

**Guía docente de la asignatura**

Asignatura	PLANIFICACIÓN Y GESTIÓN DE PROYECTOS		
Materia	DESARROLLO DE SOFTWARE		
Módulo			
Titulación	GRADO EN INGENIERÍA INFORMÁTICA (463)		
Plan	463	Código	45210
Periodo de impartición	1 ^{er} . CUATRIMESTRE	Tipo/Carácter	OBLIGATORIA
Nivel/Ciclo	CURSO DE ADAPTACIÓN	Curso	-
Créditos ECTS	6 ECTS		
Lengua en que se imparte	CASTELLANO		
Profesor/es responsable/s	Pablo de la Fuente Redondo		
Datos de contacto (E-mail, teléfono...)	TELÉFONO: 983 185611 E-MAIL: pfuente@infor.uva.es		
Horario de tutorías	Véase www.uva.es → Centros → Campus de Valladolid → Escuela Técnica Superior de Ingeniería Informática → Tutorías		
Departamento	Informática (ATC, CCIA, LSI)		



1. Situación / Sentido de la Asignatura

1.1 Contextualización

Esta asignatura está integrada en el cuarto curso del grado de ingeniería informática. La asignatura tiene como objetivo presentar los conocimientos precisos para realizar la gestión de un proyecto de desarrollo de software. Para ello, seguirá las pautas proporcionadas por el Instituto de Gestión de Proyectos (PMI) y algunas indicadas por la AEIPRO (Asociación Española de Ingeniería de Proyectos), considerando la naturaleza particular del software. También se utilizará como referencia lo indicado para la gestión de proyectos en el SWEBOK (Software Engineering Book Of Knowledge). El objetivo fundamental es dotar al alumno de las competencias básicas para poder llevar a cabo la tarea de gestión de proyectos software.

1.2 Relación con otras materias

La planificación y gestión de proyectos es un área de las definidas en el Cuerpo de conocimiento de Ingeniería de software (SWEBOK). Está relacionada, por lo tanto, con las asignaturas de la materia Ingeniería de software.

1.3 Prerrequisitos

Se supone que el alumno tiene conocimiento de los conceptos fundamentales de la ingeniería y del proceso de desarrollo del software.

PROVISIONAL

2. Competencias

2.1 Generales

Código	Descripción
G03	Capacidad de análisis y síntesis
G04	Capacidad de Organizar y Planificar
G05	Comunicación oral y escrita en la lengua propia
G06	Conocimiento de una segunda lengua (preferentemente inglés)
G08	Habilidades de gestión de la información
G09	Resolución de problemas
G10	Toma de decisiones
G11	Capacidad crítica y autocrítica
G12	Trabajo en equipo
G14	Responsabilidad y compromiso ético
G15	Liderazgo
G16	Capacidad de aplicar los conocimientos en la práctica
G17	Habilidades de investigación
G18	Capacidad de aprender
G19	Capacidad de adaptarse a nuevas situaciones
G20	Capacidad de generar nuevas ideas
G21	Habilidad para trabajar de forma autónoma
G22	Diseño y gestión de proyectos

2.2 Específicas

Código	Descripción
IS1	Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la Ingeniería del Software.
IS2	Capacidad para valorar las necesidades del cliente y especificar los requisitos software para satisfacer estas necesidades, reconciliando objetivos en conflicto mediante la búsqueda de compromisos aceptables dentro de las limitaciones derivadas del coste, del tiempo, de la existencia de sistemas ya desarrollados y de las propias organizaciones.
IS5	Capacidad de identificar, evaluar y gestionar los riesgos potenciales asociados que pudieran presentarse.
IS6	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.
CI2	Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.
CI3	Capacidad para comprender la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software.
CI4	Capacidad para elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.



