

DEPARTAMENTO DE INFORMÁTICA
CICLO FORMATIVO DE GRADO SUPERIOR EN DESARROLLO DE
APLICACIONES WEB
MÓDULO PROFESIONAL: PROGRAMACIÓN
PROGRAMACIÓN DOCENTE
CURSO 2022-2023

Contenido

1. Introducción.....	3
1.1. Estructura y organización del ciclo formativo	4
1.2. Objetivos generales del ciclo.....	5
1.3. Cualificaciones Profesionales y Unidades de Competencia	7
1.4. Competencias profesionales, personales y sociales	8
2. Contribución del módulo para el logro de las competencias y objetivos establecidos en el currículo del ciclo formativo.....	10
3. Contenidos.....	11
3.1 UNIDADES DE TRABAJO	11
U.T.01. Identificación de un programa informático.....	11
U.T.02. Aplicación de las estructuras de control	12
U.T.03. Programación orientada a objetos (POO) en Java	12
U.T.04. Nociones avanzadas de programación orientada a objetos.....	13
U.T.05. Estructuras de almacenamiento.....	14
U.T.06. Clases genéricas y control de excepciones	14
U.T.07. Lectura y escritura de información.....	15
U.T.08 Interfaces gráficas	15
U.T.09 Control de acceso y mantenimiento de bases de datos relacionales	16
3.2 Contenidos Mínimos.....	16
3.3 TEMPORALIZACIÓN.....	19
4. Metodología.....	19
5. Materiales curriculares.....	20
6. Criterios de evaluación	21
7. Procedimientos de evaluación	24

7.1. Criterios para la elaboración de la prueba de la convocatoria extraordinaria.....	25
8. Instrumentos de evaluación.....	26
9. Criterios de calificación.....	27
9.1 EVALUACIÓN TRIMESTRAL	27
9.2 EVALUACIÓN FINAL	28
9.3 RECUPERACIÓN	28
Recuperación ordinaria.....	28
9.4 EXAMEN FINAL	28
9.5 Evaluación extraordinaria	28
9.6 Evaluación de pendientes.....	28
9.7 Programación de recuperación para alumnos con imposibilidad de aplicación del proceso de evaluación continua.....	29
10. Medidas de atención a la diversidad.....	29
10.1. Medidas de refuerzo	29
10.2. Seguimiento y evaluación de las medidas adoptadas para la atención a la diversidad	29
11. Actividades complementarias y extraescolares.....	29
12. Contribución del módulo a la educación en valores y a la igualdad de derechos y oportunidades entre las personas.....	30
13. Indicadores de logro y procedimiento de evaluación de la aplicación y desarrollo de la programación docente.....	30

1. Introducción

De conformidad con lo dispuesto en el artículo 10, apartado 1, de la Ley orgánica 5/2002, de 19 de junio, de las Cualificaciones y de la Formación Profesional y en el artículo 6 del Real Decreto 1538/2006, de 15 de diciembre, por el que se establece la ordenación general de la Formación Profesional del sistema educativo, el Gobierno ha dictado el Real Decreto 686/2010, de 20 de mayo, por el que se establece el título de **Técnico Superior en Desarrollo de Aplicaciones Web** y se fijan sus enseñanzas.

Este **Ciclo Formativo de Grado Superior**, denominado **Desarrollo de Aplicaciones Web**, está dirigido a personas que ejercen su actividad en empresas o entidades públicas o privadas tanto por cuenta ajena como propia, desempeñando su trabajo en el área de desarrollo de aplicaciones informáticas relacionadas con entornos Web (intranet, extranet e internet). Las ocupaciones y puestos de trabajo más relevantes son: Programador Web, Programador Multimedia y Desarrollador de aplicaciones en entornos Web.

Este ciclo formativo ofrece al alumnado posibilidades de empleo como trabajador o trabajadora por cuenta ajena o por cuenta propia, dado que se configura un módulo específico para desarrollar la iniciativa empresarial y las características propias de los entornos Web imperantes en este sector, lo que alentará la iniciativa de los alumnos y alumnas en orden a crear su propia empresa.

Cabe señalar, asimismo, que la formación de base que ofrece este ciclo formativo es fundamental dada la continua actualización y renovación en el sector de servicios informáticos y de tratamiento de la información, en temas como los estándares y herramientas disponibles para el desarrollo, implantación y mantenimiento de aplicaciones web.

Por otra parte, las enseñanzas de este ciclo formativo reflejan la tendencia del sector de migrar e integrarse en la utilización de tecnologías orientadas al trabajo en red (en la web), mostrándose esta realidad reflejada en las enseñanzas de este ciclo formativo que agrupa conocimientos y capacidades fundamentales en este área.

Finalmente, cabe destacar que en la regulación del currículo del ciclo formativo de Grado Superior de Formación Profesional conducente a la obtención del título de Técnica Superior o Técnico Superior en Desarrollo de Aplicaciones Web se han intentado superar estereotipos, prejuicios y discriminaciones por razón de sexo, así como fomentar el aprendizaje de la resolución pacífica de conflictos, tal y como se prescribe en la Ley Orgánica 1/2004, de 28 de diciembre de Medidas de Protección Integral contra la violencia de Género, así como en la Ley orgánica 3/2007, de 22 de marzo, para la igualdad efectiva de Mujeres y Hombres, que señala que el sistema educativo incluirá entre sus fines la educación en el respeto de los derechos y libertades fundamentales y la igualdad de derechos y oportunidades entre hombres y mujeres.

1.1. Estructura y organización del ciclo formativo

El presente ciclo formativo se desarrollará a lo largo de dos años académicos y, según se establece en el artículo 2 del Real Decreto 686/2010, de 20 de mayo, tendrá una duración de 2.000 horas.

El ciclo viene definido en la siguiente tabla:

TÍTULO	Técnica Superior o Técnico Superior en Desarrollo de Aplicaciones Web
NORMA	Real Decreto 686/2010, de 20 de mayo (BOE de 12/06/2010)
NIVEL	Formación profesional de Grado Superior
DURACIÓN TOTAL	2000 horas
FAMILIA PROFESIONAL	Informática y Comunicaciones
REFERENTE EUROPEO	CINE-5b (Clasificación Internacional Normalizada de la Educación)
CÓDIGO DEL CICLO	IFC303LOE
DENOMINACIÓN CICLO	Ciclo Formativo de Grado Superior de Desarrollo de aplicaciones web

Las enseñanzas correspondientes a este ciclo, con sus correspondientes módulos, adscripción al primer o segundo año académico y duración expresada en horas totales, son las que figuran en la siguiente tabla:

MÓDULOS PROFESIONALES	CURSO	HORAS 1º	HORAS 2º
Lenguajes de marcas y sistemas de gestión de información	1	128	
Sistemas informáticos	1	160	
Bases de datos	1	192	
Programación	1	224	
Entornos de desarrollo	1	96	
Formación y orientación laboral	1	96	
Lengua extranjera para uso profesional	1	64	
Desarrollo web en entorno cliente	2		150
Desarrollo web en entorno servidor	2		150
Despliegue de aplicaciones web	2		110
Diseño de interfaces web	2		132
Proyecto de desarrollo de aplicaciones web	2		30
Empresa e iniciativa emprendedora	2		88
Formación en centros de trabajo	2		380
TOTAL de Horas por Curso		960	1040

Total de Horas del Ciclo: 2000 horas.

