

GUIA DOCENTE DE LA ASIGNATURA

METODOLOGÍAS DE DESARROLLO ÁGIL

MÓDULO	MATERIA	CURSO	SEMESTRE	CRÉDITOS	TIPO
Formación de Especialidad 2: Ingeniería del Software	Desarrollo y Gestión de Proyectos	4º	1º	6	Optativa
PROFESOR(ES)		DIRECCIÓN COMPLETA DE CONTACTO PARA TUTORÍAS (Dirección postal, teléfono, correo electrónico, etc.)			
<p>Grupo de Teoría: María Luisa Rodríguez Almendros (coordinador)</p> <p>Grupos de Prácticas Grupo 1. Francisco Luis Gutiérrez Vela Grupo2. María Luisa Rodríguez Almendros</p>		<p>E.T.S Ingeniería Informática y de Telecomunicaciones Departamento de Lenguajes y Sistemas Informáticos 3ª Planta.</p> <p>Francisco Luis Gutiérrez Vela Despacho 30. Teléfono: 958242812 fgutierr@ugr.es</p> <p>María Luisa Rodríguez Almendros Despacho 29. Teléfono: 958242811 mlra@ugr.es</p>			
		HORARIO DE TUTORÍAS			
		<p>Francisco Luis Gutiérrez Vela Ver Página Web ... http://lsi.ugr.es/lsi/fgutierr</p> <p>María Luisa Rodríguez Almendros Ver Página web ... http://lsi.ugr.es/lsi/mlra</p>			
GRADO EN EL QUE SE IMPARTE		OTROS GRADOS A LOS QUE SE PODRÍA OFERTAR			
Grado en Ingeniería Informática					



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, teniendo especial importancia la superación de las materias "Fundamentos de Ingeniería del Software", "Programación y Diseño Orientado a Objetos" y "Diseño de Interfaces de Usuario".

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

Desarrollo centrado en el usuario. Metodologías de desarrollo Ágiles. El software libre y la empresa. Metodologías de desarrollo usadas en el software libre. Técnicas de evaluación de la calidad. Desarrollo dirigido por las pruebas.

COMPETENCIAS GENERALES Y ESPECÍFICAS**Competencias Generales 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.

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.

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.

Competencias Transversales

T2. 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.



OBJETIVOS (EXPRESADOS COMO RESULTADOS DE APRENDIZAJE)

- Conocer los principales métodos de obtención de requisitos y las situaciones en las que pueden ser usados.
- Comprender el papel que juega el usuario en la ingeniería de requisitos y su posterior participación en el desarrollo del software.
- Conocer las técnicas de desarrollo centradas en el usuario y su aportación a la ingeniería del software.
- Entender la naturaleza de la filosofía del desarrollo ágil y los métodos asociados al mismo.
- Identificar y estudiar los diferentes esquemas de desarrollo ágiles y las técnicas y herramientas que proponen.
- Identificar las formas de estructurar un proyecto y las técnicas de gestión más adecuadas para el control de un proyecto de forma ágil.
- Analizar la importancia del capital humano en relación a las técnicas ágiles y su incorporación a un equipo de desarrollo.
- Conocer el software libre sus tipos y características junto con su integración en la empresa.
- Conocer la estructura y los procesos de desarrollo en ámbitos del software libre y las actividades de emprender, implantar y liderar proyectos en este ámbito.
- Conocer las propiedades de calidad que afectan a todo producto y proceso de desarrollo software y las técnicas que podemos usar para su evaluación.
- Ser capaz de aplicar un proceso básico de desarrollo de software dirigido por las pruebas.
- Ser capaz de 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.
- Ser capaz de 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.
- 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.



TEMARIO DETALLADO DE LA ASIGNATURA**TEMARIO DE TEORÍA****Tema 1. Metodologías de desarrollo Ágiles**

- 1.1 Etapas del desarrollo del Software. Enfoque tradicional vs Ágil.
- 1.2 Metodologías y enfoques Ágiles. XP y SCRUM.
- 1.3 Herramientas Ágiles.

