

## Grado en Ingeniería Informática itinerario Ingeniería del Software

### DATOS DE LA ASIGNATURA

**Nombre:**

Mantenimiento y Gestión del Cambio en Sistemas Software

**Denominación en inglés:**

Software maintenance and software change management.

**Código:**

606010223

**Carácter:**

Obligatorio

**Horas:**

	Totales	Presenciales	No presenciales
<b>Trabajo estimado:</b>	150	60	90

**Créditos:**

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

**Departamentos:**

Tecnologías de la Información

**Áreas de Conocimiento:**

Lenguaje y Sistemas Informáticos

**Curso:**

4º - Cuarto

**Cuatrimestre:**

Segundo cuatrimestre

### DATOS DE LOS PROFESORES

**Nombre:**

\*Pachón Álvarez, Victoria

**E-Mail:**

vpachon@uhu.es

**Teléfono:**

87373

**Despacho:**

60 TorreUmbría

\*Profesor coordinador de la asignatura

**1. Descripción de contenidos****1.1. Breve descripción (en castellano):**

En esta asignatura se describe la metodología básica para conseguir unas buenas competencias en el ámbito del mantenimiento del software, la gestión apropiada de los cambios en el mismo, así como la adecuada configuración de los componentes que forman una aplicación de cualquier tamaño. De forma resumida los principales objetivos son los siguientes:

- Definición de mantenimiento del software, conceptos básicos, tipos de mantenimiento y costes.
- Dificultades y actividades del mantenimiento de software.
- Soluciones técnicas, metodológicas y de gestión
- Modelos de calidad, mantenibilidad y mantenimiento, medida de la mantenibilidad, estándares (ISO 9126. ISO 12207)
- El estándar ISO/IEC 14764 sobre el proceso de mantenimiento del software
- Auditoría del mantenimiento de software: Cobit
- Metodología ITIL y la gestión del cambio.
- Gestión de la Configuración.

**1.2. Breve descripción (en inglés):**

In this subject, the basic paradigms that address the software maintenance and the proper software management and configuration for applications with any size, are described and studied. Summarizing, the main objectives are described as follows:

- Defining software maintenance and the basic concepts about types of maintenance and costs.
- Software maintenance and management activities and related issues.
- Methodological and technical solutions for software configuration and management.
- Models of quality, maintainability and maintenance, maintainability measurements, standards (ISO 9126. ISO 12207)
- The ISO / IEC 14764 on software maintenance processes.
- Auditing Software Maintenance: Cobit
- ITIL methodology for change management.
- Software configuration management.

**2. Situación de la asignatura****2.1. Contexto dentro de la titulación:**

Se trata de una asignatura obligatoria específica para lo/as estudiantes del itinerario de Ingeniería del Software del Grado de Informática. Dicha asignatura se imparte una vez que el/la alumno/a ha desarrollado las destrezas básicas que le permitirán diseñar e implementar componentes software complejos. Esta asignatura complementa al grupo de asignaturas de diseño de software en tanto que proporciona los mecanismos apropiados para llevar a cabo un manejo apropiado de las herramientas que permiten el mantenimiento y configuración del software, así como la gestión de los cambios.

**2.2. Recomendaciones:**

Ninguna

**3. Objetivos (Expresados como resultados del aprendizaje):**

Conocer los principales métodos para llevar a cabo una correcta gestión del cambio en sistemas software de manera que se cumplan los estándares establecidos. Estudiar las técnicas de mantenimiento de software para aplicaciones de cualquier tamaño a través de herramientas apropiadas (gestión de repositorios). Aprender a configurar apropiadamente los diferentes componentes software que forman una aplicación.

**4. Competencias a adquirir por los estudiantes****4.1. Competencias específicas:**

- **CE1-IS:** 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.
- **CE4-IS:** Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales
- **CE5-IS:** Capacidad de identificar, evaluar y gestionar los riesgos potenciales asociados que pudieran presentarse

#### 4.2. Competencias básicas, generales o transversales:

- **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
- **G03:** Capacidad para la resolución de problemas
- **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.
- **T01:** Uso y dominio de una segunda lengua
- **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.
- Tutorías Individuales o Colectivas. Interacción directa profesorado-estudiantes.
- Planteamiento, Realización, Tutorización y Presentación de Trabajos.
- Evaluaciones y Exámenes.

### 5.3. Desarrollo y justificación:

En cada sesión académica de teoría, el profesor explicará los conceptos básicos de cada tema mediante una clase magistral participativa. Para facilitar el proceso de aprendizaje, en el transcurso de estas sesiones se intercalarán sesiones académicas de problemas en las cuales se plantearán problemas de menor complejidad que lo/as alumno/as deberán resolver durante la sesión de forma individual.

En las sesiones prácticas en laboratorio se planteará un problema de mayor complejidad que lo/as alumno/as deberán resolver durante varias sesiones. Durante las sesiones de prácticas, los alumnos desarrollarán su trabajo con ayuda del profesorado. Los enunciados y materiales están disponibles en la web de la asignatura; aún así se recomienda la utilización de libros, recursos y fuentes de conocimiento adicionales.

Además, se llevarán a cabo actividades académicamente dirigidas que consistirán en trabajos en grupos reducidos o individuales cuya evaluación contará para la nota final y en pruebas parciales individuales. Estos conceptos forman parte del "seguimiento individual del estudiante" en el apartado evaluación de esta guía. También se propondrá la defensa de un trabajo, que contará para la nota final

La evaluación final de la asignatura se realizará teniendo en cuenta la evaluación del examen teórico, la calificación de la parte práctica, las actividades académicamente dirigidas y las pruebas parciales, tal como se detalla en la sección "evaluación" de esta guía docente.