1.2. Objetivos generales del ciclo

Los objetivos generales de este ciclo formativo son los siguientes:

- a) Ajustar la configuración lógica analizando las necesidades y criterios establecidos para configurar y explotar sistemas informáticos.
- b) Identificar las necesidades de seguridad verificando el plan preestablecido para aplicar técnicas y procedimientos relacionados.
- c) Instalar módulos analizando su estructura y funcionalidad para gestionar servidores de aplicaciones.
- d) Ajustar parámetros analizando la configuración para gestionar servidores de aplicaciones.
- e) Interpretar el diseño lógico, verificando los parámetros establecidos para gestionar bases de datos.
- f) Seleccionar lenguajes, objetos y herramientas, interpretando las especificaciones para desarrollar aplicaciones web con acceso a bases de datos.
- g) Utilizar lenguajes, objetos y herramientas, interpretando las especificaciones para desarrollar aplicaciones web con acceso a bases de datos.
- h) Generar componentes de acceso a datos, cumpliendo las especificaciones, para integrar contenidos en la lógica de una aplicación web.
- i) Utilizar lenguajes de marcas y estándares web, asumiendo el manual de estilo, para desarrollar interfaces en aplicaciones web
- j) Emplear herramientas y lenguajes específicos, siguiendo las especificaciones, para desarrollar componentes multimedia.
- k) Evaluar la interactividad, accesibilidad y usabilidad de un interfaz, verificando los criterios preestablecidos, para Integrar componentes multimedia en el interfaz de una aplicación.
- l) Utilizar herramientas y lenguajes específicos, cumpliendo las especificaciones, para desarrollar e integrar componentes software en el entorno del servidor web.

- m) Emplear herramientas específicas, integrando la funcionalidad entre aplicaciones, para desarrollar servicios empleables en aplicaciones web.
- n) Evaluar servicios distribuidos ya desarrollados, verificando sus prestaciones y funcionalidad, para integrar servicios distribuidos en una aplicación web.
- ñ) Verificar los componentes de software desarrollados, analizando las especificaciones, para completar el plan de pruebas.
- o) Utilizar herramientas específicas, cumpliendo los estándares establecidos, para elaborar y mantener la documentación de los procesos.
- p) Establecer procedimientos, verificando su funcionalidad, para desplegar y distribuir aplicaciones.
- q) Programar y realizar actividades para gestionar el mantenimiento de los recursos informáticos.
- r) Analizar y utilizar los recursos y oportunidades de aprendizaje relacionadas con la evolución científica, tecnológica y organizativa del sector y las tecnologías de la información y la comunicación, para mantener el espíritu de actualización y adaptarse a nuevas situaciones laborales y personales.
- s) Desarrollar la creatividad y el espíritu de innovación para responder a los retos que se presentan en los procesos y organización de trabajo y de la vida personal.
- t) Tomar decisiones de forma fundamentada analizando las variables implicadas, integrando saberes de distinto ámbito y aceptando los riesgos y la posibilidad de equivocación en las mismas, para afrontar y resolver distintas situaciones, problemas o contingencias.
- u) Desarrollar técnicas de liderazgo, motivación, supervisión y comunicación en contextos de trabajo en grupo para facilitar la organización y coordinación de equipos de trabajo.
- v) Aplicar estrategias y técnicas de comunicación adaptándose a los contenidos que se van a transmitir, la finalidad y a las características de los receptores, para asegurar la eficacia en los procesos de comunicación.
- x) Evaluar situaciones de prevención de riesgos laborales y de protección ambiental, proponiendo y aplicando medidas de prevención personales y colectivas, de acuerdo a la normativa aplicable en los procesos del trabajo, para garantizar entornos seguros.

y) Identificar y proponer las acciones profesionales necesarias para dar respuesta a la accesibilidad universal y al diseño para todos

z) Identificar y aplicar parámetros de calidad en los trabajos y actividades realizados en el proceso de aprendizaje para valorar la cultura de la evaluación y de la calidad y ser capaces de supervisar y mejorar procedimientos de gestión de calidad.

aa) Utilizar procedimientos relacionados con la cultura emprendedora, empresarial y de iniciativa profesional, para realizar la gestión básica de una pequeña empresa o emprender un trabajo.

ab) Reconocer sus derechos y deberes como agente activo en la sociedad, teniendo en cuenta el marco legal que regula las condiciones sociales y laborales para participar como ciudadano democrático.

1.3. Cualificaciones Profesionales y Unidades de Competencia

Esta programación didáctica se encuentra enmarcada en el proyecto curricular del ciclo.

Competencia general:

Desarrollar, implantar, y mantener aplicaciones web, con independencia del modelo empleado y utilizando tecnologías específicas, garantizando el acceso a los datos de forma segura y cumpliendo los criterios de accesibilidad, usabilidad y calidad exigidas en los estándares establecidos.

Cualificaciones profesionales completas:

Desarrollo de aplicaciones con tecnologías Web IFC154_3 (Real Decreto 1087/2005, de 16 de septiembre), que comprende las siguientes unidades de competencia:

UC0491_3 Desarrollar elementos software en el entorno cliente.

UC0492_3 Desarrollar elementos software en el entorno servidor.

UC0493_3 Implementar, verificar y documentar aplicaciones web en entornos internet, intranet y extranet.

Cualificaciones profesionales incompletas:

Programación en lenguajes estructurados de aplicaciones de gestión IFC155_3 (Real Decreto 1087/2005, de 16 de septiembre).

UC0223_3 Configurar y explotar sistemas informáticos.

UC0226_3 Programar bases de datos relacionales.

Programación con lenguajes orientados a objetos y bases de datos relacionales IFC 080_3
(Real Decreto. 295/2004, de 20 de febrero).

UC0223_3 Configurar y explotar sistemas informáticos.

UC0226_3 Programar bases de datos relacionales.

1.4. Competencias profesionales, personales y sociales

Las competencias profesionales, personales y sociales de este título, tal y como recoge el Real Decreto 686/2010, de 20 de mayo, en su art. 5, son las que se relacionan a continuación:

- a) Configurar y explotar sistemas informáticos, adaptando la configuración lógica del sistema según las necesidades de uso y los criterios establecidos.
- b) Aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad.
- c) Gestionar servidores de aplicaciones adaptando su configuración en cada caso para permitir el despliegue de aplicaciones web.
- d) Gestionar bases de datos, interpretando su diseño lógico y verificando integridad, consistencia, seguridad y accesibilidad de los datos.
- e) Desarrollar aplicaciones web con acceso a bases de datos utilizando lenguajes, objetos de acceso y herramientas de mapeo adecuados a las especificaciones.
- f) Integrar contenidos en la lógica de una aplicación web, desarrollando componentes de acceso a datos adecuados a las especificaciones.
- g) Desarrollar interfaces en aplicaciones web de acuerdo con un manual de estilo, utilizando lenguajes de marcas y estándares web.
- h) Desarrollar componentes multimedia para su integración en aplicaciones web, empleando herramientas específicas y siguiendo las especificaciones establecidas.
- i) Integrar componentes multimedia en el interface de una aplicación web, realizando el análisis de interactividad, accesibilidad y usabilidad de la aplicación.

- j) Desarrollar e integrar componentes software en el entorno del servidor web, empleando herramientas y lenguajes específicos, para cumplir las especificaciones de la aplicación.
- k) Desarrollar servicios para integrar sus funciones en otras aplicaciones web, asegurando su funcionalidad.
- l) Integrar servicios y contenidos distribuidos en aplicaciones web, asegurando su funcionalidad.
- m) Completar planes de pruebas verificando el funcionamiento de los componentes software desarrollados, según las especificaciones.
- n) Elaborar y mantener la documentación de los procesos de desarrollo, utilizando herramientas de generación de documentación y control de versiones.
- ñ) Desplegar y distribuir aplicaciones web en distintos ámbitos de implantación, verificando su comportamiento y realizando modificaciones.
- o) Gestionar y/o realizar el mantenimiento de los recursos de su área en función de las cargas de trabajo y el plan de mantenimiento.
- p) Adaptarse a las nuevas situaciones laborales, manteniendo actualizados los conocimientos científicos, técnicos y tecnológicos relativos a su entorno profesional, gestionando su formación y los recursos existentes en el aprendizaje a lo largo de la vida y utilizando las tecnologías de la información y la comunicación.
- q) Resolver situaciones, problemas o contingencias con iniciativa y autonomía en el ámbito de su competencia, con creatividad, innovación y espíritu de mejora en el trabajo personal y en el de los miembros del equipo.
- r) Organizar y coordinar equipos de trabajo, supervisando el desarrollo del mismo, con responsabilidad, manteniendo relaciones fluidas y asumiendo el liderazgo, así como, aportando soluciones a los conflictos grupales que se presentan.
- s) Comunicarse con sus iguales, superiores, clientes y personas bajo su responsabilidad utilizando vías eficaces de comunicación, transmitiendo la información o conocimientos adecuados, y respetando la autonomía y competencia de las personas que intervienen en el ámbito de su trabajo.