3. Objetivos

Código	Descripción
R01	Reconocer y valorar la importancia y necesidad de la gestión de proyectos.
R02	Utilizar herramientas de soporte a la gestión de proyectos.
R03	Conocer las funciones más importantes de un responsable de proyecto.
R04	Estimar las medidas de progreso de un proyecto informático, de productividad y de costes asociados.
R05	Identificar y analizar los riesgos en el proceso de desarrollo de software.
R06	Planificar y gestionar algún proyecto concreto como caso de estudio, señalando etapas, asignando costes, etc.

4. Tabla de dedicación del estudiante a la asignatura

ACTIVIDADES PRESENCIALES	HORAS	ACTIVIDADES NO PRESENCIALES	HORAS
Clases teórico-prácticas (T/M)	28	Estudio y trabajo autónomo individual	40
Clases prácticas de aula (A)		Estudio y trabajo autónomo grupal	50
Laboratorios (L)	24		
Prácticas externas, clínicas o de campo			
Seminarios (S)	6		
Tutorías grupales (TG)			
Evaluación (fuera del periodo oficial de exámenes)	2		
Total presencial	60	Total no presencial	90

5. Bloques temáticos

Bloque 1: Gestión de proyectos de software. Introducción.

Carga de trabajo en créditos ECTS:

a. Contextualización y justificación

Como se ha indicado anteriormente, la Planificación y Gestión de Proyectos (se supone de software), esta considerada en distintos puntos del SWEBOK. Por otra parte el PMBOK (cuerpo de conocimiento de gestión de proyectos) engloba todos los aspectos importantes en la realización de un proyecto de cualquier tipo. Este bloque pretende dar a conocer al alumno qué es un proyecto, qué lo caracteriza y la importancia que tienen la existencia de estándares en el desarrollo de software.

b. Objetivos de aprendizaje

- Conocer perfectamente las etapas de desarrollo de un proyecto
- Conocer las ideas generales de algunos estándares de Gestión de Proyectos, como PMBOK, los capítulos adecuados de SWEBOK, etc.

c. Contenidos

- Qué es un proyecto. Algunos tipos de proyectos
- Elementos a considerar en el desarrollo de un proyecto con incidencia en los proyectos de software.

d. Métodos docentes

En este primer bloque se utilizarán las clases magistrales para introducir los conceptos planteando a los alumnos cuestiones relativas a las razones del fracaso en el cumplimiento de las expectativas de tiempo y coste de los proyectos software.

En la clase de prácticas se les presentaran las características generales de un software de planificación de proyectos como MS Project.

e. Plan de trabajo

Ver Cronograma al final de la guía

f. Evaluación

En este bloque no se realiza ninguna prueba de evaluación.

g. Bibliografía básica

- Bob Hughes and Mike Cotterell. *Software Project Management*. McGraw Hill, 2002. ISBN 0 07 709834 X
- *A guide to Project Management Body of Knowledge*. PMI. Existe una versión en español de la tercera edición.



h. Bibliografía complementaria

- *Walter Royce. Software Project Management.* Addison-Wesley. 1998. ISBN 0-201-30958-0

i. Recursos necesarios

Se introducen los conceptos básicos de la práctica de gestión de proyectos. Se dispondrá de una herramienta de planificación como MS-Project o similar.

- Damith C. Rajapakse. *Tips for succeed in Software Engineering Students projects.*
<http://www.comp.nus.edu.sg/~damithch/guide/>

PROVISIONAL

**Bloque 2: Planificación y control de proyectos de software.**

Carga de trabajo en créditos ECTS: 1,9

a. Contextualización y justificación

Este bloque pretende dar a conocer al alumno cómo se debe realizar la gestión del tiempo y de los costes de un proyecto. Además plantear detalles de qué consideraciones hay que realizar a la hora de asignar recursos a las actividades de un proyecto. Por último se verá la importancia que tiene la gestión de los riesgos para la consecución con éxito de un proyecto.

b. Objetivos de aprendizaje

- Conocer las técnicas de planificación y seguimiento de proyectos
- Elaborar la planificación temporal y de costes de un proyecto de prueba
- Conocer la incidencia que los recursos humanos tienen en la planificación de un proyecto.
- Identificar y gestionar los riesgos en el desarrollo de un proyecto software
- Elaborar los aspectos correspondientes de un Plan de Desarrollo de Software

c. Contenidos

Planificación de proyectos
Gestión del tiempo
Gestión de costes
Gestión de riesgos
Seguimiento y control de la evolución de un proyecto
Plan de Desarrollo de Software.

d. Métodos docentes

Los conceptos básicos se seguirán presentando en clases magistrales planteando la participación del alumno en base a cuestiones puntuales relativas a los conceptos tratados.

