



Ingeniería de Software I

Guía de Aprendizaje – Información al estudiante

1. Datos Descriptivos

Asignatura	Ingeniería de Software I
Materia	Ingeniería del Software, Sistemas de Información y Sistemas Inteligentes
Departamento responsable	DLSIIS
Créditos ECTS	6
Carácter	Obligatoria
Titulación	Grado en Ingeniería Informática
Curso	Tercero
Especialidad	No aplica

Curso académico	2010 - 2011
Semestre en que se imparte	Sexto
Semestre principal	
Idioma en que se imparte	Español
Página Web	



2. Profesorado

NOMBRE Y APELLIDO	DESPACHO	Correo electrónico
Andrés Silva Vázquez	5102	asilva@fi.upm.es
Nelson Medinilla Martínez	5109	nelson@fi.upm.es
Angélica de Antonio	5108	angelica@fi.upm.es
Tomás San Feliú	5106	tsanfe@fi.upm.es

3. Conocimientos previos requeridos para poder seguir con normalidad la asignatura

Asignaturas superadas	<ul style="list-style-type: none">• Programación II
Otros resultados de aprendizaje necesarios	<ul style="list-style-type: none">•



4. Objetivos de Aprendizaje

COMPETENCIAS ESPECÍFICAS ASIGNADAS A LA ASIGNATURA Y SU NIVEL DE ADQUISICIÓN		
Código	Competencia	Nivel
CE-9	Poseer las destrezas que se requieren para diseñar e implementar unidades estructurales mayores que utilizan los algoritmos y las estructuras de datos, así como las interfaces por las que se comunican estas unidades.	Aplicar
CE-22	Capacidad de aplicar sus conocimientos e intuición para diseñar el software que cumple unos requisitos especificados.	Aplicar
CE-25	Concebir y diseñar la arquitectura de un sistema software.	Conocer
CE-32	Comprender el concepto de ciclo de vida que abarca el CE. Significado de sus fases, las consecuencias para el desarrollo de todos los aspectos de los sistemas informáticos y la relación entre la calidad y la gestión del ciclo de vida.	Comprender
CE-41	Elegir y usar modelos de proceso y entornos de programación apropiados para proyectos que implican aplicaciones tradicionales así como áreas de aplicación emergentes.	Comprender

LEYENDA: Nivel de adquisición 1:

Nivel de adquisición 2:

Nivel de adquisición 3:



RESULTADOS DE APRENDIZAJE DE LA ASIGNATURA			
Código	Resultado de aprendizaje	Competencias asociadas	Nivel de adquisición
RA1	Capacidad de llevar a cabo la definición y gestión de requisitos	CE-22	Aplicación
RA2	Capacidad de aplicar técnicas para el análisis, diseño y desarrollo de un sistema software.	CE-9,CE-22,CE-25,CE-32,CE-41	Aplicación
RA3	Destrezas y criterios para el diseño y desarrollo de sistemas software.	CE-9,CE-22,CE-25,CE-32,CE-41	Aplicación



Sistema de evaluación de la asignatura

INDICADORES DE LOGRO		
Ref	Indicador	Relacionado con RA
I1	Conceptos de Ingeniería del Software	RA1
I2	Ingeniería de Requisitos	RA1
I3	Diseño y argumentación de un sistema software Estructurado	RA2, RA3
I4	Diseño y argumentación de un sistema software Orientado a Objetos	RA2, RA3

EVALUACION SUMATIVA			
Breve descripción de las actividades evaluables	Momento	Lugar	Peso en la calif.
Introducción	2º semana	aula	10%
Requisitos	4ª semana	Moodle	15%
Diseño Estructurado	8º semana	Moodle	20%
Diseño Orientado a objetos	16º semana	aula	55%
			Total: 100%



CRITERIOS DE CALIFICACIÓN

La asignatura se basa en el sistema de evaluación continua.

La evaluación será parcial y acumulativa.

