

GUÍA DOCENTE

PROGRAMACION ORIENTADA A OBJETOS

GRADO EN INGENIERÍA DEL SOFTWARE

CURSO 2026-27

Fecha de publicación: 23-06-2026



I.-Identificación de la Asignatura

Tipo	OBLIGATORIA
Período de impartición	2 curso, 1Q semestre
Nº de créditos	6
Idioma en el que se imparte	Castellano

II.-Presentación

El objetivo de la asignatura es presentar los **fundamentos de la programación y el diseño orientados a objetos**. La asignatura aborda la enseñanza de la orientación a objetos tanto desde el punto de vista del diseño (usando **UML**), como desde el de la implementación (usando **Java**). Al final de la asignatura, el alumnado deberá ser capaz de:

- Conocer y comprender los conceptos básicos de programación orientada a objetos.
- Conocer y comprender los elementos esenciales del lenguaje Java
- Conocer y comprender conceptos básicos de diseño de programas orientados a objetos.
- Diseñar aplicaciones usando el paradigma de la programación orientada a objetos y el lenguaje UML.
- Codificar programas usando el paradigma de la programación orientada a objetos en el lenguaje Java.

Como **requisito previo** para cursar esta asignatura se necesitarán los conocimientos y competencias relacionados con la asignatura **Introducción a la Programación** detallados más adelante.

Se **recomienda** al alumno haber superado con éxito la asignatura **Estructuras de Datos**.

II.B-Presentación en inglés

The objective of the course is to present the fundamentals of programming and object-oriented design. The course addresses the teaching of object orientation from both the design perspective (using UML) and the implementation perspective (using Java). By the end of the course, students should be able to: - Understand and comprehend the basic concepts of object-oriented programming. - Understand and comprehend the essential elements of the Java language. - Understand and comprehend basic concepts of object-oriented program design. - Design applications using the object-oriented programming paradigm and the UML language. - Code programs using the object-oriented programming paradigm in the Java language. As a prerequisite for this course, knowledge and skills related to the Introduction to Programming course detailed below are required. It is recommended that students have successfully completed the Data Structures course.

III.-Resultados de Aprendizaje



CG05. 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

CG08. 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.

CG09. 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 del Software.

CB04. Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

CE06. Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

CE07. Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.

CE08. Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

CE17. Capacidad para diseñar y evaluar interfaces persona computador que garanticen la accesibilidad y usabilidad a los sistemas, servicios y aplicaciones informáticas.

- Leyenda para las titulaciones no adaptadas al RD 822/21: **CB** - competencias básicas, **CG** - competencias generales, **CE** - competencias específicas, **CT** - competencias transversales.
- Leyenda para las titulaciones adaptadas al RD 822/21: **CON** - conocimientos, **COM** - competencias, **HAB** - habilidades.



IV.-Contenido

IV.A.-Temario de la asignatura

Capítulo 1 - Introducción a la programación orientada a objetos

- 1.1. Complejidad del software: origen y tratamiento, Herramientas para tratar la complejidad, Evolución de los lenguajes de programación, Paradigmas de programación
- 1.2. Programación orientada a objetos: Abstraer, Encapsular, Jerarquizar, Modularizar, Tipo, Concurrencia, Persistencia

Capítulo 2 - Introducción al lenguaje Java

- 2.1. Introducción: Breve historia de Java, La máquina virtual de Java, Compiladores de Java, Entornos de desarrollo para Java, El programa hola mundo, Los comentarios, Juego de instrucciones
- 2.2. Expresiones básicas y control de flujo: Tipos primitivos, Conversiones entre tipos primitivos, Operadores, Control del flujo
- 2.3. Las clases: Las propiedades en detalle, Los métodos en detalle, Creación de objetos, Destrucción de objetos, Bloques de inicialización, Los modificadores de control de acceso a los miembros en detalle, Los modificadores de uso de los miembros en detalle, Modificadores en la definición de clases generales, Clases internas, Definición de arrays de objetos y de tipos primitivos
- 2.4. Los paquetes

Capítulo 3 - Herencia y genericidad en POO

- 3.1. La herencia de clases: Herencia en clases internas, Redefinición de miembros, La herencia y los constructores
- 3.2. Polimorfismo dinámico: El casting de referencias a objetos
- 3.3. Las interfaces

Capítulo 4 - El paquete java.lang

- 4.1. La clase Object: La clase ClassLoader
- 4.2. Clases para el manejo de cadenas de caracteres
- 4.3. Las excepciones: Tipos de excepciones, El uso práctico de las excepciones
- 4.4. Sobrecarga de los métodos básicos de object: Interfaz Cloneable, Los métodos equals y hashCode, Interfaz Comparable
- 4.5. La concurrencia: La palabra reservada synchronized, Mecanismos de compartición y comunicación
- 4.6. Los tipos enumerados
- 4.7. Envolturas

