

**Proyecto/Guía docente de la asignatura**

Asignatura	Proceso de Desarrollo del Software		
Materia	Proceso de Desarrollo del Software		
Módulo			
Titulación	Programa de Estudios Conjunto de Grado en Ingeniería Informática de Servicios y Aplicaciones y de Grado en Matemáticas-INFOMAT (VA)		
Plan	5471	Código	40817
Periodo de impartición	Semestre 8	Tipo/Carácter	OB
Nivel/Ciclo	Grado	Curso	4
Créditos ECTS	6		
Lengua en que se imparte	Español		
Profesor/es responsable/s	Francisco José González Cabrera		
Datos de contacto (E-mail, teléfono...)	Escuela de Ingeniería Informática Plaza de la Universidad 1, 40005, Campus María Zambrano, Segovia Teléfono: 921 112 432 e-mail: fjgonzalez@infor.uva.es		
Departamento	Informática (ATC, CCIA, LSI)		

1. Situación / Sentido de la Asignatura**1.1 Contextualización**

La asignatura Proceso de Desarrollo del Software (del mismo nombre que la materia a la que pertenece) pretende ser una primera aproximación del alumno a los procesos que participan de la Ingeniería del Software desde el paradigma de la orientación a objetos. En este sentido, la Ingeniería del Software se tratará como un medio para atender a las necesidades de los supuestos clientes, procurándoles la solución software más adecuada a su problema, pero siempre, buscando la garantía de calidad del producto final entregado.

En cuanto al planteamiento de soluciones la asignatura Proceso de Desarrollo del Software se orienta hacia aquellos aspectos más relacionados con el diseño de la solución, dejando el Análisis detallado y la Ingeniería de Requisitos para la asignatura Modelado de Software que el alumno cursará en cursos posteriores.

Se enseñan al alumno tanto los Modelos de Desarrollo del Software, como el Ciclo de Vida de los sistemas software, desde dos perspectivas: metodologías clásicas y metodologías ágiles. Además, se estudian los distintos tipos de diagrama para modelar un sistema software utilizando para ello un lenguaje de descripción y modelado, UML.

Se busca, por tanto, como objetivo principal, dotar al alumno de una primera visión, generalista de los procesos de la ingeniería del software.



1.2 Relación con otras materias

La asignatura GPTI está integrada en la materia Proceso de Desarrollo del Software. Presenta relación con aquellas materias que tienen que ver con el desarrollo de Sistemas de Información para el almacenamiento, procesamiento y tratamiento de la información en las facetas: desarrollo (materia Proceso de Desarrollo del Software) y finalidad en el contexto de la organización (materia Sistemas de Información). Asimismo, constituye una herramienta básica en el desarrollo del Trabajo Fin de Grado.

Dentro de su propia materia (Proceso de Desarrollo del Software), guarda una estrecha relación con:

- Gestión de Proyectos basados en las Tecnologías de la Información (Semestre 9).

Está también fuertemente relacionada con la materia Sistemas de Información a través de la asignatura:

- Modelado Software de Sistemas de Información (Semestre 7).

1.3 Prerrequisitos

Para cursar con mayores garantías esta asignatura es conveniente haber cursado previamente las asignaturas Programación Orientada a Objetos (Semestre 3) y Sistemas de Bases de Datos (Semestre 7).

2. Competencias

2.1 Generales

- G02:** Conocimientos básicos de la profesión.
- G03:** Capacidad de análisis y síntesis.
- G04:** Capacidad de organizar y planificar.
- G05:** Comunicación oral y escrita en la lengua propia.
- G08:** Habilidades de gestión de la información.
- G09:** Resolución de problemas.
- G12:** Trabajo en equipo.
- G16:** Capacidad de aplicar los conocimientos en la práctica.
- G18:** Capacidad de aprender.
- G21:** Habilidad para trabajar de forma autónoma.
- G22:** Diseño y gestión de proyectos.

2.2 Específicas

- E03:** Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
- E07:** Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad,
- E13:** Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
- E20:** Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software.



E34: Capacidad para participar activamente en la especificación, diseño, implementación y mantenimiento de los sistemas de información y comunicación.