- t) Generar entornos seguros en el desarrollo de su trabajo y el de su equipo, supervisando y aplicando los procedimientos de prevención de riesgos laborales y ambientales de acuerdo con lo establecido por la normativa y los objetivos de la empresa.
- u) Supervisar y aplicar procedimientos de gestión de calidad, de accesibilidad universal y de diseño para todos, en las actividades profesionales incluidas en los procesos de producción o prestación de servicios.
- v) Realizar la gestión básica para la creación y funcionamiento de una pequeña empresa y tener iniciativa en su actividad profesional con sentido de la responsabilidad social.
- x) Ejercer sus derechos y cumplir con las obligaciones derivadas de su actividad profesional, de acuerdo con lo establecido en la legislación vigente, participando activamente en la vida económica, social y cultural.

2. Contribución del módulo para el logro de las competencias y objetivos establecidos en el currículo del ciclo formativo

El módulo profesional PROGRAMACIÓN del ciclo formativo de grado superior de Desarrollo de Aplicaciones Web, tiene una duración de 224 horas, distribuidas en 7 sesiones semanales y está encuadrado en el primer curso.

Este módulo profesional contiene parte de la formación necesaria para desempeñar la función de programación de aplicaciones de propósito general en lenguajes orientados a objetos.

La función de programación de aplicaciones de propósito general en lenguajes orientados a objetos incluye aspectos como:

- El desarrollo de programas organizados en clases aplicando los principios de la programación orientada a objetos.
- La utilización de interfaces para la interacción de la aplicación con el usuario.
- La identificación, análisis e integración de librerías para incorporar funcionalidades específicas a los programas desarrollados.
- El almacenamiento y recuperación de información en sistemas gestores de bases de datos relacionales y orientados a objetos.

Las actividades profesionales asociadas a esta función se aplican en el desarrollo y la adaptación de programas informáticos de propósito general en lenguajes orientados a objetos.

La formación del módulo contribuye a alcanzar los objetivos generales e), j), q) del ciclo formativo y las competencias a), e), f), i), j), v), y) y z) del título.

Las líneas de actuación en el proceso de enseñanza-aprendizaje que permiten alcanzar los objetivos del módulo versarán sobre:

- La interpretación y aplicación de los principios de la programación orientada a objetos.
- La evaluación, selección y utilización de herramientas y lenguajes de programación orientados a objetos
- La utilización de las características específicas de lenguajes y entornos de programación en el desarrollo de aplicaciones informáticas.
- La identificación de las funcionalidades aportadas por los sistemas gestores de bases de datos y su incorporación a los programas desarrollados.
- La documentación de los programas desarrollados.
- El desarrollo y documentación de aplicaciones que puedan ser usadas por el mayor número posible de personas, independientemente de su edad y características físicas, siguiendo las pautas del diseño para todos.

Los objetivos del módulo están expresados en los siguientes resultados de aprendizaje:

- RA1.- Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.
- RA2.- Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.
- RA3.- Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.
- RA4.- Desarrolla programas organizados en clases, analizando y aplicando los principios de la programación orientada a objetos.
- RA5.- Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases.
- RA6.- Escribe programas que manipulan información, seleccionando y utilizando tipos avanzados de datos.
- RA7.- Desarrolla programas, aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.
- RA8.- Utiliza bases de datos orientadas a objetos, analizando sus características y aplicando técnicas para mantener la persistencia de la información.
- RA9.- Gestiona información almacenada en bases de datos relacionales, manteniendo la integridad y consistencia de los datos.

3. Contenidos

3.1 UNIDADES DE TRABAJO

U.T.01. Identificación de un programa informático

Se relaciona con el resultado de aprendizaje: RA1

CONTENIDOS

- Introducción.
- Estructura y bloques fundamentales.
- Identificadores.
- Tipos de datos.
- Conversiones de tipo.
- Variables.
- Constantes.
- Sentencias.
- Bloques de código.
- Operadores y expresiones.
- Comentarios.
- Palabras reservadas.

ACTIVIDADES DE ENSEÑANZA-APRENDIZAJE:

- Identificar los bloques que componen la estructura de un programa informático.
- Crear proyectos de desarrollo de aplicaciones.
- Utilizar entornos integrados de desarrollo.
- Identificar los distintos tipos de variables y la utilidad específica de cada uno.
- Desarrollar un programa para crear y utilizar variables referenciándolas a un ámbito concreto.
- Crear y utilizar constantes y literales.
- Clasificar, reconocer y utilizar en expresiones los operadores del lenguaje.
- Comprobar el funcionamiento de las conversiones de tipo explícitas e implícitas.
- Introducir comentarios en el código.

U.T.02. Aplicación de las estructuras de control

Se relaciona con los resultados de aprendizaje: RA1, RA3 y RA5

CONTENIDOS

- API o biblioteca de clases en java
- Entrada/Salida de datos
- Estructuras de control de flujo
 - Estructuras de selección
 - Estructuras de repetición
 - Sentencias de ruptura o estructuras de salto
- Depuración de programas.
- Introducción a los métodos.

ACTIVIDADES DE ENSEÑANZA-APRENDIZAJE

- Buscar en la API de java a qué paquete pertenece una clase concreta.
- Escribir y probar código que haga uso de estructuras de selección o alternativa.
- Escribir y probar código que haga uso de estructuras repetitivas.
- Reconocer las posibilidades de las sentencias de salto.
- Crear programas ejecutables utilizando diferentes estructuras de control.
- Probar y depurar los programas.
- Crear programas utilizando contadores y acumuladores para observar la diferencia entre ambos.
- Crear programas ejecutables de forma modularizada, utilizando diferentes estructuras de control.
- Crear el mismo programa ejecutable utilizando diferentes estructuras alternativas, repetitivas y diferentes tipos de métodos.
- Utilizar la consola para realizar operaciones sencillas de entrada y salida de información.
- Poner enunciado a programas escritos en código fuente.

U.T.03. Programación orientada a objetos (POO) en Java

Se relaciona con los resultados de aprendizaje: R.A.-1 y R.A.-2

CONTENIDOS

- Características de la POO.

- Concepto de clase. Declaración.
- Modificadores de acceso.
- Modificadores de contenido.
- Atributos o propiedades.
- Métodos en objetos. Tipos.
- Constructores y destructores.
- Métodos de instancia y de clase.
- Referencia this.
- Objetos. Declaración y creación.
- Sobrecarga de métodos.
- Paso de objetos como parámetros.
- Cadena de caracteres. Clase String y StringBuffer.
 - Métodos de la clase String y StringBuffer.
 - Formateado de la clase String.

ACTIVIDADES DE ENSEÑANZA-APRENDIZAJE

- Definir clases.
- Definir propiedades y métodos. Crear objetos.
- Crear constructores(sobrecarga).
- Desarrollar programas que instancien y utilicen objetos.
- Utilizar mecanismos para controlar la visibilidad de las clases y de sus miembros.
- Incorporar y utilizar librerías de objetos.
- Escribir llamadas a métodos tanto de instancia como de clase.
- Utilizar parámetros en la llamada a métodos.
- Instanciar objetos a partir de clases predefinidas.
- Reconocer las posibilidades de entrada/salida del lenguaje y las librerías asociadas.
- Escribir métodos, donde se pasa un objeto por parámetro.