La asignatura dispone de una página web donde el alumno puede consultar el material para preparar cada clase, así como la documentación necesaria para cada sesión práctica. Se utilizarán todos los medios tecnológicos disponibles en el aula (vídeo-proyector, wi-fi, etc.). Los alumnos que lo deseen pueden traer material a la clase (libros, portátiles, etc.).

## 6. Temario desarrollado:

Tema 0. Introducción al mantenimiento y gestión del cambio en sistemas software.

1. Importancia del mantenimiento software.
2. Definición de mantenimiento software.
3. Modelos de calidad, mantenibilidad y mantenimiento, medida de la mantenibilidad, estándares (ISO 9126. ISO 12207)
4. El estándar ISO/IEC 14764 sobre el proceso de mantenimiento del software
5. Definición de cambio en sistemas software.
6. Principales retos del mantenimiento software.
7. Estándares: ITIL y Cobit.

Tema 1. Fase de Mantenimiento en el proceso de Ingeniería Software y su ciclo de vida.

1. Aspectos históricos.
2. Costes.
3. Situación en el ciclo de vida de un proyecto.
4. Tipos de mantenimiento:
  1. Actividades de mantenimiento.
  2. Dificultades del mantenimiento.
7. Soluciones al problema del mantenimiento.
  1. Mantenibilidad (facilidad del mantenimiento software).
  2. Mantenimiento futuro. Uso de componentes reutilizables (COTS - Commercial of-the-shelf).
  3. Estándares en el proceso de mantenimiento.
  4. Medidas de calidad

Tema 2. Gestión de la configuración del software.

1. Introducción a la configuración del software
2. Elementos de la configuración del software.
3. Definición de línea base.
4. Proceso de la gestión de la configuración del software.
5. Control de cambios.
  - El problema del control de versiones.
  - Funcionamiento y gestión de los repositorios software
6. Auditorías de las configuraciones.
7. Generación de informes.
8. Introducción a la configuración del software.

Tema 3. Auditoría del mantenimiento de software Cobit

1. Introducción.
2. Aspectos del proceso de auditoría.
3. Uso de Cobit

Tema 4. Metodología ITIL

1. 1. Introducción
2. Características de ITIL
3. Uso de la metodología

## 7. Bibliografía

### 7.1. Bibliografía básica:

Apuntes de clase.

- Mario Piattini, Jose Villalba, Francisco Ruiz, Teresa Bastanchury, Macario Polo, Miguel Angel Martinez, and Cesar Nistal. Mantenimiento del Software. Modelos, Tecnicas y Metodos para la Gestion del Cambio.
- Metodología ITIL (Website).
- Metodología Cobit (Website).
- Sommerville, I., Ingeniería de Software, Pearson Educación, 2002.

### 7.2. Bibliografía complementaria:

- Técnicas de Mantenimiento de Software [Versión Kindle]. Miguel-Angel Sicilia.
- Metrics and Models in Software Quality Engineering. Second Edition. Stephen H. Kan

## 8. Sistemas y criterios de evaluación.

### 8.1. Sistemas de evaluación:

- Examen de teoría/problemas
- Defensa de Prácticas
- Defensa de Trabajos e Informes Escritos
- Seguimiento Individual del Estudiante

## 8.2. Criterios de evaluación y calificación:

La calificación final (evaluación continua) se calcula siguiendo la siguiente fórmula:

$$\text{Nota final} = 0.4 * \text{Nota de teoría} + 0.4 * \text{Nota de prácticas} + 0.1 * \text{Nota seguimiento individual del estudiantes} + 0.1 * \text{Nota Trabajo}$$

El examen de teoría consistirá en la resolución de problemas y preguntas teórico/prácticas relacionadas con el temario de teoría. Para la realización de este examen no se podrá utilizar material adicional salvo el indicado por el profesorado.

La calificación de prácticas se obtendrá mediante la defensa y realización de los enunciados propuesto en clase.

El seguimiento individual del estudiante y la calificación del trabajo se evaluarán mediante las actividades académicamente dirigidas y el desarrollo y exposición de un trabajo tal como se detalla en el apartado "Actividades Formativas y Metodologías Docentes".

Aquellos alumnos que, por circunstancias excepcionales, no puedan seguir la evaluación continua deberán indicarlo en las dos primeras semanas de curso y se les aplicará la siguiente formula de evaluación:

$$\text{Nota final} = 0.6 * \text{Nota de teoría} + 0.4 * \text{Nota de prácticas}$$

En este caso, el alumno deberá realizar y entregar todas las prácticas y realizar la defensa de las mismas.

En ambos casos, para poder aplicar estas fórmulas de evaluación, el estudiante debe obtener, al menos, 3 puntos sobre 10 en la prueba de teoría y 3 puntos sobre 10 en la calificación final de las prácticas.

**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 0	
#2	2	0	2	0	0			
#3	2	0	2	0	0		Tema 1	
#4	2	0	2	0	0			
#5	2	0	2	0	0			
#6	2	0	2	0	0	Prueba 1		
#7	2	0	2	0	0		Tema 2	
#8	2	0	2	0	0			
#9	2	0	2	0	0			
#10	2	0	2	0	0	Prueba 2		
#11	2	0	2	0	0		Tema 3	
#12	2	0	2	0	0	Exposición de trabajos		
#13	2	0	2	0	0		Tema 4	
#14	2	0	2	0	0	Exposición de trabajos		
#15	2	0	2	0	0	Defensa de prácticas		
	30	0	30	0	0			