

MÓDULO	MATERIA	CURSO	SEMESTRE	CRÉDITOS	TIPO
Formación específica de Rama	Programación e Ingeniería del Software	2º	4º	6	Obligatoria
PROFESORES*¹			DIRECCIÓN COMPLETA DE CONTACTO PARA TUTORÍAS (Dirección postal, teléfono, correo electrónico, etc.)		
<ul style="list-style-type: none"> • Miguel Lastra Leidinger: Teoría • Francisco Velasco Anguita: Grupo de prácticas PDO01 • Miguel Lastra Leidinger: Grupo de prácticas PDO02 <p>COORDINADOR DE LA ASIGNATURA: Miguel Lastra Leidinger</p>			Dpto.Lenguajes y Sistemas Informáticos, 3ª planta, E. T. S. Ingeniería Informática y Telecomunicación. Despacho 33. Teléfono: 958248427 y 958246144 Correo electrónico: mlastral@ugr.es y fvelasco@ugr.es		
			HORARIO DE TUTORÍAS Y/O ENLACE A LA PÁGINA WEB DONDE PUEDAN CONSULTARSE LOS HORARIOS DE TUTORÍAS*		
			https://lsi.ugr.es/lsi/mlastral https://lsi.ugr.es/lsi/fvelasco		
GRADO EN EL QUE SE IMPARTE			OTROS GRADOS A LOS QUE SE PODRÍA OFERTAR		
Doble grado en Ingeniería Informática y Matemáticas					

¹* Consulte posible actualización en Acceso Identificado > Aplicaciones > Ordenación Docente.

(†) Esta guía docente debe ser cumplimentada siguiendo la "Normativa de Evaluación y de Calificación de los estudiantes de la Universidad de Granada" ([http://secretariageneral.ugr.es/pages/normativa/fichasugr/ncg7121/!](http://secretariageneral.ugr.es/pages/normativa/fichasugr/ncg7121/))



PRERREQUISITOS Y/O RECOMENDACIONES (si procede)

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, teniendo especial importancia la superación de las materias de “Fundamentos de Programación” y “Metodología de la Programación”.

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 Generales del título

- **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
- **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 específicas del módulo

- **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 Básicas

- **CB2.** Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio.

Competencias Transversales

- **T1.** Capacidad de organización y planificación así como capacidad de gestión de la Información

OBJETIVOS (EXPRESADOS COMO RESULTADOS ESPERABLES DE LA ENSEÑANZA)

- 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 TEÓRICO:

- **Tema 1:** Conceptos básicos de la programación y el diseño orientados a objetos.
- **Tema 2:** Clases, objetos y mensajes
- **Tema 3:** Reutilización y polimorfismo
- **Tema 4:** Conceptos Complementarios

TEMARIO PRÁCTICO:

- **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)

- **Descripción:** Presentación en el aula de los conceptos propios de la materia haciendo uso de metodología positiva 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:** E5, E8, E9, R1, R8, CB2 y T1

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:** 25 horas presenciales (1 ECTS)
- **Competencias:** E5, E8, E9, R1, R8, CB2 y T1

3. 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:** E5, E8, E9, R1, R8, CB2 y T1 .



4. 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:** E5, E8, E9, R1, R8, CB2 y T1 .

5. 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) Orientar 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:** E5, E8, E9, R1, R8, CB2 y T1.

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

En la modalidad de evaluación continua:

- Para la parte teórica se realizarán 3 exámenes periódicos.
- Para la parte práctica se realizarán prácticas de laboratorio. Se evalúan mediante exámenes basados en entregas parciales de la práctica. En estas prácticas se realizará la implementación por etapas de un sistema software.

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).

Para los estudiantes que se acojan a la **convocatoria extraordinaria** habrá un examen teórico y un examen práctico. Para la convocatoria extraordinaria del mismo curso académico, la nota de teoría o prácticas, de la convocatoria ordinaria, superior o igual a 5 se mantiene, debiendo examinarse solo de la parte no superada.

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: https://lsi.ugr.es/lsi/normativa_examenes

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.

DESCRIPCIÓN DE LAS PRUEBAS QUE FORMARÁN PARTE DE LA EVALUACIÓN ÚNICA FINAL ESTABLECIDA EN LA "NORMATIVA DE EVALUACIÓN Y DE CALIFICACIÓN DE LOS ESTUDIANTES DE LA UNIVERSIDAD DE GRANADA"

Para los estudiantes que se acojan a la evaluación única final habrá un examen teórico y un examen práctico. El peso de cada examen en la calificación final será del 50%

INFORMACIÓN ADICIONAL

Régimen de asistencia a clase:

- La asistencia a las clases teóricas no será obligatoria, exceptuando las sesiones en las que se programen pruebas de evaluación
- La asistencia a las clases prácticas no será obligatoria, exceptuando las sesiones en las que se programen pruebas de evaluación.

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.

