

GUIA DOCENTE DE LA ASIGNATURA

PROGRAMACION Y DISEÑO ORIENTADO A OBJETOS

MÓDULO	MATERIA	CURSO	SEMESTRE	CRÉDITOS	TIPO
Formación específica de Rama	Programación e Ingeniería del Software	2º	3º	6	Obligatoria
PROFESOR(ES)		DIRECCIÓN COMPLETA DE CONTACTO PARA TUTORÍAS (Dirección postal, teléfono, correo electrónico, etc.)			
<ul style="list-style-type: none"> • María del Mar Abad Grau: Grupos de prácticas B2, C1 y C2. ▣ Ana Anaya Morito: Grupo de teoría D, Grupos de prácticas A1, D1 y D2. ▣ Zoraida Callejas Carrión: Grupo de teoría C, Grupos de prácticas C3. ▣ Nuria Medina Medina: Grupo de teoría B, Grupos de prácticas B1. ▣ María José Rodríguez Fórtiz: Grupo de teoría A, Grupos de Prácticas A3. ▣ Francisco Velasco Anguita: Grupos de prácticas B3 y A2. 		Dpto. Lenguajes y Sistemas Informáticos, 3ª planta, E. T. S. Ingeniería Informática y Telecomunicación. Despachos: 25, 31, 23, 25, 24, 33. Correos electrónicos: mabad@ugr.es , anaya@ugr.es , zoraida@ugr.es , nmedina@ugr.es , mjfortiz@ugr.es , fvelasco@ugr.es			
		HORARIO DE TUTORÍAS			
		María del Mar Abad Grau: http://lsi.ugr.es/lsi/mabad Ana Anaya Morito: http://lsi.ugr.es/lsi/anaya Zoraida Callejas Carrión: http://lsi.ugr.es/lsi/zoraida Nuria Medina Medina: http://lsi.ugr.es/lsi/nmedina María José Rodríguez Fórtiz: http://lsi.ugr.es/lsi/mjfortiz Francisco Velasco Anguita: http://lsi.ugr.es/lsi/fvelasco			
GRADO EN EL QUE SE IMPARTE		OTROS GRADOS A LOS QUE SE PODRÍA OFERTAR			
Grado en Ingeniería Informática		Doble grado Matemáticas e Informática			



PRERREQUISITOS Y/O RECOMENDACIONES (Si ha lugar)

Los alumnos no tendrán que tener asignaturas, materias o módulos aprobados como requisito indispensable para cursar la asignatura. No obstante, se recomienda la superación de los contenidos y adquisición de competencias de las materias de formación básica.

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

Conceptos, técnicas y notaciones.
Clases y objetos.
Polimorfismo.
Herencia.
Diseño con polimorfismo y herencia.
Reutilización.
Patrón MVC.

COMPETENCIAS GENERALES Y ESPECÍFICAS

Competencias Específicas de la Asignatura

R1. Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.

R8. Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

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.

E8. Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.

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 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.

T3. Capacidad de comunicación oral y escrita en el ámbito académico y profesional con especial énfasis, en la redacción de documentación técnica.



- 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.
- T7.** Capacidad de comunicación en lengua extranjera, particularmente en inglés.
- T8.** Capacidad de trabajo en equipo.
- T9.** Capacidad para el aprendizaje autónomo así como iniciativa y espíritu emprendedor.
- T10.** Motivación por la calidad y la mejora continua, actuando con rigor, responsabilidad y ética profesional.
- T11.** Capacidad para adaptarse a las tecnologías y a los futuros entornos actualizando las competencias profesionales.
- T12.** Capacidad para innovar y generar nuevas ideas.
- T14.** Respeto a los derechos fundamentales y de igualdad entre hombres y mujeres.
- T15.** Capacidad para proyectar los conocimientos, habilidades y destrezas adquiridos para promover una sociedad basada en los valores de la libertad, la justicia, la igualdad y el pluralismo.

OBJETIVOS (EXPRESADOS COMO RESULTADOS DE APRENDIZAJE)

- Comprender el origen de la orientación a objetos y justificar por qué se utilizan los objetos como clave para descomponer los sistemas en módulos en lugar de la funcionalidad.
- Conocer los principales conceptos de la tecnología de objetos y su aplicación en el diseño de software.
- Utilizar los principios de diseño de la orientación a objetos y aplicarlos en la resolución de problemas prácticos.
- Conocer notaciones básicas de diseño que permitan representar las clases, sus relaciones y los objetos.
- Comprender el concepto de polimorfismo y de ligadura dinámica, y saber aplicar ambos en el diseño de clases.
- Conocer el concepto de herencia, sus distintas formas y los problemas originados en el diseño de lenguajes de programación.
- Comprender las relaciones complejas entre el sistema de tipos, la herencia y el polimorfismo.
- Adquirir habilidades para aplicar los conceptos de herencia, polimorfismo y ligadura dinámica para realizar diseños e implementaciones reutilizables.
- Saber aplicar en la resolución de problemas concretos los principios de diseño con el objetivo de especificar jerarquías de clases y comprender los criterios para escoger entre una relación de herencia o de clientela.
- Manejar entornos de programación orientados a objetos que ejemplifiquen lenguajes con diferentes características.
- Conocer el patrón MVC y estar capacitado para desarrollar un supuesto práctico haciendo uso de él.