U.T.04. Nociones avanzadas de programación orientada a objetos

Se relaciona con los resultados de aprendizaje: R.A.-2, R.A.-3 y R.A.-4

CONTENIDOS

- Herencia
 - Fundamentos de la herencia
 - Acceso a los miembros heredados en una subclase
 - Operador super (constructores y sobreescritura de métodos)
 - Creación de una jerarquía multinivel
 - Final con la herencia
 - Clase Object
 - Tipo enumerado
- Polimorfismo
- Mecanismo casting
- Operador instanceof
- Métodos y clases abstractas
 - Métodos abstractos. Características
 - Clases abstractas. Cuando utilizarlas
 - Características de una clase abstracta
- Interfaces
 - Usos de las interfaces
 - Características de las interfaces
 - Implementación de las interfaces
 - Herencia entre interfaces
 - Interfaces y polimorfismo
 - Métodos default en interfaces

ACTIVIDADES DE ENSEÑANZA-APRENDIZAJE

- Definir y utilizar clases heredadas.
- Identificar los conceptos de herencia, superclase, subclase y sus características.

- Utilizar modificadores para bloquear y forzar la herencia de clases y métodos.
- Utilizar los diferentes modos de acceso a los miembros de una clase.
- Reconocer la incidencia de los constructores en la herencia y saber definirlos.
- Crear clases heredadas que sobrescriban la implementación de métodos de la superclase.
- Diseñar y aplicar jerarquías de clases.
- Definir y utilizar el tipo enumerado.
- Probar y depurar las jerarquías de clases.
- Realizar programas que implementen y utilicen jerarquías de clases.
- Comentar y documentar el código.
- Utilizar el operador super.
- Realizar programas usando polimorfismo y comprobar su utilidad.
- Definir y utilizar el mecanismo casting y comprobar su utilidad.
- Definir y utilizar el tipo enumerado.
- Utilizar los diferentes modos de acceso a los miembros de una clase.
- Definir y utilizar el operador instanceof.
- Definir, implementar y utilizar clases y métodos abstractos.
- Definir, implementar y utilizar interfaces en herencia múltiple.
- Implementar, codificar, compilar, depurar y ejecutar programas donde se utilice herencia, polimorfismo, clases abstractas e interfaces.

U.T.05. Estructuras de almacenamiento

Se relaciona con los resultados de aprendizaje: R.A.-3 y R.A.-4.

CONTENIDOS

- Arrays: propiedades y tipos.
- Arrays unidimensionales o vectores: declaración, creación, inicialización, acceso a elementos, recorrido y operaciones (ordenación y búsquedas).
- Cadenas de caracteres. Características y métodos.
- Arrays de objetos: declaración, creación, inicialización, acceso a los elementos, recorrido y ordenación.
- Arrays bidimensionales o matrices: declaración, creación, inicialización, acceso a los elementos y recorrido.
- Colecciones: declaración, características, recorrido (iterador), wrappers o métodos genéricos y tipos.
 - Interfaz Set (colecciones sin duplicados). Clases y métodos.
 - Interfaz List (colecciones con duplicados). Clases y métodos.
 - Interfaz Queue. Clases y métodos.
 - Interfaz Map. Clases y métodos.

ACTIVIDADES DE ENSEÑANZA-APRENDIZAJE

- Reconocer las librerías de clases relacionadas con tipos de datos avanzados.
- Crear programas que utilicen vectores. Saber utilizar la propiedad length
- Crear programas que nos permitan ordenar arrays utilizando diferentes métodos (sort, baraja, selección directa y burbuja).
- Crear programas que nos permitan realizar búsquedas en arrays desordenados (secuencial) y ordenados (secuencial y binaria).
- Crear programas que nos permitan definir arrays de objetos y utilizarlos accediendo a sus elementos y recorriéndolo.
- Crear programas que nos permitan ordenar arrays de objetos de forma predeterminada y usando métodos Compare y CompareTo.
- Crear programas que nos permitan declarar y utilizar arrays bidimensionales o matrices.
- Reconocer las características y ventajas de cada una de las colecciones de datos disponibles.
- Crear programas que nos permitan crear distintos tipos de colecciones y utilizarlas utilizando métodos y el iterador para recorrerlos.

U.T.06. Clases genéricas y control de excepciones

Se relaciona con los resultados de aprendizaje: R.A.-3.

CONTENIDOS

- Control de excepción.
- Tipos de excepciones.
- Clase Error.
- Clase Exception.
- Manejo de excepciones
 - try...catch.
 - Bloque finally.
 - Propagación de excepciones.
 - Lanzar excepciones.
- Excepciones definidas por el programador.
- Malas prácticas en el uso de excepciones.
- Recomendaciones de uso.

ACTIVIDADES DE ENSEÑANZA-APRENDIZAJE

- Utilizar los métodos de la clase Exception para el control de errores.
- Crear programas utilizando el bloque finally de forma adecuada.
- Crear programas lanzando excepciones a nivel superior.
- Resolver problemas definiendo excepciones propias.
- Resolver problemas capturando las excepciones en los métodos con try-catch.
- Resolver problemas propagando excepciones.

U.T.07. Lectura y escritura de información

Se relaciona con los resultados de aprendizaje: R.A.-5.

CONTENIDOS

- Flujos de datos en java y sus tipos.
- Clases de java relativas a flujos. Jerarquía de clases.
- Entrada/salida estándar.
- Almacenamiento de información en ficheros.
- Ficheros secuenciales
 - Ficheros de texto. Clases para manejarlos y sus métodos.
 - Ficheros binarios. Clases para manejarlos y sus métodos.
 - Serialización. Clases para manejarlos y sus métodos.
- Clase File. Métodos más representativos.

ACTIVIDADES DE ENSEÑANZA-APRENDIZAJE

- Reconocer las posibilidades de entrada/salida del lenguaje y las librerías asociadas.
- Utilizar ficheros de diferentes tipos para almacenar, manejar y recuperar información.
- Crear programas que utilicen diversos métodos de acceso al contenido de los ficheros.
- Crear programas que nos permitan almacenar, manejar y recuperar la información en ficheros utilizando buffers.
- Crear programas que nos permitan añadir datos en un fichero ya creado.
- Utilizar la clase File para crear, borrar y manipular ficheros y directorios.

U.T.08 Interfaces gráficas

Se relaciona con los resultados de aprendizaje: RA1, RA2, RA3, RA5

CONTENIDOS

- Interfaces gráficas
 - Concepto
 - Librerías
 - Tipos
 - Eventos
- JavaFX
 - Estructura
 - Contenedores
 - Controles
 - Diálogos
 - Gestión de eventos

ACTIVIDADES DE ENSEÑANZA-APRENDIZAJE

- Ejercicios guiados para desarrollar interfaces gráficas de usuario de complejidad creciente.
- Creación de interfaces de usuario a partir de prototipos.
- Realización de tareas de codificación que integren los contenidos de esta unidad con la anterior.
- Revisión de ejercicios de otras unidades para añadir interfaz gráfico reemplazando el interfaz en modo texto.
- Profundización en las posibilidades de la librería JavaFX para crear interfaces gráficas para dispositivos móviles.
- Investigación de la relación existente entre JavaFX y CSS.

U.T.09 Control de acceso y mantenimiento de bases de datos relacionales

Se relaciona con los resultados de aprendizaje: R.A.-9

CONTENIDOS

- Herramientas para el establecimiento de conexiones con bases de datos relacionales.
- Sistemas de recuperación de la información almacenada en la base de datos.
- Herramientas para manipular y actualizar la información.
- Sintaxis y características del lenguaje de consultas sobre la base de datos.

