

GUIA DOCENTE DE LA ASIGNATURA

DESARROLLO DE SOFTWARE

MÓDULO	MATERIA	CURSO	SEMESTRE	CRÉDITOS	TIPO
Formación de Especialidad 2: Ingeniería del Software	Desarrollo y Gestión de Proyectos	3º	Sexto	6	Obligatoria
PROFESORES			DIRECCIÓN COMPLETA DE CONTACTO PARA TUTORÍAS		
Grupos de Teoría Grupo ÚNICO: Manuel Capel Tuñón Martes: 17.30-19.30 Grupos de Prácticas DS1: Manuel Capel Tuñón (Viernes, 15.30-17.30) DS2: Manuel Capel Tuñón (Miércoles, 15.30-17.30) DS3: Manuel Capel Tuñón (Martes, 15.30-17.30)			E.T.S. de Ingeniería en Informática y Telecomunicaciones. Planta 3ª C./ Periodista Daniel Saucedo Aranda s/n. 18071 – Granada. Universidad de Granada Manuel I. Capel Tuñón: Desp. 37 manuelcapel@ugr.es . Tf: 958242816		
			HORARIO DE TUTORÍAS Los horarios de tutorías del profesorado pueden consultarse en la web de grado: http://grados.ugr.es/informatica/pages/infoacademica/profesorado/*/3F y en https://lsi.ugr.es/lsi/manuelcapel		
GRADO EN EL QUE SE IMPARTE			OTROS GRADOS A LOS QUE SE PODRÍA OFERTAR		
Grado en Ingeniería Informática			Doble Grado en Ingeniería Informática y Matemáticas Grado Ingeniería de Tecnologías de Telecomunicación		
PRERREQUISITOS Y/O RECOMENDACIONES (Si ha lugar)					
No es necesario que los alumnos tengan aprobadas asignaturas, materias o módulos previos como requisito indispensable para cursar este módulo. No obstante se recomienda la superación de los contenidos y adquisición de competencias de las materias de formación básica y de rama.					



ugr | Universidad
de Granada

INFORMACIÓN SOBRE TITULACIONES DE LA UGR
<http://grados.ugr.es>

BREVE DESCRIPCIÓN DE CONTENIDOS (SEGÚN MEMORIA DE VERIFICACIÓN DEL GRADO)

Patrones de diseño. Composición y clasificación de los patrones arquitectónicos. Arquitecturas de software.. Técnicas de desarrollo dirigidas por modelos. Modelado de negocio. Técnicas de verificación y validación de software. Pruebas. Mantenimiento y evolución del software.

COMPETENCIAS GENERALES Y ESPECÍFICAS

Competencias Específicas de la Asignatura

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.

IS3. Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.

IS4. 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.

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.

Competencias Específicas del Título

E1. Capacidad para concebir, redactar, organizar, planificar, desarrollar y firmar proyectos en el ámbito de la ingeniería en informática que tengan por objeto, de acuerdo con los conocimientos adquiridos, la concepción, el desarrollo o la explotación de sistemas, servicios y aplicaciones informáticas.

E2. Capacidad para dirigir las actividades objeto de los proyectos del ámbito de la informática de acuerdo con los conocimientos adquiridos.

E3. Capacidad para diseñar, desarrollar, evaluar y asegurar la accesibilidad, ergonomía, usabilidad y seguridad de los sistemas, servicios y aplicaciones informáticas, así como de la información que gestionan.

E4. Capacidad para definir, evaluar y seleccionar plataformas hardware y software para el desarrollo y la ejecución de sistemas, servicios y aplicaciones informáticas.

E5. 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

E6. Capacidad para concebir y desarrollar sistemas o arquitecturas informáticas centralizadas o distribuidas integrando hardware, software y redes.

E7. Capacidad para conocer, comprender y aplicar la legislación necesaria durante el desarrollo de la profesión de Ingeniero Técnico en Informática y manejar especificaciones, reglamentos y normas de obligado cumplimiento.

E9. 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 Técnico en Informática.

E10. Conocimientos para la realización de mediciones, cálculos, valoraciones, tasaciones, peritaciones, estudios, informes, planificación de tareas y otros trabajos análogos de informática.

E12. Conocimiento y aplicación de elementos básicos de economía y de gestión de recursos humanos, organización y planificación de proyectos, así como la legislación, regulación y normalización en el ámbito de los proyectos informáticos, de acuerdo con los conocimientos adquiridos.