En la parte práctica se continuará la presentación de las posibilidades de MS-Project, especialmente en los que atañe a la gestión de recursos.

Por último, dado que una parte de las prácticas es orientada a proyectos se definirán grupos de trabajo y se planteará el proyecto a realizar, del que habrá que elaborar un Plan de desarrollo.

e. Plan de trabajo

Ver Cronograma al final de la guía

f. Evaluación

En este bloque se realizará una prueba tipo test (elección entre varias respuestas) para valorar los conocimientos adquiridos en los Bloques 1 y 2 (Introducción y Planificación y Control de proyectos de Software)

g. Bibliografía básica

- Bob Hughes and Mike Cotterell. *Software Project Management*. McGraw Hill, 2002. ISBN 0 07 709834 X

- *A guide to Project Management Body of Knowledge*. PMI. Existe una versión en español de la tercera edición.

h. Bibliografía complementaria

- Elaine M. Hall. *Managing Risk*. Addison-Wesley. 1998. ISBN 0201255928

- *Walter Royce. Software Project Management*. Addison-Wesley. 1998. ISBN 0-201-30958-0



i. Recursos necesarios

MS-Project o herramienta similar

- Damith C. Rajapakse. *Tips for succeed in Software Engineering Students projects.*
<http://www.comp.nus.edu.sg/~damithch/guide/>

PROVISIONAL

Bloque 3: Modelos de procesos de desarrollo de software.Carga de trabajo en créditos ECTS: **a. Contextualización y justificación**

Cuando se aborda un proyecto de desarrollo de software uno de los aspectos importantes es qué modelo de proceso se va a seguir. En este bloque se presentan los modelos de proceso de forma general, incidiendo sobre el Proceso Unificado y los procesos ágiles.

b. Objetivos de aprendizaje

- Conocer de los aspectos generales de un proceso de desarrollo general como el Proceso Unificado.
- Ser capaz de elaborar un plan de desarrollo siguiendo RUP
- Conocer algunos aspectos de métodos ágiles como SCRUM
-

c. Contenidos

- Proceso Unificado introducción y aspectos fundamentales
- Planificación en el Proceso Unificado
- SCRUM

d. Métodos docentes

Los conceptos básicos se seguirán presentando en clases magistrales planteando la participación del alumno en base a cuestiones puntuales relativas a los conceptos tratados.

Se planteará un seminario donde se analicen con detalle los aspectos e implicaciones del uso de los métodos ágiles desde el punto de vista de la planificación y el control de un proyecto software.

e. Plan de trabajo

Ver Cronograma al final de la guía

f. Evaluación

Se realizará una prueba tipo test para valorar los conocimientos adquiridos en este bloque.

g. Bibliografía básica

- Philippe Kruchten. *The Rational Unified Process. An introduction*. Second Edition. Addison-Wesley, 2000. ISBN 0201707101

h. Bibliografía complementaria

- Walter Royce. *Software Project Management*. Addison-Wesley. 1998. ISBN 0-201-30958-0
- Diversos materiales disponibles en la Web.

i. Recursos necesarios

Para el Proceso Unificado se utilizarán los materiales disponibles en www.upedu.org

Para SCRUM y otros métodos ágiles se usarán diversos materiales disponibles en la Web.