ACTIVIDADES DE ENSEÑANZA-APRENDIZAJE

- Identificar las características y métodos de acceso a sistemas gestores de bases de datos relacionales.
- Programar conexiones con bases de datos.
- Escribir código para almacenar información en bases de datos.
- Crear programas para recuperar y mostrar información almacenada en bases de datos.
- Efectuar borrados y modificaciones sobre la información almacenada.
- Crear aplicaciones que ejecuten consultas sobre bases de datos.
- Crear aplicaciones para posibilitar la gestión de información presente en bases de datos relacionales.

3.2 Contenidos Mínimos

Tras el análisis de la materia, se detectan al menos tres bloques de contenidos mínimos

1) Algorítmica(U.T.01 y U.T.02): la lógica de programación, requisito para programar en cualquier lenguaje con características imperativas.

- Conocer y saber utilizar la sintaxis de las sentencias de Java.
- Saber utilizar correctamente las diferentes estructuras de control (secuenciales, alternativas, repetitivas, salto), así como la E/S de datos.
- Saber utilizar correctamente contadores y acumuladores.
- Diseñar programas de forma modularizada.
- Conocer y utilizar correctamente los distintos tipos de métodos existentes en java (los que devuelven un solo valor y los que no devuelven ningún valor).
- Saber utilizar correctamente el paso argumento-parámetro y solo cuando sea necesario.
- Realizar correctamente las llamadas a los métodos.
- Conocer la estructura de un programa java.
- Conocer y utilizar las reglas para componer identificadores.
- Conocer y distinguir los diferentes tipos de datos (primitivos y complejos).
- Conocer las secuencias de escape.
- Distinguir entre conversión implícita y explícita (casting) de datos.
- Saber que es una variable, cuando usarla y como declararla.

- Conocer y saber utilizar los distintos tipos de variables en java.
- Saber que es el ámbito de una variable y conocer los distintos tipos existentes.
- Saber referenciar una variable en un ámbito concreto.
- Saber que es una constante, diferenciarla de una variable.
- Conocer cuando utilizar constantes y saber definirla.
- Conocer y saber utilizar los diferentes operadores en expresiones (aritméticos, relacionales y condicionales).
- Conocer los tipos de comentarios existentes.
- Conocer que es una palabra reservada y cuando no se pueden utilizar.
- Saber importar paquetes y clases de la API de Java.

2) Estructura de programa, clases y herencia (U.T.03 , U.T.04 y U.T.06): conceptos asociados a la programación moderna, con buena estructura del programa y las posibilidades de la programación orientada a objetos.

- Definir y saber utilizar correctamente los métodos de instancia y de clase.
- Conocer y saber utilizar correctamente los atributos de instancia y de clase.
- Conocer y saber utilizar la sintaxis y funcionamiento de los distintos tipos de métodos: declaración, definición, llamadas y parámetros, así como las librerías de Java.
- Saber utilizar de forma adecuada los modificadores de acceso y de contenido.
- Saber invocar de forma correcta a métodos y atributos de instancia y de clase.
- Saber definir y utilizar los métodos getter, setter, de propósito general y toString.
- Saber pasar y utilizar correctamente un objeto por parámetro.
- Definir correctamente las clases (atributos y métodos) e instanciar objetos, teniendo en cuenta las características de la POO.
- Saber manipular correctamente un objeto, mediante el uso de atributos y métodos.
- Conocer la diferencia entre variables locales y globales (ámbito de las variables).
- Identificar correctamente los errores cometidos al realizar un programa.
- Definir y utilizar correctamente la referencia this.
- Definir y utilizar correctamente los constructores.
- Conocer y utilizar correctamente la sobrecarga de métodos.
- Utilizar correctamente los métodos de la clase String.
- Realizar el control de excepciones (propagación y captura).
- Saber realizar un diseño jerárquico de clases.
- Saber realizar la declaración de las subclases utilizando modificadores.
- Conocer los diferentes modos de acceso a los miembros de las clases y a los heredados en una subclase y saber utilizarlos.
- Conocer y utilizar adecuadamente el operador super en la jerarquía de clases.
- Conocer y utilizar correctamente la redefinición de métodos.
- Conocer y saber utilizar el modificador final con la herencia.
- Definir y conocer la clase Object.
- Saber definir y utilizar el tipo enumerado.
- Saber qué es el polimorfismo y utilizarlo correctamente.
- Conocer y saber utilizar correctamente el mecanismo casting en los programas para poder transformar un objeto de un tipo a otro.
- Conocer y saber utilizar correctamente el operador instanceof
- Saber definir y utilizar correctamente métodos y clases abstractas.
- Saber definir y utilizar correctamente interfaces en la herencia múltiple.
- Saber definir y utilizar correctamente los métodos default en interfaces.

- Conocer y utilizar correctamente el polimorfismo en interfaces.
- Conocer la superclase de todas las excepciones.
- Conocer los diferentes tipos de excepciones y cuales son de obligada gestión.
- Conocer la clase exception y sus subclases.
- Conocer y saber utilizar los métodos de la clase exception.
- Saber manejar o tratar los errores en java.
- Saber cuando se deben tratar o gestionar obligatoriamente las excepciones y cuando no.
- Saber utilizar correctamente el bloque try-catch para capturar y gestionar excepciones en los propios métodos.
- Conocer y saber utilizar correctamente el bloque finally.
- Conocer y saber propagar excepciones y el tipo de excepciones que deben propagarse y cuáles no.
- Saber lanzar excepciones y distinguir cuando se pueden lanzar.
- Saber definir una excepción propia.

3) Recursos de programación y datos(U.T.05, U.T.07, U.T.08 y U.T.09): diferentes recursos para los programas: estructuras de almacenamiento en RAM, estructuras de almacenamiento externo (ficheros) y acceso a bases de datos.

- Conocer y distinguir las diferentes estructuras de datos.
- Saber declarar, crear e inicializar correctamente los diferentes tipos de arrays (unidimensionales, bidimensionales y de objetos).
- Conocer y acceder correctamente a los elementos de un array.
- Conocer y utilizar correctamente la propiedad length
- Conocer y recorrer correctamente un array mediante el bucle for y foreach.
- Saber manejar correctamente los elementos de los arrays.
- Conocer y saber utilizar los arrays en los diferentes tipos de métodos existentes.
- Conocer y saber manejar correctamente los diferentes métodos de ordenación de arrays para que estos queden ordenados.
- Conocer y saber utilizar en los programas de forma correcta los métodos Compare y CompareTo.
- Conocer y saber buscar de forma adecuada elementos en un array tanto si estos están desordenados como ordenados, utilizando de forma adecuada la búsqueda que mejor se adapte a cada circunstancia.
- Conocer y saber utilizar correctamente los Wrappers.
- Conocer los diferentes tipos de colecciones y saber confeccionarlos o crearlos de forma adecuada.
- Conocer las características propias de los diferentes tipos de colecciones existentes en java y saber utilizar las más adecuadas en función del problema a resolver.
- Conocer y saber implementar los diferentes tipos de colecciones existentes en java.
- Conocer y saber manejar de forma adecuada los métodos propios de cada colección.
- Conocer y saber declarar una colección.
- Conocer y utilizar de forma adecuada el objeto iterador a través de sus métodos.
- Saber almacenar, modificar, borrar y buscar en estructuras de almacenamiento interno (arrays, colecciones).
- Conocer la necesidad de los Streams, sus tipos y las operaciones que se pueden llevar a cabo con ellos.
- Saber crear, abrir, escribir, leer y cerrar un Stream.
- Conocer las características propias de los diferentes tipos de ficheros.