Capítulo 5 - Diseño de Clases

- 5.1. Diseño top-down apoyado en UML: Relaciones de dependencia entre objetos., Relaciones de asociación entre objetos, Relaciones entre clases2
- 5.2. Diseño Dirigido por Pruebas: Pruebas automáticas, Principios del Diseño Dirigido por Pruebas, Objetivos y ventajas del enfoque TDD

Capítulo 6 - Genericidad y estructuras de datos predefinidas

- 6.1. La genericidad: Clases e interfaces genéricas, Métodos con tipos parametrizados
- 6.2. Collection, El EnumSet
- 6.3. Map
- 6.4. Iteradores
- 6.5. La clase Collections
- 6.6. El resto del paquete java.util

Capítulo 7 - Entrada salida en POO

- 7.1. Streams de bytes
- 7.2. Streams de caracteres: Streams del sistema, StreamTokenizer
- 7.3. Manejo de ficheros
- 7.4. La interfaz Serializable

Capítulo 8 - Interfaces de usuario

- 8.1 Componentes del GUI de Java
- 8.2 Listeners
- 8.3 Containers
- 8.4 Layouts



IV.B.-Actividades formativas	
Tipo	Descripción
Lecturas	Clases magistral en directo (AF. 1)
Tutorías académicas	Clases prácticas con ordenador de resolución de ejercicios (AF. 2 y AF. 3)
Trabajos individuales	Proyecto de programación (AF. 9)
Lecturas	Pruebas escritas con ordenador (AF 5.)
Otras actividades	Tutorías sobre resolución de problemas de diseño y programación (AF. 7)
Lecturas	Lecturas de material de apoyo: libros y diapositivas (AF. 8)
Otras actividades	Preparación de pruebas escritas (AF.10)
Trabajos colectivos	Proyecto de diseño (AF. 9)

V.-Tiempo de trabajo del alumnado (30h grado y 25h máster)

Clases teóricas	14
Clases de resolución de ejercicios, problemas, casos, etc.	14
Prácticas en laboratorios experimentales, tecnológicos, clínicos, campo, etc.	24
Realización de pruebas	8
Tutorías académicas	40
Actividades relacionadas: jornadas, seminarios, etc.	8
Preparación de clases teóricas	30
Preparación de prácticas/ejercicios/casos	30
Preparación de pruebas	12
Total de horas de trabajo del alumnado	180

VI.-Metodología y plan de trabajo

Tipo	Periodo	Contenido
Prácticas	Semana 1 a Semana 15	MD3 y MD6 - Simultáneamente a cada uno de los temas se realizan clases de problemas y casos prácticos, con y sin ordenador.
Pruebas	Semana 6 a Semana 8	MD8 - Evaluación de los temas 1, 2 y 3. (Examen tipo test 1)
Pruebas	Semana 11 a Semana 13	MD8 - Evaluación de los temas 4, 5 y 6. (Examen tipo test 2)
Trabajos colectivos	Semana 10 a Semana 13	MD4 y MD5 - Los alumnos realizan en grupo fuera de clase una práctica de diseño. De dicha práctica se realiza una primera reunión de explicación de requisitos, a la semana se hace una reunión de seguimiento, una semana más tarde los alumnos entregan la práctica y finalmente, la siguiente semana se realiza una sesión de revisión del resultado correcto.



Trabajos individuales	Semana 13 a Semana 15	MD4 y MD5 - Cada alumno realiza una práctica/proyecto de implementación. Dicha práctica/proyecto se explica en una primera reunión con los alumnos, y se desarrolla en un par de sesiones más, pero los alumnos la desarrollarán fuera de clase, dejándoles un tiempo suficiente. Finalmente, el día del examen de la convocatoria ordinaria o extraordinaria, se le propondrá al alumno la realización de ciertas modificaciones a la aplicación por él desarrollada. El éxito o fracaso en la realización de dichas modificaciones, junto a las penalizaciones acumuladas (por las desviaciones respecto al diseño a la funcionalidad o al estilo requeridos), constituirán la nota de la prueba práctica de programación.
Clases Teóricas	Semana 1 a Semana 15	MD1 y MD2 - Durante el curso se imparten lecciones magistrales de los Temas 1 a Tema 7. Dichas clase se acompañan de diapositivas y un libro de texto escrito por los profesores

VII.-Método de evaluación

El modelo de evaluación general es la evaluación continua, tal como establece el Reglamento de evaluación de los resultados de aprendizaje de la Universidad Rey Juan Carlos.

