



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA

GUIA DOCENTE



CURSO 2015/2016

Grado en Ingeniería Informática

DATOS DE LA ASIGNATURA

Nombre:

Principios y Fundamentos de la Ingeniería del Software

Denominación en inglés:

Principles and Foundations of Software Engineering

Código:

606010207

Carácter:

Obligatorio

Horas:

	Totales	Presenciales	No presenciales
Trabajo estimado:	150	60	90

Créditos:

Grupos reducidos				
Grupos grandes	Aula estándar	Laboratorio	Prácticas de campo	Aula de informática
3	0	0	0	3

Departamentos:**Áreas de Conocimiento:**

Tecnologías de la Información	Lenguaje y Sistemas Informáticos
-------------------------------	----------------------------------

Curso:**Cuatrimestre:**

2º - Segundo	Segundo cuatrimestre
--------------	----------------------

DATOS DE LOS PROFESORES

Nombre:**E-Mail:****Teléfono:****Despacho:**

Polo Almohano, M ^a del Pilar	polo@dti.uhu.es	87386	nº 53 Edificio Torreumbría
Roldán Ruiz, Ana María	amroldan@uhu.es	8 7387	51 TU
*Manuel de la Villa Cordero	manuel.villa@dti.uhu.es	959217639	Edif. Torreumbría, Dcho. 41

*Profesor coordinador de la asignatura

Consultar los horarios de la asignatura

1. Descripción de contenidos

1.1. Breve descripción (en castellano):

El alumnado de una titulación de Grado en Informática, independientemente del itinerario que curse, debe conocer los principios de la Ingeniería del Software al desarrollar un proyecto software completo, desde la fase de especificación hasta la entrega al cliente. Es por tanto necesario acercar al alumno el ciclo de vida de un proyecto de software, así como la necesidad de llevar a cabo dicho ciclo para desarrollar un software con garantías. Así pues es fundamental que la formación del alumno abarque las siguientes temáticas:

- Concepto de Ingeniería del Software.
- Propiedades y ciclo de vida del producto software
- El proceso de desarrollo. Ingeniería de Requisitos.
- Diseño e implementación. Validación y verificación de software.
- Mantenimiento.

1.2. Breve descripción (en inglés):

The student in Computer Science degree, regardless of the specialty, ought to know the principles of software engineering to develop a complete software project, from specification stage through to delivery to the customer, and later maintenance. It is therefore necessary show the student, the life cycle of a software project, and the need to carry out this cycle to improve software guarantees. So it is essential that the student's training covers the following topics:

- Definition of Software Engineering.
- Properties and life cycle in software products.
- The development process. Requirements Engineering.
- Design and implementation. Validation and verification of software.
- Maintenance.

2. Situación de la asignatura

2.1. Contexto dentro de la titulación:

Esta asignatura sumerge al alumno en el ámbito de la Ingeniería del Software y de los proyectos de desarrollo de software, proporcionándole conocimientos básicos de análisis y modelado que le serán necesarios no sólo para posteriores asignaturas en el contexto de la Ingeniería del Software, sino también para llegar a ser profesionales capaces de elaborar siguiendo una disciplina (aplicando métodos y herramientas contrastados) un software de alta calidad en unos plazos razonables.

2.2. Recomendaciones:

Es la asignatura de introducción al mundo de la Ingeniería del Software. El alumno debe tener conocimientos básicos de diseño y programación orientada a objetos, adquiridos en asignaturas anteriores.

3. Objetivos (Expresados como resultados del aprendizaje):

Podemos considerar los siguientes objetivos de carácter general:

- Conocer las propiedades del software y su importancia en la sociedad.
- Reconocer la responsabilidad ética y profesional de un Ingeniero de Software.
- Conocer los elementos relacionados con el proceso de producción del software.
- Tomar conciencia de la necesidad de considerar la producción del software como un proceso de ingeniería y ser capaces de aplicar dicho enfoque de ingeniería en la construcción de un sistema software.