En los temas de *Ingeniería de Requisitos y Estructurado* la evaluación se basará en trabajos prácticos a entregar por los alumnos.

En los temas de *Introducción y Diseño Orientado a Objetos* la evaluación será oral, complementada con documentos de trabajo realizados por los alumnos.

NOTA: En virtud de lo establecido por la Normativa reguladora de los sistemas de evaluación en los procesos formativos vinculados a los títulos de Grado y Máster Universitario con Planes de estudio adaptados al R.D. 1393/2007, vigente desde el 1 de septiembre de 2010, en la convocatoria ordinaria, la elección entre el sistema de evaluación continua o el sistema de evaluación mediante sólo prueba final corresponde al estudiante. El procedimiento y el plazo establecidos para optar por este sistema estará sujeto a lo que establezca la Jefatura de Estudios de conformidad con lo estipulado en dicha Normativa (más información en <http://www.fi.upm.es/?pagina=1147>).



5. Contenidos y Actividades de Aprendizaje

CONTENIDOS ESPECÍFICOS		
Bloque / Tema / Capítulo	Apartado	Indicadores Relacionados
Tema 1: Introducción a la Ingeniería del Software	1.1 Conceptos básicos de la Ingeniería de Software. Modelos y métodos. Evolución de los conceptos. .	I1, I2, I3, I4
	1.2 Métodos de simplificación: división e introducción de incertidumbre.	I1, I2, I3, I4
	1.3 Relaciones entre modelos de software, diseños y métodos de desarrollo de sistemas software.	I1, I2, I3, I4
	1.4 El software como diseño. Criterios de diseño software.	I1, I2, I3, I4
	1.5 La visión de sistema. Relaciones del software con otros tipos de sistemas. Soporte del diseño software en la Teoría General de Sistemas. Organización de sistemas.	I1, I2, I3, I4
Tema 2: Ingeniería de Requisitos	2.1 Introducción a la IR	I2
	2.2 Educción, Análisis y Negociación	I2
	2.3 Especificación y Validación	I2
	2.4 Gestión de Requisitos y Herramientas	I2
Tema 3: Diseño Estructurado de sistemas software	3.1 El método de Yourdon	I3
	3.2 Acontecimientos	
	3.3 Elaboración de DFDs	
Tema 4: Diseño Orientado a Objetos de sistemas software	4.1 El modelo de objetos y su contraste con el estructurado. Criterios de selección.	I4
	4.2 Revisión de los conceptos de objetos, mensajes y clases. UML. Principio de ocultación.	I4



	4.3 Revisión de los conceptos de herencia y polimorfismo. Representación en UML. Principio de sustitución de Liskov.	14
	4.4 Técnica de Casos de Uso. Representación en UML.	14
	4.5 Fundamentos teóricos de los patrones. Análisis de patrones relevantes.	14
	4.6 Arquitecturas de sistemas software	14

6. Breve descripción de las modalidades organizativas utilizadas y de los métodos de enseñanza empleados

Tabla 7. Modalidades organizativas de la enseñanza








MODALIDADES ORGANIZATIVAS DE LA ENSEÑANZA		
Escenario	Modalidad	Finalidad
	Clases Teóricas	<i>Hablar a los estudiantes</i>
	Seminarios-Talleres	<i>Construir conocimiento a través de la interacción y la actividad de los estudiantes</i>
	Clases Prácticas	<i>Mostrar a los estudiantes cómo deben actuar</i>
	Prácticas Externas	<i>Completar la formación de los alumnos en un contexto profesional</i>
	Tutorías	<i>Atención personalizada a los estudiantes</i>
	Trabajo en grupo	<i>Hacer que los estudiantes aprendan entre ellos</i>
	Trabajo autónomo	<i>Desarrollar la capacidad de autoaprendizaje</i>