Tema 2. Desarrollo centrado en el usuario.

- 2.1 Ingeniería de requisitos.
- 2.2 Técnicas de elicitación de requisitos.
- 2.3 Metodologías centradas en el usuario.
- 2.4 Herramientas de apoyo.

Tema 3. El Software libre y su desarrollo.

- 3.1 El Software libre y las tecnologías de desarrollo.
- 3.2 El software libre y la empresa. Modelo de Negocio.

Tema 4. Calidad, Validación y Verificación del Software.

- 4.1 Evaluación de la calidad y mejora del desarrollo.
- 4.2 Desarrollo dirigido por las pruebas. Técnicas de refactoring.

TEMARIO DE PRÁCTICAS

Se crearán grupos de trabajo que tendrán que realizar las siguientes actividades:

Práctica 1: Desarrollo de una aplicación usando las técnicas descritas en clase de teoría. Desde su concepción inicial hasta su distribución.

Práctica 2: Presentación y puesta en común de un proyecto de software libre. Planificación, definición de negocio y estudio de viabilidad.

BIBLIOGRAFÍA

BIBLIOGRAFÍA FUNDAMENTAL:

- Craig Larman. *Agile and Iterative Development: A Manager's Guide*. The Agile Software Development Series. Addison-Wesley Professional .2003
- Alistair Cockburn, *Agile Software Development*, Addison-Wesley Professional, 2001
- William C Wake, *Extreme Programming Explored*, Addison Wesley, 2001
- Ken Schwaber, *Agile, Project Management with SCRUM*, Microsoft Press, 2004
- Eric S. Raymond, *The Cathedral and the Bazaar, Musings on Linux and Open Source by an Accidental Revolutionary*. O'Reilly & Associates, 2001
- Sommerville, I., Sawyer, P. *Requirements Engineering. A good practice guide*. Wiley, 1997
- Fowler, M. et al. 1999. *Refactoring: Improving the Design of Existing Code*. Addison-Wesley

BIBLIOGRAFÍA COMPLEMENTARIA:



- Ken Beck, *Aceptar el cambio*, Addison Wesley, 2001.
- Ken Schwaber. *Agile Project Management with Scrum*. Microsoft Professional. 2004
- Andreas Rueping. *Agile Documentation: A Pattern Guide to Producing Lightweight Documents for Software Projects*. Wiley Software Patterns Series. 2002
- Stephen R. Palmer, John Felsing, *A Practical Guide to Feature-Driven Development*, Prentice Hall, 2002
- Kieran Healy and Alan Schussman. *The ecology of open source software development*, 2003
- Kent Beck, *Una explicación de la Programación Extrema: Aceptar el Cambio*. Addison-Wesley Iberoamericana España; 2000.
- Mike Cohn, *Succeeding with Agile: Software Development Using Scrum*, Addison-Wesley Professional; 2009.
- Karl Fogel and Moshe Bar, *Open Source Development with CVS*, 3rd Edition; Paraglyph Press, 2003
- DiBona, Chris; Cooper, Danese; Stone, Mark, *Open source software*, O'Reilly, 2005
- Joseph Feller, Brian Fitzgerald, *Understanding Open Source Software development*, Addison-Wesley, 2002
- Ian Sommerville. *Ingeniería del Software*. Sexta edición. Editorial Addison Wesley,
- Robertson, S., Robertson, J. *Mastering the Requirements Process*. Addison-Wesley, 1999
- Sutcliffe, A. *User-Centred Requirements Engineering*. Theory and Practice., Springer, 2002
- K.L., and Harbison, K., Lawrence, *User-centered requirements: The Scenarios-Based Engineering Process*, McGraw,; New Jersey, 1997.