Competencias Transversales o Generales

T1. 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.

T2. Capacidad de organización y planificación así como capacidad de gestión de la Información

T4. Capacidad para la resolución de problemas

T5. 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

T6. Capacidad para el uso y aplicación de las TIC en el ámbito académico y profesional.

T8. Capacidad de trabajo en equipo.

OBJETIVOS EXPRESADOS COMO RESULTADOS DEL APRENDIZAJE

- Conocer los patrones que se pueden aplicar a un diseño, su estructura y clasificación y su importancia como herramienta para mejorar la calidad de un diseño.
- Adquirir destreza en la identificación de los patrones aplicables a un determinado problema.
- Conocer las técnicas de desarrollo dirigidas por modelos, sus ventajas e inconvenientes.
- Distinguir los conceptos de validación y verificación de requisitos.
- Conocer los niveles de prueba del software, incluyendo las pruebas unitarias y de integración, las pruebas de validación, las de sistema y las de aceptación
- Ser capaz de definir e implementar pruebas unitarias, integración, regresión y sistema.
- Comprender el concepto de componente y su papel en el proceso de desarrollo de software.
- Conocer cómo modelar procesos de negocio, sus métodos, técnicas y herramientas.
- Conocer la importancia del mantenimiento y su integración en el proceso de desarrollo de software.
- Evaluar el coste del mantenimiento de un sistema (Belady-Lehman)

TEMARIO DETALLADO DE LA ASIGNATURA

TEMARIO DE TEORÍA

Tema 1. Desarrollo utilizando patrones-software

- 1.1. Análisis y diseño con patrones.
- 1.2. Catálogo de patrones.

Tema 2. Arquitecturas Software

- 2.1. Estilos arquitectónicos.
- 2.2. Notaciones actuales para representación de las arquitecturas software.
- 2.3. Arquitecturas orientadas a componentes y servicios.

Tema 3. Desarrollo dirigido por modelos

- 3.1. Introducción al desarrollo dirigido por modelos.
- 3.2. MDA (Model Driven Architecture) y factorías de software.
- 3.3. Modelado de negocio.
- 3.4. Arquitecturas de Empresa/Sistemas de Información Empresarial.

Tema 4. Validación del software

- 4.1. Objetivos y principios.
- 4.2. Validación y verificación. Relación con el ciclo de vida del software.
- 4.3. Técnicas, herramientas y marcos de trabajo actuales.

Tema 5. Mantenimiento y evolución del software.

- 5.1 Principios del mantenimiento y tipos.
- 5.2 El proceso de mantenimiento.



5.3. Actividades de mantenimiento.

TEMARIO DE PRÁCTICAS

Práctica 1: Realizar la aplicación del catálogo de patrones definido en clase al diseño de varios programas propuestos. Analizar la adecuación del diseño final a los requisitos especificados.

Práctica 2: Realizar una propuesta y definición de una arquitectura software y aplicarlo al desarrollo completo de un ejemplo entre varios propuestos utilizando para ello el lenguaje Java. Se evaluarán las propuestas arquitectónicas durante las sesiones de prácticas.

Práctica 3: Aplicar distintos tipos de pruebas de software a los proyectos desarrollados en la práctica 2.

BIBLIOGRAFÍA**BIBLIOGRAFÍA FUNDAMENTAL:**

- Bruegge, B. Dutoit, A.H. "Object-Oriented Software Engineering. Using UML Patterns, and Java". Pearson Education, 2004.
- Larman, C. "Applying UML and Patterns: An Introduction to Object-oriented Analysis and Design and Iterative Development", Prentice-Hall, 2008 ("UML y patrones: Una Introducción al Análisis y Diseño Orientado a Objetos y al Proceso Unificado". Pearson Educación).
- Lethbridge, T., Laganieri, R. "Object-Oriented Software Engineering: Practical Software Development Using UML and Java". McGraw-Hill 2005.
- Fowler, M. "Patterns of Enterprise Application Architecture". Addison-Wesley Professional, 2002
- Gamma, E., Helm, E., Johnson, R., Vlissides, J. "Design Patterns. Elements of Reusable Object-Oriented Software". Addison-Wesley, 1995. Traducción al español de César Fernández Acebal y Juan Manuel Cueva Lovelle. Pearson Educación, 2009.
- Jazarbeck, S. "Effective Software Maintenance and Evolution: A Reuse-Based Approach". Auerbach, 2007.
- Pressman, R.S. Ingeniería del Software, 7ª edición. McGraw Hill, 2010.