Tabla 5. Métodos de enseñanza

MÉTODOS DE ENSEÑANZA		
	Método	Finalidad
	Método Expositivo/Lección Magistral	Transmitir conocimientos y activar procesos cognitivos en el estudiante
	Estudio de Casos	Adquisición de aprendizajes mediante el análisis de casos reales o simulados
	Resolución de Ejercicios y Problemas	Ejercitar, ensayar y poner en práctica los conocimientos previos
	Aprendizaje Basado en Problemas (ABP)	Desarrollar aprendizajes activos a través de la resolución de problemas
	Aprendizaje orientado a Proyectos	Realización de un proyecto para la resolución de un problema, aplicando habilidades y conocimientos adquiridos
	Aprendizaje Cooperativo	Desarrollar aprendizajes activos y significativos de forma cooperativa
	Contrato de Aprendizaje	Desarrollar el aprendizaje autónomo

Se conoce como método expositivo "la presentación de un tema lógicamente estructurado con la finalidad de facilitar información organizada siguiendo criterios adecuados a la finalidad pretendida". Esta metodología -también conocida como lección (lecture)- se centra fundamentalmente en la exposición verbal por parte del profesor de los contenidos sobre la materia objeto de estudio. El término "lección magistral" se suele utilizar para denominar un tipo específico de lección impartida por un profesor en ocasiones especiales.

Análisis intensivo y completo de un hecho, problema o suceso real con la finalidad de conocerlo, interpretarlo, resolverlo, generar hipótesis, contrastar datos, reflexionar, completar conocimientos, diagnosticarlo y, en ocasiones, entrenarse en los posibles procedimientos alternativos de solución.

Situaciones en las que se solicita a los estudiantes que desarrollen las soluciones adecuadas o correctas mediante la ejercitación de rutinas, la aplicación de fórmulas o algoritmos, la aplicación de procedimientos de transformación de la información disponible y la interpretación de los resultados. Se suele utilizar como complemento de la lección magistral.

Método de enseñanza-aprendizaje cuyo punto de partida es un problema que, diseñado por el profesor, el estudiante ha de resolver para desarrollar determinadas competencias previamente definidas.

Método de enseñanza-aprendizaje en el que los estudiantes llevan a cabo la realización de un proyecto en un tiempo determinado para resolver un problema o abordar una tarea mediante la planificación, diseño y realización de una serie de actividades, y todo ello a partir del desarrollo y aplicación de aprendizajes adquiridos y del uso efectivo de recursos.

Enfoque interactivo de organización del trabajo en el aula en el cual los alumnos son responsables de su aprendizaje y del de sus compañeros en una estrategia de corresponsabilidad para alcanzar metas e incentivos grupales. Es tanto un método, a utilizar entre otros, como un enfoque global de la enseñanza, una filosofía.

Un acuerdo establecido entre el profesor y el estudiante para la consecución de unos aprendizajes a través de una propuesta de trabajo autónomo, con una supervisión por parte del profesor y durante un periodo determinado. En el contrato de aprendizaje es básico un acuerdo formalizado, una relación de contraprestación recíproca, una implicación personal y un marco temporal de ejecución.

