



UNIVERSIDAD
POLITÉCNICA
DE MADRID

PROCESO DE
COORDINACIÓN DE LAS
ENSEÑANZAS PR/CL/001



E.T.S. de Ingenieros
Informaticos

ANX-PR/CL/001-01

GUÍA DE APRENDIZAJE

ASIGNATURA

105000023 - Ingenieria Del Software I

PLAN DE ESTUDIOS

10II - Grado En Ingenieria Informatica

CURSO ACADÉMICO Y SEMESTRE

2021/22 - Segundo semestre

Índice

Guía de Aprendizaje

1. Datos descriptivos.....	1
2. Profesorado.....	1
3. Conocimientos previos recomendados.....	2
4. Competencias y resultados de aprendizaje.....	2
5. Descripción de la asignatura y temario.....	3
6. Cronograma.....	6
7. Actividades y criterios de evaluación.....	9
8. Recursos didácticos.....	12
9. Otra información.....	14

1. Datos descriptivos

1.1. Datos de la asignatura

Nombre de la asignatura	105000023 - Ingenieria del Software I
No de créditos	6 ECTS
Carácter	Obligatoria
Curso	Tercero curso
Semestre	Sexto semestre
Período de impartición	Febrero-Junio
Idioma de impartición	Castellano
Titulación	10II - Grado en Ingenieria Informatica
Centro responsable de la titulación	10 - Escuela Tecnica Superior De Ingenieros Informaticos
Curso académico	2021-22

2. Profesorado

2.1. Profesorado implicado en la docencia

Nombre	Despacho	Correo electrónico	Horario de tutorías *
Andres Silva Vazquez	D5111	andres.silva@upm.es	L - 11:00 - 14:00 X - 11:00 - 14:00
Tomas San Feliu Gilabert (Coordinador/a)	D5107	tomas.sanfeliu@upm.es	X - 10:00 - 13:00 J - 10:00 - 13:00
Nelson Medinilla Martinez	D5109	nelson.medinilla@upm.es	M - 11:00 - 14:00 V - 11:00 - 14:00

* Las horas de tutoría son orientativas y pueden sufrir modificaciones. Se deberá confirmar los horarios de tutorías con el profesorado.

3. Conocimientos previos recomendados

3.1. Asignaturas previas que se recomienda haber cursado

- Programacion li

3.2. Otros conocimientos previos recomendados para cursar la asignatura

El plan de estudios Grado en Ingenieria Informatica no tiene definidos otros conocimientos previos para esta asignatura.

4. Competencias y resultados de aprendizaje

4.1. Competencias

CG-1/21 - Capacidad de resolución de problemas aplicando conocimientos de matemáticas, ciencias e ingeniería.

CG-3/4 - Saber trabajar en situaciones carentes de información y bajo presión, teniendo nuevas ideas, siendo creativo.

CG-6 - Capacidad de abstracción, análisis y síntesis

Ce 2 - Formalización y especificación de problemas reales cuya solución requiere el uso de la informática.

Ce 21 - Educir, analizar y especificar las necesidades de los clientes (empresas o usuarios individuales), plazos, medios disponibles y posibles condicionantes que pudieran afectar al sistema a desarrollar.

Ce 22 - Capacidad de aplicar sus conocimientos e intuición para diseñar el hardware/software que cumple unos requisitos especificados.

Ce 25 - Concebir y diseñar la arquitectura de un sistema software.

Ce 32 - Comprender el concepto de ciclo de vida, que abarca el significado de sus fases (planificación, desarrollo, instalación y evolución), las consecuencias para el desarrollo de todos los aspectos de los sistemas informáticos (el software, el hardware, y el interfaz humano-máquina), y la relación entre la calidad y la gestión del ciclo de vida.

Ce 39 - Conocer y aplicar los principios de la ingeniería del software y de sus tecnologías para garantizar que las implementaciones de software sean robustas, fiables y apropiadas para la audiencia a la que van destinadas.