3. Objetivos

Los objetivos concretos que persigue la asignatura son:

- Mostrar al alumno una visión general del proceso de desarrollo del software.
- Formar a los alumnos en procesos de análisis y diseño de un sistema software (paradigma de tecnologías orientadas a objetos).
- Conocer, de manera detallada, los modelos de desarrollo del software y el ciclo de vida de sistemas software.
- Comprender el Proceso Unificado de Desarrollo del Software (RUP).
- Conocer y utilizar correctamente un lenguaje de descripción y modelado de sistemas software, UML (Lenguaje Unificado de Modelado).
- Estudiar, desarrollar y especificar el análisis del modelo estático de un sistema software.
- Comprender, desarrollar y especificar el análisis del comportamiento dinámico de un sistema software.
- Conocer los diferentes aspectos del diseño de un sistema software: diseño arquitectónico, arquitectura del sistema y arquitectura de las aplicaciones.
- Representar textual y gráficamente los resultados del proceso de desarrollo software.

Poniendo en relación los objetivos que persigue la asignatura con las competencias profesionales que debe alcanzar un Ingeniero en Informática, podemos establecer los siguientes objetivos:

- OBJ-1. Capacidad para concebir, redactar, organizar, planificar, desarrollar y firmar proyectos en el ámbito de la ingeniería en informática que tengan por objeto el desarrollo o la explotación de sistemas, servicios y aplicaciones informáticas.
- OBJ-2. Capacidad para dirigir las actividades objeto de los proyectos del ámbito de la informática.
- OBJ-3. Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones informáticas empleando los métodos de la ingeniería del software como instrumento para el aseguramiento de su calidad.
- OBJ-4. 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.
- OBJ-5. 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 en Informática.
- OBJ-6. Capacidad para analizar y valorar el impacto social y medioambiental de las soluciones técnicas, comprendiendo la responsabilidad ética y profesional de la actividad del Ingeniero en Informática.

Competencias Específicas vs Objetivos		OBJ-1	OBJ-2	OBJ-3	OBJ-4	OBJ-5	OBJ-6
Formación Básica	E03				X		



Competencias comunes a informática	E07			X			X
	E13			X			
	E20	X		X	X		
Sistemas de Información	E34		X			X	

Tabla Resumen Objetivos vs Competencias Específicas de la asignatura.

4. Contenidos y/o bloques temáticos

Bloque 1: “Introducción al Proceso de Desarrollo del Software”

Carga de trabajo en créditos ECTS: 2.4

a. Contextualización y justificación

El gran número de aplicaciones software con los que el futuro graduado en Ingeniería Informática de Servicios y Aplicaciones se encuentra y el número aún mucho mayor de procesos software con los que se enfrentará - para analizar, diseñar, implementar, actualizar, modificar y mantener - deben de llevar al alumno a reflexionar sobre el hecho de que sin una buena organización metodológica, común a todos los miembros de la profesión, esta tarea sería muy difícil de realizar.

Por esto, se enseña al alumno un enfoque metodológico bien organizado y estructurado, a través de los diferentes modelos de desarrollo de sistemas software, y de los ciclos de vida más adecuados para esos modelos. Así, el alumno dispondrá de los conocimientos y herramientas adecuadas para decantarse por un tipo de proceso de desarrollo u otro en función de las características de la aplicación que se plantea construir.

b. Objetivos de aprendizaje

- Conocer y saber aplicar, en cada caso el modelo de desarrollo software más apropiado según las características del proyecto.
- Conocer el ciclo de vida de un sistema software y las fases en las que se divide.
- Conocer el Proceso Unificado de Desarrollo del Software.
- Conocer y aplicar los modelos de desarrollo aplicados a las Metodologías Clásicas.
- Conocer y aplicar los modelos de desarrollo aplicados a las Metodologías Ágiles.

c. Contenidos

- Introducción a los Procesos de Desarrollo del Software: Ingeniería del software.
- El Proceso Unificado de Desarrollo del Software (RUP).
- Modelos de Desarrollo del Software y Ciclo de Vida de Sistemas Software (Metodologías Clásicas).
- Modelos de Desarrollo del Software y Ciclo de Vida de Sistemas Software (Metodologías Ágiles).

d. Métodos docentes

- Lección magistral: Exposición de los contenidos teóricos de la materia mediante la enseñanza basada en ejemplos, que se estudiarán en clase siguiendo un enfoque colaborativo.