- Conocer y saber utilizar de forma adecuada según los tipos de almacenamiento externo, las clases relativas a los flujos de datos, sus constructores y métodos.
- Saber almacenar, modificar, borrar y buscar información en estructuras de almacenamiento externo.
- Saber manejar adecuadamente la información contenida en los ficheros para obtener los resultados esperados
- Conocer y saber utilizar los buffers en el manejo de la información desde o hacia el almacenamiento externo.
- Programar conexiones con bases de datos.
- Escribir código para almacenar información en bases de datos.
- Crear programas para recuperar y mostrar información almacenada en bases de datos.
- Efectuar borrados y modificaciones sobre la información almacenada.
- Crear aplicaciones que ejecuten consultas sobre bases de datos.
- Desarrollar interfaces gráficas sencillas para comunicar aplicaciones y usuarios
- Conocer los componentes de uso más común para la creación de interfaces gráficas
- Estudiar el concepto de evento y su utilización en el flujo de ejecución de las interfaces de usuario.
- Crear interfaces de usuario básicas mediante javaFX

3.3 TEMPORALIZACIÓN

El desarrollo de unidades temáticas por evaluaciones será el siguiente:

PRIMERA EVALUACIÓN:

- U.T.01. Identificación de un programa informático (12 horas)
- U.T.02. Aplicación de las estructuras de control (41 horas)
- U.T.03. Programación orientada a objetos en Java (36 horas)

SEGUNDA EVALUACIÓN:

- U.T.03. Programación orientada a objetos en Java (10 horas)
- U.T.04. Nociones avanzadas de Programación Orientada a Objetos (35 horas).
- U.T. 05. Estructuras de Almacenamiento (42 horas)

TERCERA EVALUACIÓN:

- U.T. 05. Estructuras de Almacenamiento (18 horas)
- U.T. 06. Clases genéricas y control de excepciones (3 horas)
- U.T. 07. Lectura/Escritura de la información (19 horas)
- U.T. 08. Interfaces gráficas (9 horas)
- U.T. 09. Control de acceso y mantenimiento de BD relacionales (9 horas)

4. Metodología

1. Exposición del concepto a tratar utilizando el mayor número posible de gráficos y esquemas.
2. El proceso de enseñanza-aprendizaje se ha programado, fundamentalmente, basándose en la realización de una serie de actividades que pretenden propiciar la iniciativa del alumnado y el proceso de autoaprendizaje, desarrollando capacidades de comprensión, análisis, relación, búsqueda y manejo de la información y que intentan, además, conectar el aula con el mundo real: empresas, profesionales y organismos administrativos que conforman el entorno profesional y de trabajo del/la técnico/a que se quiere formar.
3. Realización de actividades prácticas relacionadas con los conceptos estudiados. Las prácticas a realizar serán diversas, de aplicación directa en los equipos informáticos, utilizando diferentes

herramientas software que el alumnado tiene a su disposición para el desarrollo, depuración y prueba de programas.

4. Las unidades didácticas se expondrán en dos fases:
 - a. Parte teórica: se compondrá de una exposición de la unidad, explicando los contenidos desarrollados en cada unidad, posibilitando en la medida de lo posible el autoaprendizaje, incluyendo en parte de la exposición ciertos interrogantes que el alumnado deberá de resolver por sí mismo.
 - b. Parte práctica: realizando supuestos prácticos, que sirvan para afinar los conocimientos teóricos, siendo éstos lo más reales posible, al objeto de mantener una cierta motivación en el aprendizaje de la materia.

En caso de enfermedad el alumnado podrá consultar información del módulo a través del Microsoft Teams. En esta plataforma se irán colgando apuntes, actividades para realizar, etc.

5. Materiales curriculares

- El profesor proporcionará contenidos multimedia, presentaciones y guiones para las prácticas.
- Como libro de referencia para los contenidos teóricos se utilizará el siguiente libro de texto:

Título: "Programación"

Editorial: Síntesis

Autor: Mario Dorrego Martín

ISBN: 978-84-9171-323-4

Material Hardware:

- Un aula de informática con 30 ordenadores personales (para el alumnado) gama media/alta y con posibilidad de funcionar de forma autónoma o en red.
- Un ordenador para el profesor, igualmente con posibilidad de funcionar de forma autónoma o en red.
- Proyector (conectado al ordenador del profesorado).
- Conexión a Internet.

Material Software:

- Sistemas Operativos (Windows y Linux) Servidores y Clientes.
- Software de Virtualización: Oracle VirtualBox.
- Programas auxiliares (lector PDF, compresor, antivirus, etc.).
- Entornos de Desarrollo (se podrán utilizar entornos como Eclipse, Netbeans, Visual Studio, o similares en función de las necesidades).

Material Bibliográfico de Consulta:

- Libro de texto “Programación”. Editorial: Síntesis (Mario Dorrego Martín).
- Documentación específica aportada por el profesor.
- Como recurso principal para la búsqueda de información se usará Internet.

6. Criterios de evaluación

Con la finalidad de evaluar la adquisición de las competencias profesionales, personales y sociales, a alcanzar con este módulo, es necesario comprobar si el alumnado ha adquirido los aprendizajes correspondientes. Por ello, en este apartado estableceremos los criterios de evaluación que midan la consecución de los resultados de aprendizaje ligados a los objetivos definidos para el módulo.

Todos los criterios de evaluación establecidos se consideran mínimos.

RA1.- Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.

Criterios de evaluación:

- a) Se han identificado los bloques que componen la estructura de un programa informático.
- b) Se han creado proyectos de desarrollo de aplicaciones
- c) Se han utilizado entornos integrados de desarrollo.
- d) Se han identificado los distintos tipos de variables y la utilidad específica de cada uno.
- e) Se ha modificado el código de un programa para crear y utilizar variables.
- f) Se han creado y utilizado constantes y literales.
- g) Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje.
- h) Se ha comprobado el funcionamiento de las conversiones de tipo explícitas e implícitas.
- i) Se han introducido comentarios en el código.

RA2.- Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.

Criterios de evaluación:

- a) Se han identificado los fundamentos de la programación orientada a objetos.
- b) Se han escrito programas simples.
- c) Se han instanciado objetos a partir de clases predefinidas.
- d) Se han utilizado métodos y propiedades de los objetos.
- e) Se han escrito llamadas a métodos estáticos.
- f) Se han utilizado parámetros en la llamada a métodos.
- g) Se han incorporado y utilizado librerías de objetos.
- h) Se han utilizado constructores.
- i) Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples.

RA3.- Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.

Criterios de evaluación:

- a) Se ha escrito y probado código que haga uso de estructuras de selección.
- b) Se han utilizado estructuras de repetición.
- c) Se han reconocido las posibilidades de las sentencias de salto.
- d) Se ha escrito código utilizando control de excepciones.
- e) Se han creado programas ejecutables utilizando diferentes estructuras de control.
- f) Se han probado y depurado los programas.
- g) Se ha comentado y documentado el código.

RA4.- Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.

Criterios de evaluación:

- a) Se ha reconocido la sintaxis, estructura y componentes típicos de una clase.
- b) Se han definido clases.
- c) Se han definido propiedades y métodos.
- d) Se han utilizado constructores y destructores.
- e) Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.
- f) Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros.
- g) Se han definido y utilizado clases heredadas.
- h) Se han creado y utilizado métodos estáticos.
- i) Se han definido y utilizado interfaces.
- j) Se han creado y utilizado conjuntos y librerías de clases.

RA5.- Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases.

Criterios de evaluación:

- a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información.
- b) Se han aplicado formatos en la visualización de la información.
- c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.
- d) Se han utilizado ficheros para almacenar y recuperar información.
- e) Se han creado programas que utilicen diversos métodos de acceso al contenido de los ficheros.
- f) Se han implementado técnicas para envío y almacenamiento de objetos en ficheros

- g) Se han utilizado las herramientas del entorno de desarrollo para crear interfaces gráficas de usuario simples.
- h) Se han programado controladores de eventos.
- i) Se han escrito programas que utilicen interfaces gráficas para la entrada y salida de información.