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.

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.

4.2. Resultados del aprendizaje

RA247 - Diseñar y construir sistemas informáticos capaces de resolver problemas para los que no se conoce solución.

RA248 - Capacidad de llevar a cabo la definición y gestión de requisitos.

5. Descripción de la asignatura y temario

5.1. Descripción de la asignatura

La asignatura está organizada entorno a los cinco temas siguientes:

1. Introducción a la Ingeniería del Software. Breve recorrido por los conceptos básicos que configuran la ingeniería del software. Se presenta además el concepto de ciclo de vida y se presentan metodologías Ágiles.

2.- Ingeniería de Requisitos. Introducción a la IR. Educación, análisis y Negociación. La gestión de requisitos y herramientas. Casos de Uso.

3.-Diseño estructurado de sistemas software. El método de Yourdon. Acontecimientos. Elaboración de DFD.

4.-Diseño Orientado a Objetos de sistemas software. El modelo de objetos y su contraste con el estructurado. Criterios de selección.Revisión de los conceptos de objetos, mensajes y clases. UML. Principio de ocultación.Revisión de los conceptos de herencia y polimorfismo. Representación en UML. Principio de sustitución de Liskov.

5.Arquitecturas Software. Se presenta que es una arquitectura software. Principales actividades, patrones y mecanismos de comunicación. La actividad de documentación de la arquitectura y la actividad de evaluación de la arquitectura.

5.2. Temario de la asignatura

1. Introducción a la Ingeniería del Software

1.1. 1.1 Conceptos básicos de la Ingeniería de Software.

1.2. Modelos y métodos

1.3. 1.2 Enfoques ágiles

2. Ingeniería de Requisitos

2.1. 2.1 Introducción a la IR

2.2. Educción, Análisis y Negociación

2.3. Especificación y Validación

2.4. Gestión de Requisitos y Herramientas

2.5. Casos de Uso

3. Diseño estructurado de sistemas software

3.1. El método de Yourdon

3.2. Acontecimientos

3.3. Elaboración de DFDs

4. Diseño Orientado a Objetos de sistemas Software

4.1. El modelo de objetos y su contraste con el estructurado. Criterios de selección.

4.2. Revisión de los conceptos de objetos, mensajes y clases. UML. Principio de ocultación.

4.3. Revisión de los conceptos de herencia y polimorfismo. Representación en UML. Principio de sustitución de Liskov.

5. Arquitecturas Software

5.1. Conceptos básicos de Arquitecturas. Arquitecturas ágiles

5.2. Patrones Arquitectónicos

5.3. Documentación arquitectura

5.4. Evaluación arquitectura

6. Cronograma

6.1. Cronograma de la asignatura *

Sem	Actividad presencial en aula	Actividad presencial en laboratorio	Tele-enseñanza	Actividades de evaluación
1	Introducción a IS Duración: 04:00 LM: Actividad del tipo Lección Magistral			
2	Introducción a IS Duración: 04:00 LM: Actividad del tipo Lección Magistral			Test de conocimientos adquiridos ET: Técnica del tipo Prueba Telemática Evaluación continua No presencial Duración: 01:00
3	Puzzle IR Duración: 02:00 LM: Actividad del tipo Lección Magistral Ejercicio en Grupo de Especificación Duración: 02:00 AC: Actividad del tipo Acciones Cooperativas			
4	Teoría y Práctica de IR Duración: 02:00 LM: Actividad del tipo Lección Magistral Trabajo en grupo de educación de requisitos Duración: 02:00 AC: Actividad del tipo Acciones Cooperativas			
5	Teoría de estructurado Duración: 02:00 LM: Actividad del tipo Lección Magistral Trabajo en grupo de estructurado Duración: 02:00 AC: Actividad del tipo Acciones Cooperativas			
6	Trabajo en grupo de estructurado Duración: 02:00 AC: Actividad del tipo Acciones Cooperativas Discusión y practicas en aula Duración: 02:00 LM: Actividad del tipo Lección Magistral			Entrega trabajo estructurado TG: Técnica del tipo Trabajo en Grupo Evaluación continua No presencial Duración: 04:00
7	Teoría Diseño de Sistemas OO Duración: 02:00 LM: Actividad del tipo Lección Magistral			