Deberán utilizarse todos los sistemas de evaluación establecidos para la asignatura en la memoria de la titulación, excepto aquellos que tuviesen una ponderación mínima del 0%, que podrán utilizarse en los cursos académicos en los que el profesorado lo considere oportuno. Cada uno de los sistemas de evaluación podrá ser aplicado mediante una o más actividades de evaluación, coherentes con ese sistema. Ninguna de las actividades de evaluación podrá superar individualmente el 60% de la calificación global de la asignatura.

La suma de las actividades de evaluación no revaluables no podrá superar el 40% de la calificación global de la asignatura y, en general, no deberían tener nota mínima (salvo en el caso de actividades de carácter práctico en las que, estrictamente, no pudieran reproducirse en la convocatoria extraordinaria las condiciones de evaluación de la convocatoria ordinaria).

El alumnado que no consiga superar la asignatura en la convocatoria ordinaria, o no se hayan presentado, podrán presentarse a la convocatoria extraordinaria únicamente a las actividades de evaluación revaluables no superadas.

El uso de herramientas basadas en inteligencia artificial está prohibido en cualquier ejercicio y prueba de evaluación de la asignatura salvo que el profesor lo indique de forma explícita en el enunciado o en el material de esta. En caso de que esté permitido, se darán indicaciones precisas de qué usos están permitidos y cuáles no.

La distribución y características de las actividades de evaluación son las que se describen a continuación.

VII.A.- Descripción de las actividades de evaluación y su ponderación



Sistema de evaluación	Revaluable en	Ponderación	Actividad de evaluación	Nota mínima	Contenidos	Fecha
SE.1 - Pruebas escritas de respuesta abiertas o tipo test	Sí (mismo formato que en convocatoria ordinaria)	50%	Examen tipo Test (10%)	1	Temas 1, 2 y 3	Semanas 6 a 8
			Examen tipo Test (10%)	1	Temas 4, 5 y 6	Semanas 11 a 13
			Examen de pregunta abierta (30%)	5	Diseño orientado a objetos (todos los temas)	Fecha oficial de convocatoria ordinaria
SE.2 - Resolución de problemas y casos prácticos	No*	20%	Trabajo en grupo de diseño en UML	-	Diseño orientado a objetos (temas 1 y 5)	Entrega en la semana 12 o 13
SE.3 - Prácticas con ordenador	Sí (mismo formato que en convocatoria ordinaria)	30%	Examen de pregunta abierta de programación basada en la práctica **	5	Programación orientada a objetos (todos los temas)	Fecha oficial de convocatoria ordinaria

Cálculo de la nota final

- (*) La no revaluabilidad de la parte de resolución de problemas y casos prácticos, que corresponde a un 20% de la nota, se deriva de que dicha práctica implica una carga de trabajo continuada e importante por parte del alumno (varias semanas de desarrollo) con seguimiento y apoyo por parte del profesorado (exposición de requisitos, seguimiento del diseño y exposición del nuevo diseño). Es por ello que no es posible repetirla en un periodo posterior a la convocatoria ordinaria. Los estudiantes con dispensa académica deberá asistir al menos a dos tutorías: una donde se les explicará la práctica y otra donde se les hará un seguimiento de la misma.
- (**) El alumno desarrollará en Java de manera individual una aplicación original fuera del tiempo de clase contando para ello de un periodo de tiempo amplio y suficiente. Dicha aplicación estará sujeta a unos requisitos de funcionamiento, de estilo del código y de diseño previamente establecidos. Las desviaciones a peor leves respecto a dichos requisitos supondrán una penalización en la nota final, las desviaciones grandes a peor supondrán el suspenso de la práctica. Cualquier copia, total o parcial, de código (de una IA, de una web, de un compañero...) también será motivo de suspenso. Finalmente, el día del examen de la convocatoria ordinaria o extraordinaria, se le propondrá al alumno la realización de ciertas modificaciones a la aplicación por él desarrollada. El éxito o fracaso en la realización de dichas modificaciones, junto a las penalizaciones acumuladas, constituirán la nota de la prueba práctica de programación.
- La nota final se calcula como la media ponderada de las notas de las pruebas evaluables según los porcentajes indicados, siempre y cuando se hayan superado con la nota mínima indicada para cada una de ellas.
- Para aprobar la asignatura hay que obtener al menos un 5 en la media ponderada de las cinco pruebas.
- Si alguna de las pruebas evaluables no se ha superado con la nota mínima necesaria para hacer media, la nota final de la asignatura será el mínimo entre la media ponderada de las cinco pruebas y 4.
- Si no se ha presentado a alguna de las pruebas evaluables en la convocatoria ordinaria que requiera de nota mínima, la nota final será el mínimo entre la media ponderada de las cinco pruebas y 4. Además, de cara al cómputo de la nota, las pruebas no realizadas serán evaluadas con la nota de 0.
- Si no se ha presentado a ninguna prueba evaluable, la nota final será "No presentado".