BIBLIOGRAFÍA COMPLEMENTARIA:

- Albin, S.T. *The Art of Software Architecture: Design Methods and Techniques*. Indianapolis, Indiana: John Wiley & Sons, 2003.
- Arlow, J. y Neustad, I. UML 2, Anaya, 2006.
- Bass, L., Clements, P., Kazman, R. "Software Architecture in Practice (3rd Edition) (SEI Series in Software Engineering)". Addison-Wesley, 2012.
- Booch, G. Handbook of Software Architecture, 2008. <http://www.handbookofsoftwarearchitecture.com/>
- Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., Stal, M. "Pattern-Oriented Software Architecture Volume 1: A System of Patterns". Wiley, 2004.
- Esposito, D., Saltarello, A. "Microsoft .NET: Architecting Applications for the Enterprise (PRODeveloper)", Microsoft Press; 2008.
- Everett, G.D., McLeod, R. "Software Testing: Testing Across the Entire Software Development Life Cycle", Wiley, 2007.
- Lewis, W.E. "Software Testing and Continuous Quality Improvement", CRC Press (Auerbach), 2009.
- Pigoski, T.M. "Practical Software Maintenance: Best Practices for Managing Your Software Investment", Wiley, 1996.
- Shaw, M., Garlan, D. "Software Architecture: Perspectives on an Emerging Discipline", Prentice-Hall, 1994.
- Rainsberger, J.B. JUnit Recipes: Practical Methods for Programmer Testing. Manning Publications, 2005.
- Zyperski, C.A., D. Gruntz, S. Murer. "Component Software: Beyond Object-Oriented Programming", Addison-Wesley, 2002.

ENLACES RECOMENDADOS: Tanto en la página web de la asignatura (accesible desde <http://lsi.ugr.es/~ist>) como en la plataforma Tutor (<http://swad.ugr.es>) se encontraran los enlaces recomendados para la asignatura.



METODOLOGÍA DOCENTE

1. Lección magistral (Clases teóricas-expositivas) (grupo grande)

Descripción: Presentación en el aula de los conceptos propios de la materia haciendo uso de metodología expositiva con lecciones magistrales participativas y medios audiovisuales. Evaluación y examen de las capacidades adquiridas.

Propósito: Transmitir los contenidos de la materia motivando al alumnado a la reflexión, facilitándole el descubrimiento de las relaciones entre diversos conceptos y formarle una mentalidad crítica. Contenido en ECTS: 30 horas presenciales (1.2 ECTS).

Competencias: IS1, IS3, IS4, IS6, E1, E2, E3, E4, E5, E6, E7, E9, E10, E12, T1, T2, T4, T5, T6

2. Actividades prácticas (Clases prácticas de laboratorio) (grupo pequeño)

Descripción: Actividades a través de las cuales se pretende mostrar al alumnado cómo debe actuar a partir de la aplicación de los conocimientos adquiridos. Se desarrollará un sistema software de complejidad mediana basado en un supuesto práctico que se proporcionará. La codificación completa, pruebas unitarias y de integración del sistema aludido se consideran también incluidas.

Propósito: Desarrollo en el alumnado de las habilidades instrumentales de la materia.

Contenido en ECTS: 15 horas presenciales (0.6 ECTS) Competencias: IS1, IS3, IS4, IS6, E1, E2, E3, E4, E5, E6, E7, E9, E10, E12, T1, T2, T4, T5, T6

Regimen de Asistencia: dado el carácter continuado de las actividades a realizar en esta parte, sólo se permitirá faltar al 10% de las sesiones prácticas para proseguir con el sistema de evaluación continuada de las prácticas.

3. Seminarios (grupo pequeño)

Descripción: Modalidad organizativa de los procesos de enseñanza y aprendizaje donde tratar en profundidad una temática relacionada con la materia. Incorpora actividades basadas en la indagación, el debate, la reflexión y el intercambio.

Propósito: Desarrollo en el alumnado de las competencias cognitivas y procedimentales de la materia.

Contenido en ECTS: 10 horas presenciales (0.4 ECTS)

Competencias: IS1, IS3, IS4, IS6, E1, E2, E3, E4, E5, E6, E7, E9, E10, E12, T1, T2, T4, T5, T6

4. Actividades no presenciales individuales (Estudio y trabajo autónomo)

Descripción: 1) Actividades (guiadas y no guiadas) propuestas por el profesor a través de las cuales y de forma individual se profundiza en aspectos concretos de la materia posibilitando al estudiante avanzar en la adquisición de determinados conocimientos y procedimientos de la materia; 2) Estudio individualizado de los contenidos de la materia; 3) Actividades evaluativas (informes, exámenes, ...)