8	<p>Teoría Diseño de Sistemas OO Duración: 02:00 LM: Actividad del tipo Lección Magistral</p> <p>Análisis colectivo del trabajo Duración: 02:00 AC: Actividad del tipo Acciones Cooperativas</p>			<p>Test de conocimientos adquiridos OO1 ET: Técnica del tipo Prueba Telemática Evaluación continua No presencial Duración: 01:00</p>
9	<p>Teoría Diseño de Sistemas OO Duración: 02:00 LM: Actividad del tipo Lección Magistral</p> <p>Análisis colectivo del trabajo Duración: 02:00 AC: Actividad del tipo Acciones Cooperativas</p>			
10	<p>Teoría Diseño de Sistemas OO Duración: 02:00 LM: Actividad del tipo Lección Magistral</p> <p>Análisis colectivo del trabajo Duración: 02:00 AC: Actividad del tipo Acciones Cooperativas</p>			<p>Test de conocimientos adquiridos OO2 ET: Técnica del tipo Prueba Telemática Evaluación continua No presencial Duración: 01:00</p>
11	<p>Teoría Arquitectura Duración: 02:00 LM: Actividad del tipo Lección Magistral</p> <p>Análisis colectivo del trabajo Duración: 02:00 AC: Actividad del tipo Acciones Cooperativas</p>			
12	<p>Teoría Arquitectura Duración: 02:00 LM: Actividad del tipo Lección Magistral</p> <p>Análisis colectivo del trabajo Duración: 01:00 AC: Actividad del tipo Acciones Cooperativas</p>			<p>Test de conocimientos adquiridos ET: Técnica del tipo Prueba Telemática Evaluación continua No presencial Duración: 01:00</p>
13	<p>Teoría Arquitectura Duración: 02:00 LM: Actividad del tipo Lección Magistral</p> <p>Análisis colectivo del trabajo Duración: 02:00 AC: Actividad del tipo Acciones Cooperativas</p>			
14	<p>Teoría Arquitectura Duración: 02:00 LM: Actividad del tipo Lección Magistral</p> <p>Análisis colectivo del trabajo Duración: 01:00 AC: Actividad del tipo Acciones Cooperativas</p>			<p>Test de conocimientos adquiridos ET: Técnica del tipo Prueba Telemática Evaluación continua No presencial Duración: 01:00</p>

15	Teoría Arquitectura Duración: 02:00 LM: Actividad del tipo Lección Magistral Análisis colectivo del trabajo Duración: 02:00 AC: Actividad del tipo Acciones Cooperativas			Discusión trabajos sobre arquitectura software TG: Técnica del tipo Trabajo en Grupo Evaluación continua No presencial Duración: 02:00
16				
17				Prueba global EX: Técnica del tipo Examen Escrito Evaluación continua Presencial Duración: 02:00 Prueba global EX: Técnica del tipo Examen Escrito Evaluación sólo prueba final Presencial Duración: 02:00

Para el cálculo de los valores totales, se estima que por cada crédito ECTS el alumno dedicará dependiendo del plan de estudios, entre 26 y 27 horas de trabajo presencial y no presencial.

* El cronograma sigue una planificación teórica de la asignatura y puede sufrir modificaciones durante el curso derivadas de la situación creada por la COVID-19.