RA6.- Escribe programas que manipulen información, seleccionando y utilizando tipos avanzados de datos.

Criterios de evaluación:

- a) Se han escrito programas que utilicen arrays.
- b) Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados.
- c) Se han utilizado listas para almacenar y procesar información.
- d) Se han utilizado iteradores para recorrer los elementos de las listas.
- e) Se han reconocido las características y ventajas de cada una de las colecciones de datos disponibles.
- f) Se han creado clases y métodos genéricos.
- g) Se han utilizado expresiones regulares en la búsqueda de patrones en cadenas de texto.
- h) Se han identificado las clases relacionadas con el tratamiento de documentos XML.
- i) Se han realizado programas que realicen manipulaciones sobre documentos XML.

RA7.- Desarrolla programas, aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.

Criterios de evaluación:

- a) Se han identificado los conceptos de herencia, superclase y subclase.
- b) Se han utilizado modificadores para bloquear y forzar la herencia de clases y métodos.
- c) Se ha reconocido la incidencia de los constructores en la herencia.
- d) Se han creado clases heredadas que sobrescriban la implementación de métodos de la superclase.
- e) Se han diseñado y aplicado jerarquías de clases.
- f) Se han probado y depurado las jerarquías de clases.
- g) Se han realizado programas que implementen y utilicen jerarquías de clases.
- h) Se ha comentado y documentado el código.

RA8.- Utiliza bases de datos orientadas a objetos, analizando sus características y aplicando técnicas para mantener la persistencia de la información.

Criterios de evaluación:

- a) Se han identificado las características de las bases de datos orientadas a objetos.
- b) Se ha analizado su aplicación en el desarrollo de aplicaciones mediante lenguajes orientados a objetos.
- c) Se han instalado sistemas gestores de bases de datos orientados a objetos.

- d) Se han clasificado y analizado los distintos métodos soportados por los sistemas gestores para la gestión de la información almacenada.
- e) Se han creado bases de datos y las estructuras necesarias para el almacenamiento de objetos.
- f) Se han programado aplicaciones que almacenen objetos en las bases de datos creadas.
- g) Se han realizado programas para recuperar, actualizar y eliminar objetos de las bases de datos.
- h) Se han realizado programas para almacenar y gestionar tipos de datos estructurados, compuestos y relacionados.

RA9.- Gestiona información almacenada en bases de datos relacionales manteniendo la integridad y la consistencia de los datos.

Criterios de evaluación:

- a) Se han identificado las características y métodos de acceso a sistemas gestores de bases de datos relacionales.
- b) Se han programado conexiones con bases de datos.
- c) Se ha escrito código para almacenar información en bases de datos.
- d) Se han utilizado diferentes asistentes del entorno de desarrollo.
- e) Se han creado programas para recuperar y mostrar información almacenada en bases de datos.
- f) Se han efectuado borrados y modificaciones sobre la información almacenada.
- g) Se han creado aplicaciones que ejecuten consultas sobre bases de datos.
- h) Se han creado aplicaciones para posibilitar la gestión de información presente en bases de datos relacionales.

7. Procedimientos de evaluación

La evaluación del módulo considerará los siguientes cuatro momentos a lo largo del curso académico:

- **Evaluación inicial o de diagnóstico:** Que permitirá evaluar habilidades de los alumnos. No tendrá repercusión en la calificación final del módulo. La evaluación inicial o de diagnóstico se realiza, de manera prescriptiva, durante los primeros días del curso con el fin de detectar el alumnado con problemas de aprendizaje.
- **Evaluación continua y formativa:** Se realiza a lo largo de todo el curso académico. Se tratará de llevar un seguimiento, lo más intenso posible, del proceso de aprendizaje seguido por cada alumno. De esta manera será factible proponer, en el momento más adecuado, las actividades de refuerzo necesarias en cada caso para poder resolver los problemas detectados en el aprendizaje individual. **El alumnado perderá el derecho de evaluación continua** (en cada una de las evaluaciones) **si alcanzase el 15% de faltas de asistencia respecto a la totalidad de horas del módulo** (en la correspondiente evaluación).

- **Evaluación Final Ordinaria.** Se realiza al final del periodo lectivo. Esta evaluación será tanto para el alumnado que no supere el módulo, tras ser evaluado de la forma anteriormente descrita y de acuerdo con los criterios que posteriormente se citan, como para aquel que haya perdido el derecho a la evaluación continua.
- **Evaluación Final Extraordinaria.** Se realiza al final del curso académico. Esta evaluación será para el alumnado que no haya superado la Evaluación Final Ordinaria.

7.1. Criterios para la elaboración de la prueba de la convocatoria extraordinaria

La prueba extraordinaria tendrá contenidos teóricos y prácticos. Consistirá en resolver diferentes supuestos de dificultad semejante a los ejercicios realizados durante el curso.

8. Instrumentos de evaluación

La evaluación se realizará agrupando las unidades temáticas por evaluaciones trimestrales. El proceso de evaluación debe atender a los siguientes puntos:

- En el desarrollo de las U.T. en que se divide el módulo, se realiza un proceso de evaluación continua.
- A lo largo del curso los alumnos almacenarán en un medio digital sus apuntes y los trabajos, actividades y ejercicios que se vayan realizando.
- La evaluación continua se realiza valorando los siguientes instrumentos:

A. REALIZACIÓN DE PRUEBAS ESPECÍFICAS.

Consiste en la realización de exámenes prácticos sobre ordenador relativos a los contenidos impartidos en el aula.

B. ANALISIS DE PRODUCCIONES DE LOS ALUMNOS.

Tareas o ejercicios prácticos individuales realizados en el ordenador bien en clase o de manera telemática.

C. OBSERVACION SISTEMATICA.

Se evaluará la actitud que manifiesta el alumno a lo largo del curso, basándose en los indicadores siguientes:

- Participación e interés.
- Trabajo en clase y /o en casa.
- Iniciativa y autonomía en el trabajo.
- Uso adecuado de medios y recursos.

9. Criterios de calificación

9.1 EVALUACIÓN TRIMESTRAL

En cada evaluación trimestral, la calificación del alumno se obtiene de:

1. PRUEBAS ESPECÍFICAS (70%)

Calificación obtenida por el alumno, como resultado de las notas obtenidas en los exámenes de evaluación acordados para el trimestre.

Cada examen de evaluación realizado tendrá un carácter práctico y versará sobre los contenidos impartidos a lo largo del trimestre y de trimestres anteriores si los hubiera. Se valorará de 0 a 10 puntos de acuerdo con las especificaciones establecidas en el mismo. Un examen se considera superado, si el alumno obtiene una calificación igual o superior a 5 sobre 10.

En el supuesto de que solamente se realizase un examen en el trimestre, la nota de este apartado será dicha nota. Si a lo largo del periodo de evaluación se hubieran realizado varios exámenes, la nota de este apartado será la media de las calificaciones obtenidas en dichas pruebas.

2. PRODUCCIONES DE LOS ALUMNOS (20%)

En este apartado se incluyen las tareas o ejercicios prácticos individuales realizados en clase o de forma telemática, que el profesor considere oportuno evaluar durante el trimestre. En caso de que no se entregue alguna tarea o ejercicio, o bien se entregue fuera del plazo establecido por el profesor, se evaluará con una nota de cero. En caso contrario, se puntuará de 0 a 10 puntos de acuerdo con las especificaciones establecidas en cada tarea o ejercicio.

El profesor si lo considera oportuno puede pedir al alumno la defensa de la tarea. Consistirá en realizarle preguntas sobre la resolución de esta. Por cada pregunta que el alumno no conteste o conteste de forma incorrecta, se descontará la puntuación correspondiente según especificaciones del ejercicio.