TEMARIO DETALLADO DE LA ASIGNATURA

TEMARIO DE TEORIA**Tema 1: Conceptos básicos de la programación y el diseño orientados a objetos.**

- 1.1. Conceptos básicos de orientación a objetos.
- 1.2. Técnicas para diseño y lenguajes de programación orientados a objetos.

Tema 2: Clases, objetos y mensajes

- 2.1. Concepto y características de las clases.
- 2.2. Diagramas estructurales para la representación de clases.
- 2.3. Concepto y características de los objetos y los mensajes.
- 2.4. Diagramas de interacción entre objetos.

Tema 3: Reutilización y polimorfismo

- 3.1. Mecanismos de reutilización de código.
- 3.2. Representación en UML de los mecanismos de reutilización.
- 3.3. Polimorfismo.

Tema 4: Patrón de diseño MVC (Modelo/Vista/Controlador)

- 4.1. Definición y diseño del patrón MVC.
- 4.2. Desarrollo de la vista y del controlador.

TEMARIO DE PRACTICAS

Práctica 1 (P1). Ejercicios básicos de programación orientados a objetos.

Práctica 2 (P2). Implementación de la estructura de clases de un sistema orientado a objetos, dado su diseño estructural.

Práctica 3 (P3). Implementación de la funcionalidad del sistema, dado su diseño dinámico.

Práctica 4 (P4). Ampliación del sistema incluyendo herencia y polimorfismo.

Práctica 5 (P5). Implementación de una interfaz gráfica de usuario básica siguiendo el patrón MVC.



BIBLIOGRAFÍA

BIBLIOGRAFÍA FUNDAMENTAL:

- T. Budd. *An Introduction to Object-Oriented Programming*. Addison-Wesley. 2002.
- D. Pilone, N. Pitman. *UML 2.0 in a Nutshell*. O'Reilly Media. 2005.
- J. Rumbaugh, I. Jacobson, G. Booch. *The Unified Modeling Language Reference Manual*. Addison-Wesley. 2004.

BIBLIOGRAFÍA COMPLEMENTARIA:

- F. Durán, F. Gutiérrez, E. Pimentel. *Programación orientada a objetos con Java*. Thomson. 2007.
- D. Thomas, C. Fowler, A. Hunt. *Programming Ruby* (Vol. 13). Pragmatic Bookshelf. 2004.
- G. Booch, R. Maksimchuk, M. Engle, B. Young, J. Conallen, K. Houston. *Object-oriented analysis and design with applications, 3ª ed.* Addison-Wesley Professional (ACM Online Book Program). 2007.
- C. Larman. *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, 3ª ed.* Prentice Hall. 2005.

ENLACES RECOMENDADOS

Enlaces de la asignatura: https://groups.diigo.com/group/pdoo_ugr

Otros enlaces:

- JAVA: <http://www.java.com>
- Ruby: <http://www.ruby-lang.org/es/>

METODOLOGÍA DOCENTE

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

Contenido en ECTS: 30 horas presenciales (1.2 ECTS)

Competencias: E1, E3, E4, E5, E6, E7, E8, E12, T1, T3, T4, T5, T6, T10, T11, R6, R8, R11, R14



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

Contenido en ECTS: 15 horas presenciales (0.6 ECTS)

Competencias: E1, E3, E4, E5, E7, E8, E9, E10, E11, T1, T2, T3, T4, T5, T6, T8, T10, T11, T12, T13, T14, T15, R6, R8, R11, R14

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

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

Competencias: E1, E4, E7, E9, T1, T2, T3, T4, T5, T6, T7, T9, T10, T11, T12, T14, T15, R6, R8, R11, R14

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

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

Competencias: E1, E2, E4, E5, E9, T1, T2, T3, T4, T5, T6, T7, T8, T10, T11, T12, T14, T15, R6, R8, R11, R14

5. Tutorías académicas (grupo pequeño)

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

Competencias: E1, E4, E7, T2, T3, T4, T5, T10, T11, T12, T14, R6, R8, R11, R14

RÉGIMEN DE ASISTENCIA A CLASE

La asistencia a clase tanto de teoría como de prácticas será la suficiente para que el profesor pueda hacer un seguimiento adecuado del aprendizaje y evolución del alumno.

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

Se utilizarán alguna o algunas de las siguientes técnicas de evaluación:

- Para la parte teórica se realizarán exámenes periódicos, así como diferentes entregas y



evaluaciones 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.
- 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 siguiente tabla indica el peso de la evaluación de cada actividad formativa:

Actividades Formativas	Ponderación
Parte Teórica	50%
Parte Práctica	50%

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 4,5 (sobre 10).

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/>

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.



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.