**Bloque 4: Gestión de Configuraciones de software.**Carga de trabajo en créditos ECTS: **a. Contextualización y justificación**

La gestión de configuraciones es un área de conocimiento de SWEBOK. Por otro lado, la utilización de al menos un sistema de control de versiones es fundamental en el desarrollo de software.

b. Objetivos de aprendizaje

- Entender la diferencia entre control de versiones y gestión de configuraciones
- Valorar la importancia que la gestión de configuraciones tiene en el éxito de los proyectos de software.
- Conocer los estándares de la IEEE correspondientes a gestión de configuraciones.
- Elaborar los aspectos correspondientes en un Plan de Desarrollo de Software

c. Contenidos

- Conceptos generales
- Operaciones de la gestión de configuraciones de software

d. Métodos docentes

Los conceptos básicos se seguirán presentando en clases magistrales planteando la participación del alumno en base a cuestiones puntuales relativas a los conceptos tratados.

En la parte práctica se exigirá que los grupos utilicen alguna herramienta de software libre para la gestión de configuraciones.

e. Plan de trabajo

Ver Cronograma al final de la guía

f. Evaluación

Se realizará una prueba tipo test para valorar los conocimientos adquiridos en este bloque.

g. Bibliografía básica

- S.A. Dart, *Concepts in Configuration Management Systems*, Proceedings of the third International Workshop on Software Configuration Management, ACM 1991, pag. 1-18
- Anne Mette Jonasses Hass. *Configuration Management. Principles and Practice*. Pearson Education 2003.. ISBN 0-321-11766-2

h. Bibliografía complementaria

- A guide to Project Management Body of Knowledge. PMI. Existe una versión en español de la tercera edición.
- Casi todos los libros clásicos de Ingeniería de software (Pressman, Sommerville, etc.) contienen un capítulo dedicado a gestión de configuraciones.

i. Recursos necesarios

Alguno de los paquetes de control de versiones como CVS, Subversión, Git, etc.

Bloque 5: Medidas y Calidad en procesos de desarrollo de software

Carga de trabajo en créditos ECTS: 1,7

a. Contextualización y justificación

La calidad es algo cada vez más demandado en el desarrollo de software. Pero para poder mejorar cualquier tipo de proceso debemos tener información sobre él. Las medidas son una parte cada vez más importante en cualquier proceso que aborde la obtención de un producto que cumpla unos requisitos de calidad.

b. Objetivos de aprendizaje

- Conocer la incidencia de la calidad en el desarrollo de un proyecto.
- Comprender la importancia de las medidas en la mejora de los procesos y de los productos de desarrollo de software.
- Conocer y elaborar un plan de medidas de acuerdo a las características de un proyecto y al objetivo del mismo.
- Conocer los niveles de CMMI u otros estándares y las implicaciones en la gestión de proyectos en dichos niveles.
-

c. Contenidos

Calidad. Introducción y aspectos generales.
Medidas de Calidad en Software
De proceso
De proyecto
De producto
Garantía de calidad. Ejemplo: CMMI, ISO/IEC 15504 (SPICE)

d. Métodos docentes

Los conceptos básicos se seguirán presentando en clases magistrales planteando la participación del alumno en base a cuestiones puntuales relativas a los conceptos tratados.

Se planteará un seminario donde se analicen con detalle los aspectos e implicaciones de intentar conseguir un nivel 3 en CMMI o SPICE.

e. Plan de trabajo

Ver Cronograma al final de la guía

f. Evaluación

Se realizará una prueba tipo test para valorar los conocimientos adquiridos en este bloque.