- Estudio autónomo por parte del alumno, incluyendo realización de problemas, consulta bibliográfica y realización de prácticas y pruebas de evaluación.
- Trabajo en el aula con actividades encaminadas al aprendizaje cooperativo y colaborativo para el desarrollo de supuestos prácticos y estudio de casos.
- Realización de prácticas guiadas y libres de laboratorio mediante el trabajo en equipo.
- Método de proyectos y aprendizaje por tareas y exposición del trabajo realizado por parte del alumno.
- Trabajos Tutelados. Realización de un trabajo tutelado y (individual o grupal), en el que los alumnos deberán hacer uso de los conocimientos adquiridos y capacidad autónoma para su realización.
- Presentaciones/exposiciones. Presentación de los trabajos realizados por los alumnos.

e. Plan de trabajo

Los contenidos se desarrollarán al comienzo del curso (semanas 1-6), alternando las lecciones magistrales en el aula con supuestos teórico prácticos. Se dará la posibilidad de desarrollar durante este periodo un trabajo tutelado que consistirá en la investigación y posterior presentación, por parte de los alumnos de algunos aspectos concretos de los ciclos de vida clásicos o bien de los diferentes tipos de metodologías ágiles.

f. Evaluación

Seguimiento y evaluación (grupal e individual). A la finalización de este bloque se realizará una prueba escrita (control) de los contenidos aprendidos, con el fin de liberar materia de cara al examen final. Dicha prueba contendrá cuestiones breves y/o supuestos prácticos y su peso será de un 30% de la Nota Final de la asignatura. De no superar el control, el alumno se examinará en la convocatoria ordinaria/extraordinaria de los contenidos del bloque.

Sobre los contenidos de este bloque se posibilitará la realización de un trabajo tutelado que se evaluará en una sesión de presentación/exposición.

g. Bibliografía básica

- Todos los recursos docentes de la asignatura estarán disponibles en el Campus Virtual de la Universidad de Valladolid <http://campusvirtual.uva.es/>
- Además:
 - Sommerville, Ian. "Software Engineering" Spring, 2015 (10th Ed).
 - Pressman, Roger S. "Software Engineering: A Practitioner's Approach" MacGraw-Hill, 2015 (8th Ed).
 - Palacio, Juan. "Gestión de proyectos Scrum Manager", Scrum Manager, Versión 2.5 – 2014.
 - SWEBOOK. Software Engineering Body of Knowledge IEEE, 2013.
 - Garzas, Javier, Enríquez, Juan e Irrazábal, Enmanuel; "Gestión ágil de proyectos software", Ediciones Kybele Consulting, 2012
 - Álvarez, Alonso; De las Heras, Rafael y Lasa, Carmen; "Manual imprescindible de métodos ágiles", Ed. Anaya, 2012.

h. Bibliografía complementaria

**i. Recursos necesarios**

- Aula y laboratorio equipados con pizarra, ordenador con conexión a internet y cañón de proyección.
- Despacho o seminario adecuado para tutorías.

j. Temporalización

CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
2.4	Semanas 1 - 6

Bloque 2: “Lenguajes de descripción y modelado de sistemas software”Carga de trabajo en créditos ECTS: **a. Contextualización y justificación**

Una vez establecidas las bases de una metodología de desarrollo (Bloque I) el alumno deberá aprender el conocimiento y manejo de UML (Lenguaje Unificado de Modelado) como herramienta básica de modelado de un proyecto de desarrollo de software. Se busca así, representar textual y gráficamente, a través de modelos y diagramas, los resultados de los procesos de análisis y diseño de soluciones software.

b. Objetivos de aprendizaje

- Conocer y utilizar lenguajes de descripción y modelado de sistemas software (UML)..
- Representar textual y gráficamente los resultados del proceso de desarrollo software. Especificación de sistemas mediante UML.

c. Contenidos

- Introducción a UML (Lenguaje Unificado de Modelado).
- Reglas, Modelo Conceptual, Definición de Arquitectura, Vistas, ...
- Modelo de Casos de Uso: Descripción y construcción de casos de uso, relaciones entre casos de uso,...
- Modelado del Dominio: Diagramas de clases y objetos, jerarquía de clases, técnicas de identificación de clases y asociaciones, relaciones entre clases, ...
- Modelado del Comportamiento:
 - Modelado de la Interacción: Diagramas de interacción, mensajes síncronos y asíncronos, relación con los casos de uso (especificación de un caso de uso).
 - Modelado Dinámico: Diagramas de transición de estados, máquinas de estados, diagramas de actividades, ...
- Modelado de la Implementación: Diagramas de componentes y despliegue, ...

d. Métodos docentes

- Lección magistral: Exposición de los contenidos teóricos de la materia mediante la enseñanza basada en ejemplos, que se estudiarán en clase siguiendo un enfoque colaborativo.