Convocatoria extraordinaria

- La convocatoria extraordinaria no es sustitutiva de la evaluación continua. Es decir, el estudiante solo podrá ser revaluado de aquellas actividades que no haya superado en la convocatoria ordinaria (bien por haber superado la nota mínima, o bien por no haberse presentado).
- Si un estudiante hubiese superado la nota mínima establecida en todas las actividades revaluables y aun así la calificación global no le alcanzara para aprobar la asignatura, de forma excepcional podrá ser revaluado en convocatoria extraordinaria de alguna o de todas las actividades de evaluación revaluables en las que, habiendo superado la nota mínima, hubiera obtenido una calificación inferior a 5. Si no tuviese ninguna actividad reevaluable en esta situación, o si fuese imposible aprobar solo



teniendo en cuenta las actividades revaluables en dicha situación, los estudiantes podrán presentarse a las actividades revaluables que considere, según su elección, para conseguir aprobar la asignatura.

- Para el cálculo de la nota final en esta convocatoria se utilizará la calificación de las pruebas superadas en convocatoria ordinaria y las notas obtenidas en las pruebas revaluadas (aunque sean menores).
- El cálculo de la nota final se realiza tal y como se indica en el apartado "Cálculo de la nota final".
- La revaluación de las pruebas se realizará en la fecha oficial indicada para la convocatoria extraordinaria.

Convocatoria adelantada

- En la convocatoria adelantada de septiembre se realizarán todas las pruebas en la fecha oficial de la convocatoria.
- Ante la imposibilidad de realizar la práctica de diseño los porcentajes de las diferentes pruebas serán: 15% examen tipo test 1, 15% examen tipo test 2, 30% examen de diseño, 30% examen de modificaciones de código sobre la práctica.
- Ante la imposibilidad de realizar una nueva práctica de codificación, las modificaciones de código del examen se basarán en la práctica del año anterior.

Conducta académica

En el caso de **fraude académico** en alguna actividad de evaluación, se otorgará una calificación de cero puntos en dicha actividad lo que, para aquellas actividades con nota mínima superior a cero implica el suspenso en la convocatoria correspondiente.

El uso de la IA generativa está prohibido en cualquier ejercicio y prueba de evaluación de la asignatura.

Se recuerda además que, atendiendo al artículo 9.d) de la **Normativa de Convivencia de la Universidad Rey Juan Carlos** (<https://www.urjc.es/images/Universidad/Presentacion/normativa/normativa%20convivencia%20universitaria.pdf>) el **fraude académico** en alguna actividad de evaluación se considera **falta grave**. Las sanciones correspondientes a las faltas graves, según el artículo 11 de la referida normativa, son la expulsión temporal de la Universidad, o la pérdida al derecho de la convocatoria ordinaria.

VII.B.- Evaluación del alumnado con dispensa académica de asistencia a clase

La concesión de Dispensa Académica de Asistencia a Clase (DAAC) no implica que se quede automáticamente eximido de participar en las actividades de evaluación continua ni en las actividades formativas presenciales de asistencia obligatoria establecidas en la guía docente. Una vez concedida la dispensa, el alumnado deberá contactar con el docente, que podría proponerle las adaptaciones que considere convenientes, siempre que garanticen la adquisición y adecuada evaluación de los resultados de aprendizaje previstos. El alumnado deberá mantener a lo largo de curso una comunicación fluida con el docente para que este le proporcione información sobre las fechas en que se realizarán esas actividades formativas y de evaluación, en caso de que su programación no estuviese ya fijada y a disposición del alumnado en el momento de la concesión de la dispensa.

Asignatura con posibilidad de dispensa: Sí

VII.C.- Revisión y reclamación de las actividades de evaluación

Se realizará conforme al Reglamento de evaluación de los resultados de aprendizaje de la Universidad Rey Juan Carlos.