En el periodo de exámenes se realizará una prueba escrita y cada grupo deberá presentar su práctica.

g. Bibliografía básica

- Norman E. Fenton. Shari Lawrence Pfleeger. *Software Metrics: A rigorous & practical approach*. International Thomson Computer Press, 1996. ISBN 1-85032-275-9
- Stephen H. Kan. *Metrics and Models in Software Quality Engineering*. Second Edition. Addison- Wesley 2003. ISBN 0201729156

h. Bibliografía complementaria

- Roger S. Pressman. *Ingeniería de Software: Un enfoque práctico*. McGraw-Hill 2002. El capítulo 19 está dedicado a métricas y calidad de software.
- Ian Sommerville. *Ingeniería de Software*. Addison-Wesley, 2002. Los capítulos 23, 24 y 25 tratan de temas relacionados.
- Web del International Function Points Users Group: <http://www.ifpug.org/>
- Diversos materiales sobre CMMI de www.sei.cmu.edu



i. Recursos necesarios

Damith C. Rajapakse. *Tips for succeed in Software Engineering Students projects.*
<http://www.comp.nus.edu.sg/~damithch/guide/>

Diversos materiales de la Web.

PROVISIONAL

6. Temporalización (por bloques temáticos)

BLOQUE TEMÁTICO	CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
Bloque 1: Gestión de proyectos de software. Introducción.	0,6 ECTS	Semana 1 y parte de la 2
Bloque 2: Planificación y control de proyectos de desarrollo de software.	1,9 ECTS	Parte de la semana 2 hasta la semana 7
Bloque 3: Modelos de proceso de desarrollo de software.	0,9 ECTS	Semanas 8 y 9
Bloque 4: Gestión de Configuraciones de Software	0,9 ECTS	Semanas 10 y 11
Bloque 5: Medidas y Calidad en procesos de software	1,7 ECTS	De la semana 12 hasta la semana 15

7. Sistema de calificaciones – Tabla resumen

INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Valoración de la práctica	40%	<p>Será necesaria la defensa, con asistencia de todos los miembros del grupo de la práctica.</p> <p>Para poder considerar la nota obtenida en la parte práctica será necesario haber alcanzado, al menos, un 4,5 en la calificación de dicha prueba. En dicha calificación se tendrá en cuenta, tanto los aspectos de funcionalidad, como la calidad de la documentación aportada y la defensa realizada.</p>
Examen final escrito	40%	Para poder compensar se deberá obtener una nota de 4,5 o superior en la parte teórica. Se realizará en periodo de exámenes
Pruebas tipo test	10%	Se realizarán 4 pruebas a lo largo del desarrollo temporal de la asignatura
Valoración de las tareas, de la aportación en seminarios y tutorías	10%	

CRITERIOS DE CALIFICACIÓN

- **Convocatoria ordinaria:**
 - Los criterios se han indicado en el punto de sistema de calificaciones. Conviene recordar la necesidad de superar un 4,5 en el examen de teoría o de práctica para poder compensar la otra parte (práctica o teoría). En el caso en que algún alumno no superara un 4,5 en la parte de teoría o de práctica la nota que se reflejaría sería el mínimo de entre la obtenida utilizando la suma ponderada indicada antes ($\text{práctica} \cdot 0,4 + \text{examen final} \cdot 0,4 + \text{pruebas tipo test} \cdot 0,1 + \text{Valoración} \cdot 0,1$) y 4,0.
- **Convocatoria extraordinaria:**
 - Si la práctica ya está superada se realizará un examen de teoría con el mismo formato que en la convocatoria ordinaria. Si, por el contrario, la práctica no estuviera superada deberá volver a presentar la práctica (en una única entrega) y realizar un examen de teoría con el mismo formato que en la convocatoria ordinaria. En ambos casos las pruebas se considerarán con similar peso y fórmula de cálculo para la nota final que en la convocatoria ordinaria.

**8. Anexo: Métodos docentes**

Actividad	Metodología
Clase de teoría	<ul style="list-style-type: none">• Clase magistral participativa• Estudio de casos en aula
Clase práctica	<ul style="list-style-type: none">• Clase magistral participativa• Realización de un proyecto guiado por el profesor, que encargará y guiará el trabajo que se realizará en grupos (4 alumnos), siguiendo un enfoque colaborativo.
Seminarios	<ul style="list-style-type: none">• Talleres de aprendizaje
Tutoría	<ul style="list-style-type: none">• Evaluación de los contenidos teóricos y de los proyectos

PROVISIONAL