- Estudio autónomo por parte del alumno, incluyendo realización de problemas, consulta bibliográfica y realización de prácticas y pruebas de evaluación.
- Trabajo en el aula con actividades encaminadas al aprendizaje cooperativo y colaborativo para el desarrollo de supuestos prácticos y estudio de casos.
- Realización de prácticas guiadas y libres de laboratorio mediante el trabajo en equipo.
- Método de proyectos y aprendizaje por tareas y exposición del trabajo realizado por parte del alumno.
- Trabajos Tutelados. Realización de un trabajo tutelado y (individual o grupal), en el que los alumnos deberán hacer uso de los conocimientos adquiridos y capacidad autónoma para su realización.
- Presentaciones/exposiciones. Presentación de los trabajos realizados por los alumnos.

e. Plan de trabajo

Los contenidos se desarrollarán durante las semanas 7-15, dando un peso muy importante, en este bloque, a las sesiones de trabajo en el laboratorio y a la realización de prácticas y ejercicios guiados y libres.

f. Evaluación

Seguimiento y evaluación (grupal e individual). La evaluación de este bloque supondrá un 70% de la Nota Final de la asignatura. El bloque se evaluará de la siguiente manera:

- Teoría: A la finalización del bloque se realizará una prueba escrita (control) de los contenidos aprendidos, con el fin de liberar materia de cara al examen final. Dicha prueba contendrá problemas y supuestos prácticos y su peso será de un 40% de la Nota Final de la asignatura. De no superar el control, el alumno se examinará en la convocatoria ordinaria/extraordinaria de los contenidos del bloque.
- Laboratorio: El alumno (individualmente o en grupo dependiendo del número de alumnos matriculados en la asignatura) desarrollará un Proyecto Final Práctico, propuesto por el profesor, que una vez finalizado deberá presentar y defender ante la clase. El peso del proyecto y su exposición supondrán un 30% de la Nota Final de la asignatura.

g. Bibliografía básica

- Todos los recursos docentes de la asignatura estarán disponibles en el Campus Virtual de la Universidad de Valladolid <http://campusvirtual.uva.es/>
- Además:
 - Sommerville, Ian. "Software Engineering" Spring, 2015 (10th Ed).
 - Pressman, Roger S. "software Engineering: A Practitioner's Approach" MacGraw-Hill, 2015 (8th Ed).
 - Joyanes y Zahonero. "Programación en C, C++, Java y UML" MacGraw-Hill, 2010.

h. Bibliografía complementaria

- Booch, Rumbaugh y Jacobson. "El Lenguaje Unificado de Modelado. Manual de referencia", Pearson-Addison Wesley, 2007 (2ª Edición).
- Booch, Rumbaugh y Jacobson. "El Lenguaje Unificado de Modelado. Guía del Usuario", 2006 (2ª Edición).



- Larman, C., "UML y Patrones. Una introducción al análisis y diseño orientado a objetos y al proceso unificado", 2006 (2ª Edición).

i. Recursos necesarios

- Aula y laboratorio equipados con pizarra, ordenador con conexión a internet y cañón de proyección.
- Laboratorio equipado con ordenadores y el software adecuado para la realización de las prácticas.
- Despacho o seminario adecuado para tutorías.

j. Temporalización

CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
3.6	Semanas 7 - 15

5. Métodos docentes y principios metodológicos

- Lección magistral: Exposición de los contenidos teóricos de la materia mediante la enseñanza basada en ejemplos, que se estudiarán en clase siguiendo un enfoque colaborativo.
- Estudio autónomo por parte del alumno, incluyendo realización de problemas, consulta bibliográfica y realización de prácticas y pruebas de evaluación.
- Trabajo en el aula con actividades encaminadas al aprendizaje cooperativo y colaborativo para el desarrollo de supuestos prácticos y estudio de casos.
- Realización de prácticas guiadas y libres de laboratorio mediante el trabajo en equipo.
- Método de proyectos y aprendizaje por tareas y exposición del trabajo realizado por parte del alumno.
- Trabajos Tutelados. Realización de un trabajo tutelado y (individual o grupal), en el que los alumnos deberán hacer uso de los conocimientos adquiridos y capacidad autónoma para su realización.
- Presentaciones/exposiciones. Presentación de los trabajos realizados por los alumnos.