VII.D.- Alumnado con discapacidad o necesidades educativas especiales

Con el fin de garantizar la igualdad de oportunidades, la no discriminación, la accesibilidad universal y la mayor garantía de éxito académico, las adaptaciones curriculares para alumnado con discapacidad o con necesidades educativas especiales serán pautadas por la Unidad de Atención a Personas con Discapacidad, de acuerdo con la normativa que regula el servicio de Atención a Estudiantes con Discapacidad, aprobada por Consejo de Gobierno de la Universidad Rey Juan Carlos.

Para que esas adaptaciones puedan realizarse, será requisito la emisión de un informe de adaptaciones curriculares por parte de dicha Unidad, por lo que el alumnado con discapacidad o necesidades educativas especiales deberán contactar con este servicio para analizar conjuntamente las distintas opciones.

VII.E.- Conducta académica, integridad y honestidad académica



La Universidad Rey Juan Carlos está plenamente comprometida con los más altos estándares de integridad y honestidad académica, por lo que estudiar en la URJC supone asumir y suscribir los valores de integridad y la honestidad académica recogidos en el Código Ético de la Universidad.

Para acompañar este proceso, la Universidad dispone de la Normativa de Convivencia de la Universidad Rey Juan Carlos y de diferentes herramientas (antiplagio, supervisión) que ofrecen una garantía colectiva para el completo desarrollo de estos valores esenciales.

Las herramientas basadas en inteligencia artificial pueden mejorar la experiencia de aprendizaje, pero debe utilizarse de forma ética, legal, honesta y transparente. Por este motivo, deben seguirse las recomendaciones de uso adecuado indicadas por los docentes de la asignatura.

VIII.-Recursos y materiales didácticos

Bibliografía básica

Diseñar y programar todo es empezar, Jose F. Vélez, Alberto Peña. Patxi Gortazar y Ángel Sánchez, Ed. Dickinson, 2011.

Bibliografía complementaria

Programación II, Teoría y práctica del módulo de programación orientada a objetos L. Fernández Muñoz, F. Arroyo Montoro. Departamento de publicaciones de la EUI de Madrid, 1997.

Análisis y diseño estructurado y orientado a objetos de sistemas informáticos. De Amescua Seco y otros. McGraw Hill, 2003.

Programación en Java: Fundamentos de programación y principios de diseño. Fernando Berzal. Universidad de Granada
<http://elvex.ugr.es/decsai/java/>

Utilización de UML en Ingeniería del software con objetos y componentes. Stevens, Pooley. Addison-Wesley, 2010.

Java, how to program. Deitel & Deitel. Ninth Edition. Prentice Hall, 2012. Cómo programar en Java. Deitel & Deitel. Séptima edición. Prentice Hall, 2008.

Programación orientada a objetos con Java. Francisco Durán, Francisco Gutiérrez, Ernesto Pimentel. Thomson 2007.

Programación en Java 6. Algoritmos, programación orientad a objetos e interfaz gráfica de usuario. Joyanes, Zahonero. McGrawHill, 2011.

IX.-Profesorado

Nombre y apellidos	DAVID ORTEGA DEL CAMPO
Correo electrónico	david.ortega.delcampo@urjc.es
Departamento	Informática y Estadística
Campus de impartición	Móstoles
Categoría	Profesor/a Permanente Laboral
Titulación académica	Doctor
Responsable de asignatura	No
Horario de tutorías	Para consultar las tutorias póngase en contacto con el/la profesor/a a través de correo electrónico
Nº de Quinquenios	1
Nº de Sexenios	0
Nº de Sexenios de transferencia	0
Nº de evaluaciones positivas Docencia	2
Nombre y apellidos	JOSÉ FRANCISCO VÉLEZ SERRANO
Correo electrónico	jose.velez@urjc.es
Departamento	Informática y Estadística



Campus de impartición	Móstoles
Categoría	Catedrático/a de Universidad
Titulación académica	Doctor
Responsable de asignatura	Si
Horario de tutorías	Para consultar las tutorías póngase en contacto con el/la profesor/a a través de correo electrónico
Nº de Quinquenios	4
Nº de Sexenios	3
Nº de Sexenios de transferencia	1
Nº de evaluaciones positivas Docencia	6
Nombre y apellidos	MARÍA TERESA GONZÁLEZ DE LENA ALONSO
Correo electrónico	mariateresa.gonzalezdelena@urjc.es
Departamento	Informática y Estadística
Campus de impartición	Móstoles
Categoría	Profesor/a Colaborador/a
Titulación académica	Doctor
Responsable de asignatura	No
Horario de tutorías	Para consultar las tutorías póngase en contacto con el/la profesor/a a través de correo electrónico
Nº de Quinquenios	4
Nº de Sexenios	0
Nº de Sexenios de transferencia	0
Nº de evaluaciones positivas Docencia	4