4. Competencias a adquirir por los estudiantes

4.1. Competencias específicas:

- **CC01:** Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.
- **CC05:** Conocimiento, administración y mantenimiento de sistemas, servicios y aplicaciones informáticas.
- **CC16:** Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software.
- **CC17:** Capacidad para diseñar y evaluar interfaces persona computador que garanticen la accesibilidad y usabilidad a los sistemas, servicios y aplicaciones informáticas.

4.2. Competencias básicas, generales o transversales:

- **CB1:** Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio
- **CG0:** Capacidad de análisis y síntesis: Encontrar, analizar, criticar (razonamiento crítico), relacionar, estructurar y sintetizar información proveniente de diversas fuentes, así como integrar ideas y conocimientos.
- **G01:** Capacidad de organización y planificación así como capacidad de gestión de la Información.
- **G02:** Capacidad de comunicación oral y escrita en el ámbito académico y profesional con especial énfasis, en la redacción de documentación técnica
- **G04:** Capacidad para tomar decisiones basadas en criterios objetivos (datos experimentales, científicos o de simulación disponibles) así como capacidad de argumentar y justificar lógicamente dichas decisiones, sabiendo aceptar otros puntos de vista
- **G05:** Capacidad de trabajo en equipo.
- **G07:** Motivación por la calidad y la mejora continua, actuando con rigor, responsabilidad y ética profesional.
- **G08:** Capacidad para adaptarse a las tecnologías y a los futuros entornos actualizando las competencias profesionales.
- **G09:** Capacidad para innovar y generar nuevas ideas.
- **T02:** Conocimiento y perfeccionamiento en el ámbito de las TIC's

5. Actividades Formativas y Metodologías Docentes

5.1. Actividades formativas:

- Sesiones de Teoría sobre los contenidos del Programa.
- Sesiones de Resolución de Problemas.
- Sesiones Prácticas en Laboratorios Especializados o en Aulas de Informática.
- Actividades Académicamente Dirigidas por el Profesorado: seminarios, conferencias, desarrollo de trabajos, debates, tutorías colectivas, actividades de evaluación y autoevaluación.

5.2. Metodologías docentes:

- Clase Magistral Participativa.
- Desarrollo de Prácticas en Laboratorios Especializados o Aulas de Informática en grupos reducidos.
- Resolución de Problemas y Ejercicios Prácticos.
- Planteamiento, Realización, Tutorización y Presentación de Trabajos.
- Evaluaciones y Exámenes.

5.3. Desarrollo y justificación:

Sesiones de Teoría sobre los contenidos del Programa: Consisten en clases magistrales donde se impartirá la base teórica de la asignatura y se expondrán ejemplos aclaratorios de la misma al grupo. Estas sesiones se irán intercalando con las de problemas a lo largo del curso. La metodología que se utilizará para impartir la teoría y los ejemplos aclaratorios será la exposición mediante mesa tecnológica y pizarra electrónica. El profesor podrá solicitar la participación activa del alumno mediante preguntas rápidas valorando su interés, capacidad y grado de asimilación de la asignatura. En la página web de la asignatura (plataforma Moodle) se encontrarán las transparencias y otros materiales de referencia necesarios para el seguimiento de las sesiones. Se intentará que al final de cada tema el alumno disponga de un ejercicio autoevaluativo.

Sesiones de Resolución de Problemas: Al final de cada tema y como consolidación de lo aprendido, se realizarán sesiones exclusivas de resolución de problemas, que podrán estar divididas en dos fases; una primera en la que el profesor explicará uno o varios problemas tipo y una segunda en la que los alumnos podrán ser evaluados con la resolución que aporten a problemas planteados.

Sesiones Prácticas en Laboratorios Especializados o en Aulas de Informática: Consisten en el estudio y diseño de sistemas software mediante el uso de herramientas CASE. Los alumnos dispondrán con antelación del problema a resolver y la metodología de trabajo a utilizar. El trabajo podrá realizarse de forma individual o por grupos, y la asistencia a las sesiones de laboratorio es obligatoria.