6. Tabla de dedicación del estudiante a la asignatura

ACTIVIDADES PRESENCIALES	HORAS	ACTIVIDADES NO PRESENCIALES	HORAS
Clases teórico-prácticas (T/M)	30	Estudio y trabajo autónomo individual (conocer, comprender, plantear dudas, experimentar, etc.)	52
Laboratorios (L)	20	Estudio y trabajo autónomo grupal (preparación de prácticas)	30
Trabajos tutelados (TG)	2	Desarrollo trabajos tutelados	4
Presentaciones/Exposiciones	4	Preparación presentaciones	4
Evaluación	4		
Total presencial	60	Total no presencial	90

7. Sistema y características de la evaluación



INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Teoría, Evaluación Continua, TC1 y TC2: Realización de dos controles de carácter teórico-práctico.	70% (30% + 40%)	Se realizarán dos controles de clase para evaluar los conocimientos de los alumnos, el primero en el mes de marzo (TC1: 30% de peso respecto de la nota final de la asignatura) y el segundo a la finalización del periodo docente (TC2: 40% de peso respecto de la nota final de la asignatura). Serán de carácter voluntario y liberarán materia de cara al Examen de la Convocatoria Ordinaria (ECO) de la asignatura. Se exigirá un mínimo de 5/10 de la puntuación en TC1 y TC2 para poder hacer media y superar la Teoría.
Teoría, Convocatoria Ordinaria, ECO: Realización de examen final escrito de carácter teórico-práctico (cuestiones breves y problemas).	70%	Se realizará un examen final para evaluar los conocimientos de los alumnos. A este examen deberán presentarse los alumnos que no hayan superado la teoría en la Evaluación Continua y abarcará todos los contenidos de la asignatura. En el supuesto de que un alumno hubiera superado uno sólo de los dos controles, podrá optar, por examinarse tan sólo de los contenidos del control no superado. Se exigirá un mínimo de 5/10 de la puntuación para hacer media con la práctica.
Teoría, Convocatoria Extraordinaria, ECE: Realización de examen final escrito de carácter teórico-práctico (cuestiones breves y problemas).	70%	Se realizará un examen final para evaluar los conocimientos de los alumnos. A este examen deberán presentarse todos los alumnos que no hayan superado la teoría en la Evaluación Continua o en la Convocatoria Ordinaria y abarcará todos los contenidos de la asignatura. Se exigirá un mínimo de 5/10 de la puntuación para hacer media con la práctica.
Prácticas, Realización y Defensa de un Proyecto Final Práctico (PFP).	30%	Para superar la parte práctica de la asignatura se realizará y defenderá un Proyecto Final Práctico que se desarrollará a lo largo del periodo de docencia (coincidiendo con el Bloque II). Por tanto, este proyecto deberá de ser realizado por todos los alumnos (individualmente o en grupo dependiendo del número de alumnos matriculados en la asignatura). A la finalización del periodo docente se realizará un ejercicio de exposición y defensa del mismo. La calificación del proyecto será del 25% respecto de la nota final de la asignatura y la calificación del ejercicio de defensa será del 5% respecto de la nota final de la asignatura. Se exigirá un mínimo del 5/10 de la puntuación en el apartado de prácticas para poder hacer media con la parte teórica. Para puntuar en este apartado es obligatorio que el alumno exponga y defienda su proyecto.
Proyectos y Trabajos Voluntarios (VT).	10%	De manera voluntaria y siempre que el desarrollo de las clases y el número de alumnos matriculados en la asignatura lo permita, se propondrá la realización de trabajos voluntarios (individualmente o en grupo). Se evaluarán tanto la claridad de presentación como la calidad del mismo.



CRITERIOS DE CALIFICACIÓN

- **Convocatoria ordinaria:**
 - Descritos en la tabla anterior.
- **Convocatoria extraordinaria:**
 - Descritos en la tabla anterior.

8. Consideraciones finales

- **Consultas y Tutorías:**
 - Horario: Lunes de 9:00 a 11:00 horas y de 13:00 a 14:00 horas y martes de 9:00 a 12:00 horas.
 - Lugar: Despacho 11 (Sede de Santa Eulalia) / Tutoría A-103 (Campus María Zambrano).
- **Horario de clases:**
 - Teoría: Lunes de 18:00 a 20:00 horas (Aula A-102, Campus María Zambrano).
 - Prácticas*: Martes de 18:00 a 20:00 horas (Laboratorio Li-006, Campus María Zambrano).
 - (*) Nota: En caso de necesidad de desdoble (por el número de alumnos matriculados en la asignatura), habrá un segundo turno de prácticas que será: Lunes de 11:00 a 13:00 horas (Laboratorio Li-006, Campus María Zambrano).
- **Exámenes:**
 - Ordinario: Jueves 11 de Junio de 2020, 11:30 h. (Aula 3, Santa Eulalia).
 - Extraordinario: Jueves 2 de Julio de 2020, 11:30 h. (Aula 3, Santa Eulalia).