BREVE DESCRIPCIÓN DE LAS MODALIDADES ORGANIZATIVAS UTILIZADAS Y METODOS DE ENSEÑANZA EMPLEADOS

CLASES DE TEORIA	Durante una clase de teoría o lección magistral, el profesor realiza una exposición verbal de los contenidos sobre la materia objeto de estudio, mediante la cual suministra a los alumnos información esencial y organizada procedente de diversas fuentes con unos objetivos específicos predefinidos (motivar al alumno, exponer los contenidos sobre un tema, explicar conocimientos, efectuar demostraciones teóricas, presentar experiencias, etc.) pudiendo utilizar para ello, además de la exposición oral, otros recursos didácticos (audiovisuales, documentos, etc.).
CLASES DE PROBLEMAS	<p>Este método de enseñanza se utiliza como complemento de la clase de teoría (lección magistral) y se basa en solicitar a los estudiantes que desarrollen soluciones adecuadas a un determinado fin, mediante la ejercitación de rutinas, la aplicación de fórmulas o algoritmos, la aplicación de procedimientos de transformación de la información disponible y la interpretación de los resultados.</p> <p>La intención principal es la de aplicar lo ya aprendido para favorecer la comprensión tanto de la importancia como del contenido de un nuevo tema, afianzar conocimientos y estrategias y su aplicación en las situaciones prácticas que se planteen.</p>
PRÁCTICAS	Se trata de la realización de proyectos de desarrollo software de tamaño medio completos. Los alumnos deberán trabajar a partir de un documento con la descripción detallada de las especificaciones funcionales que debe cumplimentar el proyecto. El producto final producido deberá pasar un conjunto exhaustivo de pruebas funcionales.
TRABAJOS AUTONOMOS	Se trata de actividades que el alumno deberá abordar de forma individual, sin supervisión del profesor, pero recibiendo retroalimentación por parte de este último y soporte a través de tutorías no programadas. El propósito principal es desarrollar su capacidad de autoaprendizaje.
TRABAJOS EN GRUPO	Se trata de actividades donde varios alumnos, como grupo, deben resolver determinada tarea o proyecto. A parte de la complejidad inherente al proyecto en sí, ese tipo de trabajos exige que el grupos de alumnos se divida y gestione la elaboración del proyecto por partes.
TUTORÍAS	Atención personalizada a los estudiantes mediante un conjunto de reuniones programadas dirigidas a grupos muy reducidos de alumnos en que éstos podrán además interactuar entre sí y con el profesor.



7. Recursos didácticos

RECURSOS DIDÁCTICOS	
BIBLIOGRAFÍA	Gamma et al. "Design Pattern". Ed. Addison Wesley 1994
	Larman, Craig "Applying UML and Patterns" Second Edition Prentice Hall 2002
	Parnas, David "On the Criteria TO Be Used in Decomposition Systems and Modules" Com. ACM Dec 1972 Vol. 15 Nº 12 pp. 1053-1058
	Medinilla, Nelson "Revisión de la Ingeniería de Software" Facultad de Informática. UPM.
	Leffingwell D., Widrig D. "Managing Software Requirements", 2nd Edition. Addison-Wesley, 2003.
	E. Yourdon. "Análisis Estructurado Moderno".Prentice-Hall Hispanoamericana, 1993.
	Gutiérrez; I. et al. "Diseño Orientado a Objetos" Facultad de Informática. UPM
RECURSOS WEB	Página web de la asignatura (http://)
	Sitio Moodle de la asignatura (http://)
EQUIPAMIENTO	Laboratorio
	Aula
	Sala de trabajo en grupo



8. Cronograma de trabajo de la asignatura

Semana	Actividades en Aula	Actividades en Laboratorio	Trabajo Individual	Trabajo en Grupo	Actividades de Evaluación	Otros
Semana 1 (10 horas)	<ul style="list-style-type: none"> Introducción IS, Clases teóricas y asignación de trabajos (4 horas) 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none"> Estudio bibliográfico (4 horas) 	<ul style="list-style-type: none"> Discusión colectiva y distribución de tareas (2 horas) 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none">
Semana 2 (10 horas)	<ul style="list-style-type: none"> La evaluación es en el aula 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none"> Estudio bibliográfico (2 horas) 	<ul style="list-style-type: none"> Discusión colectiva para preparar la presentación del trabajo (4 horas) 	Evaluación en aula de los trabajos realizados (4 h)	<ul style="list-style-type: none">
Semana 3 (10 horas)	<ul style="list-style-type: none"> Puzzle IR parte 1 (2 horas) 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none"> Lectura y búsqueda de información (3 horas) 	<ul style="list-style-type: none"> Ejercicio en grupo de Especificación de requisitos (4 horas) 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none"> Tutorías presenciales (1 hora)
Semana 4 (10 horas)	<ul style="list-style-type: none"> Puzzle IR parte 2 (2 horas) 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none"> Lectura y búsqueda de información (3 horas) 	<ul style="list-style-type: none"> Ejercicio en grupo de Validación de Requisitos (4 horas) 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none"> Tutorías presenciales (1 hora)
Semana 5 (10 horas)	<ul style="list-style-type: none"> Teoría Estructurado (2 h) 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none"> Lectura y estudio (1 h) Ejercicios indiv. (2 horas) 	<ul style="list-style-type: none"> Trabajo en grupo de Estructurado (4 horas) 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none"> Tutorías presenciales (1 hora)
Semana 6 (10 horas)	<ul style="list-style-type: none"> 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none"> Ejercicios individuales (3 horas) 	<ul style="list-style-type: none"> Trabajo en grupo de Estructurado (4 horas) 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none"> Tutorías presenciales (3 horas)