7. Actividades y criterios de evaluación

7.1. Actividades de evaluación de la asignatura

7.1.1. Evaluación continua

Sem.	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
2	Test de conocimientos adquiridos	ET: Técnica del tipo Prueba Telemática	No Presencial	01:00	5%	3 / 10	Ce 41 Ce 32
6	Entrega trabajo estructurado	TG: Técnica del tipo Trabajo en Grupo	No Presencial	04:00	18%	3 / 10	CG-3/4 CG-1/21 Ce 21 CG-6 Ce 9 Ce 32 Ce 2
8	Test de conocimientos adquiridos OO1	ET: Técnica del tipo Prueba Telemática	No Presencial	01:00	15%	3 / 10	CG-1/21 CG-6 Ce 9 Ce 22
10	Test de conocimientos adquiridos OO2	ET: Técnica del tipo Prueba Telemática	No Presencial	01:00	15%	3 / 10	CG-1/21 CG-6 Ce 9 Ce 22
12	Test de conocimientos adquiridos	ET: Técnica del tipo Prueba Telemática	No Presencial	01:00	2%	3 / 10	Ce 25
14	Test de conocimientos adquiridos	ET: Técnica del tipo Prueba Telemática	No Presencial	01:00	2%	3 / 10	Ce 25
15	Discusion trabajos sobre arquitectura software	TG: Técnica del tipo Trabajo en Grupo	No Presencial	02:00	15%	3 / 10	Ce 25 Ce 39
17	Prueba global	EX: Técnica del tipo Examen Escrito	Presencial	02:00	28%	4 / 10	Ce 9 Ce 22 Ce 25 Ce 32 Ce 39 CG-3/4 CG-1/21 Ce 41

7.2. Criterios de evaluación

Evaluación Sólo prueba final

La evaluación tipo prueba final consta de un examen que aborda cuestiones teóricas y prácticas de la asignatura. La nota de este examen corresponde al 100% de la nota de la asignatura. La nota mínima para aprobar es un 5,0. Los alumnos interesados en seguir este sistema de evaluación deberán comunicarlo siguiendo el procedimiento y plazo especificado por la Ordenación Académica de la Escuela.

Convocatoria extraordinaria.

La convocatoria extraordinaria consta de un único examen que aborda la totalidad del tema. Se plantearán cuestiones teóricas y prácticas de la asignatura.

Al igual que a evaluación de sólo prueba final, la evaluación de la convocatoria extraordinaria requieren que se obtenga al menos un 5 (sobre 10).

Evaluación Continua

La asignatura se basa en el sistema de evaluación continua con prueba global de evaluación. La evaluación de cada tema se basará en diferentes pruebas parciales con contenido teórico-prácticos a entregar por los alumnos, cuyo contenido y forma se anunciará vía Moodle.

Los dos test de conocimientos de Objetos formarán una sola prueba parcial. Los dos test de conocimientos del tema de Arquitectura Software junto con el trabajo de Arquitectura Software formarán una sola prueba parcial.

En cada prueba parcial se exigirá una nota mínima de 3 puntos para asegurar que los alumnos alcanzan los logros de aprendizaje planteados en esta guía. Las pruebas parciales tendrán un componente teórico y práctico.

Si no se alcanzan los 3 puntos en una prueba parcial se considerará no ha alcanzado los niveles exigidos de aprendizaje y por tanto no habrán alcanzado los requisitos mínimos para aprobar la asignatura.

Indicar que la prueba global tipo examen escrito de la semana 17 requiere una puntuación mínima de 4 para tenerse en cuenta en la calificación final.

Las actividades evaluables tendrán carácter obligatorio. La no realización de las actividades supondrá el abandono del sistema de evaluación continua, pasando al sistema de evaluación final.

Adicionalmente, la evaluación podrá ser mediante presentaciones orales, complementada con documentos de trabajo realizados por los alumnos.

Los requisitos mínimos para aprobar la asignatura requieren haber superado la nota mínima en todas y cada una de las actividades de evaluación que las tengan. La compensación en caso de no superar el valor de 5,0 sólo se podrá realizar cuando se haya superado la nota mínima exigida en la actividad de evaluación. No es posible aprobar la asignatura por partes.