ENLACES RECOMENDADOS

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: E1, E2, E5, E9, T2

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

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

Contenido en ECTS: 15 horas presenciales (0.6 ECTS)

Competencias: E1, E2, E5, E9, T2

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: E1, E2, E5, E9, T2

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: 4,5 horas no presenciales (1.8 ECTS)

Competencias: E1, E2, E5, E9, T2

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: 4,5 horas no presenciales (1.8 ECTS)

Competencias: E1, E2, E5, E9, T2

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: E1, E2, E5, E9, T2

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

Todo lo relativo a la evaluación se regirá por la Normativa de evaluación y calificación de los estudiantes vigente en la Universidad de Granada, que puede consultarse en:

<http://secretariageneral.ugr.es/bougr/pages/bougr71/ncg712/>

Preferentemente, la evaluación se ajustará al sistema de evaluación continua del aprendizaje del estudiante siguiendo el artículo 7 de la anterior Normativa.

Se utilizarán alguna o algunas las siguientes técnicas de evaluación:

- Para la parte teórica se realizarán exámenes finales o parciales, sesiones de evaluación y entregas de ejercicios sobre el desarrollo y los resultados de las actividades propuestas.
- Para la parte práctica se realizarán prácticas de laboratorio, resolución de problemas y desarrollo de proyectos (individuales o en grupo), y se valorarán las entregas de los informes/memorias realizados por los alumnos, o en su caso las entrevistas personales con los alumnos y las sesiones de evaluación.
- En el caso de la evaluación continua, su caso, los seminarios se evaluarán teniendo en cuenta la asistencia, los



problemas propuestos que hayan sido resueltos y entregados por los alumnos, en su caso, las entrevistas efectuadas durante el curso y la presentación oral de los trabajos desarrollados.

Para los estudiantes que se acojan a la evaluación única final, esta modalidad de evaluación estará formada por todas aquellas pruebas que el profesor estime oportunas, de forma que se pueda acreditar que el estudiante ha adquirido la totalidad de las competencias generales y específicas descritas en el apartado correspondiente de esta Guía Docente.

La calificación global corresponderá por tanto a la puntuación ponderada de los diferentes aspectos y actividades que integran el sistema de evaluación. Por tanto, el resultado de la evaluación será una calificación numérica obtenida mediante la suma ponderada de las calificaciones correspondientes a una parte teórica, una parte práctica y, en el caso de la evaluación continua, una parte relacionada con el trabajo autónomo de los alumnos, los seminarios impartidos y el aprendizaje basado en proyectos. La adaptación del sistema de evaluación general propuesto a las características de esta asignatura, con indicación explícita del peso de la evaluación de cada formativa, se ajustará a lo indicado en la siguiente tabla:

Actividades Formativas	Ponderación
Parte Teórica	20% - 80%
Parte Práctica	30% - 60%
Otros (seminarios,...)	10% - 30%

Para aprobar la asignatura es necesario tener una calificación numérica superior o igual a 5 (sobre 10). No obstante, además del requisito anterior, se establece como requisito adicional para superar la asignatura que tanto la calificación correspondiente a la parte teórica como la correspondiente a la parte práctica sean mayores o iguales a 5 (sobre 10).

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.

RÉGIMEN DE ASISTENCIA

- La asistencia a las clases teóricas no será obligatoria, aunque la participación activa en la clase y la entrega de ejercicios planteados por el profesorado se tendrá en cuenta dentro del sistema de evaluación continua de la asignatura.
- Dentro de la evaluación continua el régimen de asistencia a las secciones de prácticas es obligatorio, pudiendo solo faltar justificadamente a un número determinado de sesiones que se indicará al comienzo del curso.

INFORMACIÓN ADICIONAL



Definición de grupo grande y grupo pequeño:
Los grupos grandes son grupos de 45 a 60 estudiantes.
Los grupos pequeños son grupos de 15 a 20 estudiantes.