Semana 7 (10 horas)	•	•	• Ejercicios individuales (3 horas)	• Trabajo en grupo de Estructurado (4 horas)	•	• Tutorías presenciales (3 horas)
Semana 8 (10 horas)	•	•	• Ejercicios individuales (3 horas)	• Trabajo en grupo de Estructurado (4 horas)	•	• Tutorías presenciales (3 horas)
Semana 9 (10 horas)	• Teoría Diseño de Sistemas OO. Asignación de trabajos. (2h)	•	• Estudio individual. Trabajo en la solución. (5 horas)	• Análisis colectivo del trabajo y de las posibles soluciones. Decisión de una alternativa de solución inicial. Distribución de trabajo. (3 horas)	•	•
Semana 10 (10 horas)	• Teoría Diseño de Sistemas OO (2h)	•	• Estudio individual. Trabajo en la solución. (5 horas)	• Desarrollo conjunto de la solución. (3 horas)	•	•
Semana 11 (10 horas)	• La discusión es en el aula	•	• Estudio individual. Trabajo en la solución. (5 horas)	• Desarrollo conjunto de la solución. (3 horas)	• Discusión pública de los trabajos de diseño (2h)	•
Semana 12 (10 horas)	• Teoría Diseño de Sistemas OO. Patrones. (2h)	•	• Estudio individual. Trabajo en la solución. (5 horas)	• Desarrollo conjunto de la solución. (3 horas)	•	•
Semana 13 (10 horas)	• La discusión es en el aula	•	• Estudio individual. Trabajo en la solución. (5 horas)	• Desarrollo conjunto de la solución. (3 horas)	• Discusión pública de los trabajos de diseño (2h)	•
Semana 14 (10 horas)	• Teoría Diseño de Sistemas OO. Arquitecturas. (2h)	•	• Estudio individual. Trabajo en la solución. (5 horas)	• Desarrollo conjunto de la solución. (3 horas)	•	•



Semana 15 (10 horas)	<ul style="list-style-type: none">• La discusión es en el aula	<ul style="list-style-type: none">•	<ul style="list-style-type: none">• Preparación individual. de la evaluación (5 h)	<ul style="list-style-type: none">• Preparación conjunta de la evaluación. (3 horas)	<ul style="list-style-type: none">• Discusión pública de los trabajos de diseño (2 h)	<ul style="list-style-type: none">•
Semana 16 (10 horas)	<ul style="list-style-type: none">• La evaluación es en el aula	<ul style="list-style-type: none">•	<ul style="list-style-type: none">• Preparación individual. de la evaluación (5 h)	<ul style="list-style-type: none">• Preparación conjunta de la evaluación. (3 horas)	<ul style="list-style-type: none">• Evaluación pública final de los trabajos de diseño (2h)	<ul style="list-style-type: none">•

Nota: Para cada actividad se especifica la dedicación en horas que implica para el alumno.



POLITÉCNICA



UNIVERSIDAD POLITÉCNICA DE MADRID
FACULTAD DE INFORMÁTICA
Campus de Montegancedo
Boadilla del Monte. 28660 Madrid