Durante el semestre se publicarán las notas obtenidas por los alumnos en las pruebas parciales, que junto con la valoración global final concretarán la calificación final de la asignatura.

En la evaluación de convocatoria extraordinaria se seguirán los mismos criterios de la evaluación de sólo prueba final.

Plazos de Entrega.

Las fechas exactas establecidas como plazo máximo para la entrega de las diferentes pruebas serán publicadas en el Moodle de la asignatura. Los plazos de entrega estarán escalonados a lo largo del curso con el objetivo de facilitar el desarrollo de las prácticas en paralelo con el del temario y maximizar así su beneficio. En ningún caso se admitirán prácticas entregadas fuera de plazo

8. Recursos didácticos

8.1. Recursos didácticos de la asignatura

Nombre	Tipo	Observaciones
Sitio Moodle	Recursos web	http://moodle.upm.es/titulaciones/oficiales/course/view.php?id=1580
Biblioteca Safari accesible desde VPN de la UPM	Recursos web	https://learning.oreilly.com/home/

Engineering Software Products: An Introduction to Modern Software Engineering	Bibliografía	Sommerville, Editorial Pearson, 2019
Business Analysis, Requirements, and Project Management: A Guide for Computing Students	Bibliografía	Cox, Auerbach 2021
Managing Software Requirements the Agile Way	Bibliografía	Heath, Packt Publishing, 2020.
Design Pattern	Bibliografía	Gamma et al. ?Design Pattern?. Ed. Addison Wesley 1994
Leffingwell	Bibliografía	Leffingwell D., Widrig D. ?Managing Software Requirements?, 2nd Edition. Addison-Wesley, 2003.
Yourdon	Bibliografía	E. Yourdon. "Análisis Estructurado Moderno".Prentice-Hall Hispanoamericana, 1993.
Software Engineering at Google	Bibliografía	Wright, H., Winters, T. D., & Manshreck, T. (2020).
Arquitectura de Software	Bibliografía	Cervantes, H., Velasco, P., Castro, L.,Arquitectura de Software: Conceptos y ciclo de desarrollo, Editorial: Cengage Learning, ISBN978-607522-456-5
Software Architecture in Practice	Bibliografía	Len Bass, Paul Clements, Rick Kazman, Ed. Addison-Wesley, 2021
Software Systems Architecture: Working with Stakeholders Using Viewpoints and Perspectives	Bibliografía	Rozanski, N., & Woods, E. (2012).. Addison-Wesley.
Documenting Software Architectures: Views and Beyond	Bibliografía	P. Clements, F. Bachmann, L. Bass, D. Garlan, J. Ivers, R. Little, P. Merson, R. Nord & J. Stafford (2010)
Designing Software Architectures: A Practical Approach	Bibliografía	H. Cervantes, R. Kazman (2016). Addison-Wesley Professional

9. Otra información

9.1. Otra información sobre la asignatura

Actuación ante copias y otros comportamientos fraudulentos

Los derechos y deberes de los estudiantes están desarrollados en los Estatutos de la Universidad y en el Estatuto del Estudiante. El Estatuto del Estudiante especifica como deber "abstenerse de la utilización o cooperación en procedimientos fraudulentos en las pruebas de evaluación, en los trabajos que se realicen o en documentos oficiales de la universidad". Si se detecta que algún alumno ha copiado en alguna prueba o algún grupo ha copiado en la realización de las pruebas, se le calificará con la puntuación de cero al estudiante o estudiantes implicados, de acuerdo a la normativa de evaluación vigente.

En particular, en el caso de las prácticas, se tendrá en cuenta que la responsabilidad del trabajo está compartida por todos los miembros del grupo, en caso de detectar alguna copia la norma se aplicará a todos los miembros de los grupos involucrados en la copia (tanto los que copian como los que se dejan copiar).

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 Ordenación de Estudios de la Escuela de conformidad con lo estipulado en dicha Normativa (más información en <http://www.fi.upm.es/?pagina=1147>).