar con expertos de empresas de software que puedan compartir su experiencia.

## PRÁCTICAS

- El trabajo en el laboratorio se desarrollará en torno a un proyecto, sobre el cual habrá que ir generando diferentes entregables correspondientes a las actividades de Requisitos, Análisis, Diseño e Implementación propuestas por el Proceso Unificado de Desarrollo de Software.
- Las entregas concretas y su planificación se detallarán en el Anexo I mencionado en la guía, que se facilitará al comienzo de la asignatura.

## Principios Metodológicos/Métodos Docentes

- Clase magistral participativa.
- Estudio de casos en aula y en laboratorio.
- Aprendizaje colaborativo.
- Método de proyectos.

## Criterios y sistemas de evaluación

La evaluación de la adquisición de competencias se basará en:

- Los informes realizados por los alumnos sobre el proyecto en el que se trabaje en la asignatura.
- La prueba escrita de evaluación al finalizar la asignatura.  
Sistema de calificaciones – Tabla resumen

### INSTRUMENTO/PROCEDIMIENTO

### PESO EN LA NOTA FINAL

### OBSERVACIONES

Valoración por el profesor de los entregables de equipo relativos al proyecto a desarrollar.

70%

Es condición necesaria (pero no suficiente) para superar la asignatura, el alcanzar una calificación igual o superior al 50% en esta parte.

Los entregables se agruparán en cuatro grupos de actividades (Análisis, Diseño, Implementación y Otros), siendo necesario obtener una calificación igual o superior al 20% de la nota máxima agregada en cada grupo de actividades. Las entregas concretas y su planificación se detallarán en el Anexo I mencionado en la guía, que se facilitará al comienzo de la asignatura.

En caso de no realizarse alguno de los entregables propuestos la calificación final en la asignatura será de No Presentado (N.P.).

### Prueba escrita final.

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% de esta parte para que se haga la media ponderada con las otras calificaciones obtenidas 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 al menos 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 contemplan los siguientes escenarios:

1. Para el primer instrumento de la tabla (entregables correspondientes al proyecto de ingeniería de software):
  - a. Si en la convocatoria ordinaria se supera el 20% en cada grupo de entregables (Análisis, Diseño, Implementación y Otros) del proyecto, y el 50% en la suma de todos los entregables, se conserva la nota de este elemento para la convocatoria extraordinaria.
  - b. Si en la convocatoria ordinaria no se supera el 20% en la nota asignada en alguna de las cuatro categorías de entregables (Análisis, Diseño, Implementación y Otros) del proyecto, el alumno debe realizar nuevos entregables, sólo de las categorías en las que no ha superado el 20%, a partir de enunciados que se publicarán el día de la revisión de notas de la convocatoria ordinaria. La nota del resto de las categorías de entregables se mantiene para la convocatoria extraordinaria.
  - c. Si, aun habiendo superado el 20% en cada categoría de entregables (Análisis, Diseño, Implementación y Otros) del proyecto, no se supera el 50% en la suma de todos los entregables, los alumnos deben realizar nuevos entregables, sólo para las categorías de tareas en las que no hayan superado el 50%, a partir de enunciados que se publicarán el día de la revisión de notas de la convocatoria ordinaria. La nota del resto de las categorías de entregables se mantiene para la convocatoria extraordinaria.
  - d. A petición del interesado, el alumno puede renunciar a la calificación del proyecto (primer instrumento de la tabla) y realizar una serie de entregables a partir de enunciados que se publicarán el día de la revisión de notas de la convocatoria extraordinaria. Dicha renuncia debe hacerla por escrito hasta esa misma fecha. En ese caso, el alumno deberá repetir los entregables de las categorías a cuya nota ha renunciado.
2. Para el segundo instrumento de la tabla (prueba escrita):
  - a. Si en la convocatoria ordinaria se obtiene al menos el 50% de la nota de la prueba escrita final, se conserva la nota de este elemento para la convocatoria extraordinaria.
  - b. Si en la convocatoria ordinaria no se supera el 50% de la nota de la prueba escrita final, se debe repetir la prueba escrita en la convocatoria extraordinaria.

Además deberá tenerse en cuenta que, al igual que en la convocatoria ordinaria, si un alumno no alcanza los requisitos mínimos establecidos, su calificación final en la asignatura será el mínimo entre el valor calculado según la ponderación descrita y 4,5.

## Recursos de aprendizaje y apoyo tutorial

Horario de tutorías:

GRADO EN INGENIERÍA DE TECNOLOGÍAS ESPECÍFICAS DE TELECOMUNICACIÓN:

<http://www.uva.es/export/sites/uva/2.docencia/2.01.grados/2.01.02.ofertaformativagrados/2.01.02.01.alfabetica/Grado-en-Ingenieria-de-Tecnologias-Especificas-de-Telecomunicacion/>

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 una herramienta CASE que permita construir artefactos UML y con el IDE Eclipse.
- Documentación de apoyo.
- Acceso al material bibliográfico complementario recomendado en la biblioteca del Campus o mediante otras vías como Internet.

## Calendario y horario

Véase calendario académico UVA

Ver horarios titulación

## Tabla de Dedicación del Estudiante a la Asignatura/Plan de Trabajo

Dedicación del Estudiante:

ACTIVIDADES PRESENCIALES

HORAS

ACTIVIDADES NO PRESENCIALES

HORAS

Clases teóricas

20

Estudio y trabajo autónomo individual

45

Clases prácticas

---

Estudio y trabajo autónomo grupal

45

Laboratorios

40

Prácticas externas, clínicas o de campo

Seminarios

Otras actividades

Total presencial

60

Total no presencial

90

Plan de Trabajo:

El Plan de Trabajo particularizado para cada grupo estará disponible en el Campus Virtual antes de comenzar la asignatura.

---

Responsable de la docencia (recomendable que se incluya información de contacto y breve CV en el que aparezcan sus líneas de investigación y alguna publicación relevante)

María Ángeles Pérez Juárez (coordinadora de la materia de Fundamentos de Sistemas Software) - e-mail: mperez@tel.uva.es

Guillermo Vega Gorgojo - e-mail: guiveg@tel.uva.es

---

Idioma en que se imparte

Castellano

---