La nota de este apartado será la media de las calificaciones obtenidas en las tareas o ejercicios prácticos, en el caso de que se hayan tenido que entregar varias tareas. Si solo hubo que entregar una tarea en el trimestre, la nota de dicha tarea será la que figure en este apartado.

3. OBSERVACIÓN SISTEMÁTICA (10%)

Valoración de la actitud observada en el alumno. Esta nota la proporciona la puntuación conseguida en cada uno de los indicadores siguientes, valorados como:

- Negativa (0 puntos),
- Aceptable (1.5 puntos) y,
- muy positiva (2.5 puntos).

Este apartado tendrá un peso del 10% de la nota trimestral.

- **Participación e interés.** El alumno manifiesta interés por la materia, presta atención a las explicaciones y participa en las clases.
- **Trabajo en clase o casa.** Se trata de valorar el esfuerzo que el alumno realiza en el aula o en casa, para intentar resolver los ejercicios planteados por el profesor de forma adecuada.
- **Iniciativa y autonomía en el trabajo.** El alumno tiene espíritu emprendedor. Manifiesta interés por ampliar sus conocimientos y buscar soluciones para los problemas sin ayuda del profesor.
- **Uso adecuado de medios y recursos:** Utiliza los recursos materiales del centro adecuadamente y con responsabilidad. No utiliza los equipos informáticos nada más que para utilizar el software permitido, y tiene su equipo y materiales asignados ordenados, configurados, y es cuidadoso con ellos.

NOTA DEL TRIMESTRE:

Nota trimestral = suma de:

- 70% de la nota obtenida en el apartado pruebas específicas
- 20% de la nota obtenida en el apartado producciones alumnos
- 10% de la nota obtenida en el apartado observación sistemática,

El alumno aprobará el trimestre, si obtiene una calificación total mínima de 5.

9.2 EVALUACIÓN FINAL

La calificación final del alumno será la nota media de las calificaciones obtenidas en las evaluaciones trimestrales siempre y cuando el tercer trimestre tenga una calificación de al menos un 5.

En el caso de que la nota del tercer trimestre es al menos 5 y la nota media de las evaluaciones trimestrales sea menor de 5 la nota del alumno será 5.

El alumno aprobará el módulo, si obtiene una calificación total mínima de 5.

9.3 RECUPERACIÓN

Recuperación ordinaria

Todos los alumnos que hayan obtenido una calificación menor a 5 en algún trimestre lo recuperarán si aprueban o recuperan el trimestre siguiente, puesto que los contenidos de un trimestre se siguen aplicando en el resto de forma habitual. Si se tiene suspenso el tercer trimestre, independientemente de la nota obtenida en los anteriores, el alumno tendrá que realizar un examen final.

9.4 EXAMEN FINAL

A este tipo de prueba se someterán los alumnos que tengan el tercer trimestre suspenso. El examen final podrá realizarse en varios días. Consistirá en la realización de una prueba práctica basada en los contenidos impartidos durante el curso y reflejados en la programación. La prueba será calificada de 0 a 10 puntos de acuerdo con la puntuación especificada en la misma.

La nota obtenida en la prueba sustituirá a la del tercer trimestre para aplicarla a la media en la evaluación final.

9.5 Evaluación extraordinaria

El alumno que obtenga una evaluación negativa del módulo en la evaluación final ordinaria deberá realizar una prueba objetiva que se celebrará en el mes de junio.

Consistirá en la realización de un examen práctico basado en los contenidos impartidos durante el curso y reflejados en la programación. No se exigirá la presentación de actividades para evaluarlas, aunque el alumno tendrá acceso a todas las realizadas durante el curso. La prueba será calificada de 0 a 10 puntos de acuerdo con la puntuación especificada en la misma, siendo necesaria una calificación de al menos un 5 para considerar aprobado el módulo. La nota obtenida en la prueba será la que figure en la evaluación extraordinaria de junio.

9.6 Evaluación de pendientes

En el caso de aquellos alumnos que hayan promocionado a 2º y tengan este módulo suspenso, realizarán una prueba de recuperación cuya fecha será comunicada con la suficiente antelación al alumno.

La prueba consistirá en la realización de un examen práctico basado en los contenidos impartidos durante el curso anterior. No se exigirá la presentación de actividades para evaluarlas, aunque se le proporcionarán al alumno junto con el plan de recuperación para que pueda preparar el módulo de forma adecuada.

La prueba será calificada de 0 a 10 puntos de acuerdo con la puntuación especificada en la misma, siendo necesario al menos un 5 para considerar el módulo aprobado.

La nota obtenida en la prueba será la que figure en la evaluación correspondiente.

9.7 Programación de recuperación para alumnos con imposibilidad de aplicación del proceso de evaluación continua

Los alumnos cuyo número de faltas de asistencia en un trimestre sea del 15% o más de pérdida lectiva para ese trimestre lo recuperarán si aprueban o recuperan el trimestre siguiente, puesto que los contenidos de un trimestre se siguen aplicando en el resto de forma habitual. Si se tiene suspenso el tercer trimestre, independientemente de la nota obtenida en los anteriores, el alumno tendrá que realizar un examen final.

10. Medidas de atención a la diversidad

10.1. Medidas de refuerzo

Se pretende que el alumnado alcance el máximo aprovechamiento de las materias que estudian y evitar en lo posible el fracaso académico en este módulo. Para poder detectar, con suficiente antelación, anomalías que se presenten en el proceso de enseñanza-aprendizaje se proponen las siguientes medidas:

- Presentar los contenidos diferenciando claramente los elementos que resultan básicos y esenciales de aquellos que los amplían y profundizan.
- Desarrollar actividades y trabajos siempre afines a los contenidos realizados.
- Uso de ejemplos cercanos a la realidad a fin de facilitar su asimilación.
- Preparar actividades con diferentes niveles de dificultad de forma que el alumnado pueda encontrar espacios de respuesta conforme a sus capacidades.
- Proponer actividades de refuerzo en aquellos contenidos que no se hayan asimilado correctamente.

10.2. Seguimiento y evaluación de las medidas adoptadas para la atención a la diversidad

El seguimiento se realizará en las reuniones periódicas del equipo docente y a través de las labores de tutoría del alumnado implicado.

11. Actividades complementarias y extraescolares

No se programa ninguna actividad específica para este módulo, aunque si durante el curso surgen temas de interés para esta materia como charlas, conferencias, presentaciones de productos... se procurará la asistencia de los alumnos/as a estas actividades.

12. Contribución del módulo a la educación en valores y a la igualdad de derechos y oportunidades entre las personas

Las características de los contenidos y los aprendizajes correspondientes a este ciclo formativo desarrollan la educación en valores que nuestro sistema educativo promueve. Se trabajará, realizando prácticas en grupo, los aspectos relacionados con las diferencias entre las personas, las distintas formas diferentes de expresión, el respeto a las realizaciones prácticas de las compañeras y se promoverá una educación no sexista enmarcada en el ámbito legislativo.

13. Indicadores de logro y procedimiento de evaluación de la aplicación y desarrollo de la programación docente

El procedimiento de evaluación de la programación será el que el propio centro determine en la Programación General Anual.

Se incluirán en un cuestionario específico los siguientes indicadores de logro:

- Resultados de la evaluación del curso en cada una de las materias, por curso y grupo.
- Adecuación de los materiales, recursos didácticos, y distribución, en su caso, de espacios y tiempos a la secuenciación de contenidos y criterios de evaluación asociados.
- Contribución de los métodos pedagógicos y medidas de atención a la diversidad aplicadas a la mejora de los resultados obtenidos.
- Valoración de actividades complementarias organizadas por el Departamento o con participación del mismo.