Propósito: Favorecer en el estudiante la capacidad para autorregular su aprendizaje, planificándolo, diseñándolo, evaluándolo y adecuándolo a sus especiales condiciones e intereses.

Contenido en ECTS: 45 horas no presenciales (1.8 ECTS)

Competencias: IS1, IS3, IS4, IS6, E1, E2, E3, E4, E5, E6, E7, E9, E10, E12, T1, T2, T4, T5, T6

5. Actividades no presenciales grupales (Estudio y trabajo en grupo)

Descripción: Actividades (guiadas y no guiadas) propuestas por el profesor a través de las cuales y de forma grupal se profundiza en aspectos concretos de la materia posibilitando a los estudiantes avanzar en la adquisición de determinados conocimientos y procedimientos de la materia.

Propósito: Favorecer en los estudiantes la generación e intercambio de ideas, la identificación y análisis de diferentes puntos de vista sobre una temática, la generalización o transferencia de conocimiento y la valoración crítica del mismo.

Contenido en ECTS: 45 horas no presenciales (1.8 ECTS)

Competencias: IS1, IS3, IS4, IS6, E1, E2, E3, E4, E5, E6, E7, E9, E10, E12, T1, T2, T4, T5, T6, T8

6. Tutorías académicas (grupo pequeño)

Descripción: manera de organizar los procesos de enseñanza y aprendizaje que se basa en la interacción directa entre el estudiante y el profesor.

Propósito: 1) Orientan el trabajo autónomo y grupal del alumnado; 2) profundizar en distintos aspectos de la



materia y 3) orientar la formación académica-integral del estudiante.

Contenido en ECTS: 5 horas presenciales, grupales e individuales (0.2 ECTS)

Competencias: IS1, IS3, IS4, IS6, E1, E2, E3, E4, E5, E6, E7, E9, E10, E12, T1, T2, T4, T5, T6

EVALUACIÓN (INSTRUMENTOS DE EVALUACIÓN, CRITERIOS DE EVALUACIÓN Y PORCENTAJE SOBRE LA CALIFICACIÓN FINAL, ETC.)

Se utilizarán las siguientes técnicas de evaluación:

- De acuerdo con la normativa recientemente aprobada por la Ugr, existen dos modalidades de evaluación que han de ser concedidas al alumno al comienzo de la impartición: (1) examen final de toda la asignatura (teoría y práctica) y (2) evaluación continuada.
- Para la evaluar la parte teórica se realizará una prueba objetiva final obligatoria. Como se indica más abajo, la calificación de esta parte incluirá el trabajo autónomo mediante la nota obtenida en los ejercicios asignados en clase si se entregan en plazo. La ponderación de este bloque es del 50%.
- Para evaluación de esta parte se calificarán todas las prácticas asignadas y entregadas en el plazo establecido. Es obligatoria la entrega de todas las prácticas y la asistencia. Las prácticas se pueden compensar con la calificación de la teoría sólo si se alcanza una calificación mínima establecida. La ponderación de este bloque es del 50% .
- En su caso, la parte de trabajo autónomo se evaluará teniendo en cuenta la participación activa en clase y la asistencia a los seminarios, cuya asistencia es obligatoria para todo el alumnado, los problemas propuestos que hayan sido resueltos y entregados por los alumnos y la presentación oral de los trabajos desarrollados. La ponderación de este bloque está incluida en la evaluación de los 2 anteriores.

El resultado de la evaluación será una calificación numérica de acuerdo con la modalidad de evaluación que se aplique a cada alumno.

Todo lo relativo a la evaluación se regirá por la normativa sobre planificación docente y organización de exámenes vigente en la Universidad de Granada, que puede consultarse en:

[http://secretariageneral.ugr.es/bougr/pages/bougr71/ncg712/!](http://secretariageneral.ugr.es/bougr/pages/bougr71/ncg712/)

El sistema de calificaciones se expresará mediante calificación numérica de acuerdo con lo establecido en el art. 5 del R. D 1125/2003, de 5 de septiembre, por el que se establece el sistema europeo de créditos y el sistema de calificaciones en las titulaciones universitarias de carácter oficial y validez en el territorio nacional.