Actividades Académicamente Dirigidas por el Profesorado

6. Temario desarrollado:

BLOQUE I. INTRODUCCIÓN HISTÓRICA Y METODOLÓGICA

Tema 1. Introducción

- 1.1 El desarrollo de software como un proceso.
- 1.2 El software. Factores de calidad del software.
- 1.3 Problemas en el desarrollo de software.
- 1.4 La Ingeniería del Software. Visión general del proceso de Ingeniería del Software.
- 1.5 Responsabilidad ética y profesional en Ingeniería del Software.

BLOQUE II. PROCESOS ORGANIZATIVOS

Tema 2. Gestión proceso

- 2.1 Metodologías basadas en el proceso.
- 2.2 La tiranía del proceso. La corriente Ágil.
- 2.3 Panorama de metodologías actuales.

Tema 3. Gestión del proyecto

- 3.1 Introducción.
- 3.2 El proyecto. El estudio de viabilidad.
- 3.3 Gestión de Riesgos
- 3.4 Planificación.
- 3.5 Estimación de proyectos.

BLOQUE III. PROCESOS PRINCIPALES Tema 4. Ingeniería de Requisitos

- 4.1 Actividades generales de la ingeniería de requisitos.
- 4.2 Técnicas de recogida de la información.
- 4.3 Documentos de especificación de requisitos.
- 4.4 Casos de uso.
- 4.5 Análisis de Requisitos.

Tema 5. Diseño

- 5.1 Modelos de diseño. Diseño estructurado. Diseño O.O.
- 5.2 UML y sus principales diagramas.

Tema 6. Desarrollo, Operación y Mantenimiento

- 6.1 Diagramas UML de Implementación: Diagrama de Componentes y diagrama de Despliegue
- 6.2 Desarrollo. Del Diseño al Código
- 6.3 Diseño Basado en la Reutilización. Patrones de Diseño. Familias de Aplicaciones. Líneas de Producto
- 6.4 Arquitectura del Sistema.
- 6.5 Mantenimiento. Proceso. Beneficios.

BLOQUE IV. PROCESOS DE SOPORTE Y APOYO

Tema 7. Pruebas de software

- 7.1 Introducción. Definiciones
- 7.2 Estrategias pruebas de defectos: Pruebas de caja negra y de caja blanca
- 7.3 Pruebas de Bajo nivel: Pruebas unitarias, de integración y de Regresión
- 7.4 Pruebas de Alto nivel: Pruebas de stress o carga y de Aceptación.

Tema 8. Gestión de la configuración y Documentación

- 8.1 Introducción. "La primera ley"
- 8.2 La gestión del cambio. Proceso. Líneas base (baseline)
- 8.3 La gestión de versiones.
- 8.4 Sistemas de control de versiones.

7. Bibliografía

7.1. Bibliografía básica:

- Utilización de UML en Ingeniería del Software con Objetos y Componentes. P. Stevens. Addison Wesley
- UML gota a gota. Martin Fowler ; con Kendall Scott. Addison-Wesley.
- UML y patrones. Craig Larman. Pearson Prentice Hall

7.2. Bibliografía complementaria:

- Ingeniería del Software. Un enfoque práctico. Roger S. Pressman. Mc Graw-Hill
- Ingeniería del Software. Ian Sommerville. Prentice Hall.
- An introduction to object-oriented programming with java. C. Thomas Wu. McGraw-Hill.
- Patrones de diseño : elementos de software orientado a objetos reutilizable / Erich Gamma... [et al.] Pearson Educación.

8. Sistemas y criterios de evaluación.

8.1. Sistemas de evaluación:

- Examen de teoría/problemas
- Defensa de Prácticas
- Defensa de Trabajos e Informes Escritos

8.2. Criterios de evaluación y calificación:

EXAMEN DE TEORÍA/PROBLEMAS (PARTE A) Examen que se realizará sobre el contenido completo de la asignatura y que se llevará a cabo en las convocatorias y fechas que correspondan según lo establecido por los órganos competentes del Centro. Con este examen se evaluarán las competencias T5 y C16.

DEFENSA DE PRÁCTICAS (PARTE B) Se realizarán un conjunto de prácticas guiadas, con el fin de asimilar los conceptos teóricos y prácticos vistos en clase. Estos ejercicios se explicarán, supervisarán y se comenzarán a trabajar en las clases destinadas para ello, siendo considerada la falta de asistencia a estas sesiones en la evaluación de la asignatura. Los alumnos realizarán, en grupos reducidos o individualmente, ejercicios de aplicación que deberán entregar para ser evaluados.

Si bien, la evaluación de las actividades prácticas se obtiene mediante la media ponderada por dificultad e importancia de cada actividad, los profesores tienen la potestad de considerar suspensa el conjunto de prácticas si alguna de ellas ha sido realizada de manera especialmente deficiente, recuperándose ésta de manera aislada en un ejercicio en la convocatoria de septiembre. Con la realización de prácticas en laboratorio se evaluarán las competencias T1, T2, T3, T6, T8, T10, T11, T12, B4, B5, C1 y C5. P

Defensa de Trabajos e Informes Escritos Los profesores propondrán ejercicios y trabajos para que el alumno desarrolle parte de la asignatura de manera autónoma. Dichos trabajos podrán defenderse públicamente para ser evaluados. De manera excepcional y siempre que las circunstancias lo permitan se podrían organizar conferencias o viajes a centros de producción de las que el alumno debería entregar una memoria. Con estas actividades dirigidas se evaluarán las competencias T8, T10, T11, T12, T15 y C16

CRITERIOS DE EVALUACIÓN Y CALIFICACIÓN

Examen Teórico Escrito 50%

Evaluación de las prácticas entregadas/defendidas 40%

Evaluación de trabajos 10%

La nota en acta del alumno se obtendrá del siguiente modo, teniendo en cuenta que las evaluaciones de los conocimientos teóricos (Parte A), prácticos (Parte B) y de los trabajos se realizarán sobre 10:

Si $\text{Nota_Parte_A} \geq 5$ y $\text{Nota_Parte_B} \geq 5$ Entonces $\text{Nota_Final} = \text{Nota_Parte_A} \times 0,5 + \text{Nota_Parte_B} \times 0,4 + \text{Nota_AAD} \times 0,1$

En caso contrario $\text{Nota_Final} = \text{Mínimo}(\text{Nota_Parte_A}, \text{Nota_Parte_B})$

De manera que estará aprobada la asignatura sólo si la Nota_Final es ≥ 5 .

A los alumnos que no alcancen el aprobado en la asignatura, se les mantendrá la nota de la Parte_A, Parte_B hasta la última convocatoria anual, siempre y cuando su puntuación en las mismas sea ≥ 5 y la nota de los trabajos.

9. Organización docente semanal orientativa:

Semanas	Grupos Grandes	Grupos Reducidos	Aula Estándar	Grupos Reducidos	Aula de Informática	Grupos Reducidos	Laboratorio	Grupos Reducidos	prácticas de campo	Pruebas y/o actividades evaluables	Contenido desarrollado
#1	2	0	2	0	0						Tema 1
#2	2	0	2	0	0						Tema 2
#3	2	0	2	0	0	Comienzo Práctica 1					Tema 2
#4	2	0	2	0	0						Tema 3
#5	2	0	2	0	0	Comienzo Práctica 2					Tema 3
#6	2	0	2	0	0						Tema 4
#7	2	0	2	0	0						Tema 4
#8	2	0	2	0	0						Tema 5
#9	2	0	2	0	0	Comienzo Práctica 3					Tema 5
#10	2	0	2	0	0						Tema 5
#11	2	0	2	0	0						Tema 6
#12	2	0	2	0	0						Tema 6
#13	2	0	2	0	0						Tema 6, 7
#14	2	0	2	0	0						Tema 7, 8
#15	2	0	2	0	0						Tema 8
	30	0	30	0	0						