

PROGRAMACIÓN DIDÁCTICA

MÓDULO PROGRAMACIÓN

IES AGUADULCE
CICLO FORMATIVO DE GRADO SUPERIOR
DESARROLLO DE APLICACIONES MULTIPLATAFORMA (DAM)
CURSO 2023/2024
PROFESOR: DIOSDADO SÁNCHEZ HERNÁNDEZ

ÍNDICE

1. Introducción.....	3
2. Contextualización de la Programación.....	4
2.1. CFGS de Desarrollo de Aplicaciones Multiplataforma en el contexto del IES Aguadulce.....	4
2.2. Características del Grupo.....	4
3. Competencias y Objetivos Generales del Título.....	5
4. Unidades de Competencia y Resultados de Aprendizaje.....	8
5. Unidades de Trabajo.....	9
5.1. Contenidos y Secuencia de las Unidades de Trabajo.....	9
5.2. Relación de Resultados de Aprendizaje con las Unidades de Trabajo Propuestas.....	11
5.3. Temporalización.....	12
6. Metodología.....	13
6.1. Utilización del aula virtual como apoyo a la docencia.....	13
6.2. Estrategias metodológicas en el aula.....	14
7. Evaluación.....	15
7.1. Criterios de Evaluación.....	16
7.2. Instrumentos de Evaluación.....	19
7.3. Criterios de calificación.....	21
7.4. Criterios de corrección de la expresión escrita.....	27
7.5. Proceso de recuperación y mejora de calificaciones.....	27
8. Orientaciones pedagógicas.....	28
9. Fomento de la Lectura.....	29
10. Medidas previstas para la consecución de la plena igualdad entre hombres y mujeres.....	29
11. Medidas de Atención a la Diversidad.....	30
11.1. Tratamiento del alumnado con NEE.....	30
12. Actividades complementarias y extraescolares.....	31
13. Recursos Didácticos.....	31
14. Bibliografía.....	31

1. Introducción.

Este documento establece la programación didáctica del módulo de **Programación** que se imparte en el primer curso del **Ciclo Formativo de Grado Superior de Desarrollo de Aplicaciones Multiplataforma**. Dicho módulo se debe impartir por profesores del cuerpo de Enseñanza Secundaria de la especialidad de Informática y Comunicaciones. La duración del módulo es de **256 horas lectivas** y se desarrolla a lo largo de los tres trimestres del curso (**ocho horas semanales**). Su equivalencia en créditos ECTS es de 14 y su código el 0485.

Nuestro sistema educativo en el [Real Decreto 659/2023, de 18 de julio](#), desarrolla la ordenación del Sistema de Formación Profesional y deroga el anterior [Real Decreto 1147/2011, de 29 de julio](#).

El Real Decreto 659/2023 en su *Disposición transitoria segunda, Vigencia de la ordenación de los títulos de formación profesional*, establece que *"Hasta tanto no se proceda reglamentariamente a su modificación, permanecerá vigente la ordenación de los títulos de formación profesional básica, de grado medio o de grado superior recogida en cada uno de los reales decretos por los que se establecen"*. En consecuencia, el Ciclo Formativo correspondiente al título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma está regulado durante el presente curso por las siguientes normativas:

- El [Real Decreto 450/2010](#), de 16 de abril de 2010, que **establece el título** de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma y fija sus enseñanzas mínimas.
- La [Orden de 16 de junio de 2011](#), por la que **se desarrolla el currículo** correspondiente al título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma en **Andalucía**.

Dado que la Formación Profesional debe ofrecer una respuesta eficaz y competente a las necesidades del actual mundo laboral, con un esquema flexible que permita la adecuación a los cambios tecnológicos que se están experimentando en los diferentes procesos productivos, el módulo profesional de Programación contiene parte de la formación necesaria para desempeñar la función de **programación de aplicaciones de propósito general en lenguajes orientados a objetos**. Esta función incluye aspectos tales 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**.

2. Contextualización de la Programación.

El entorno profesional, social, cultural y económico del centro, su ubicación geográfica y las características y necesidades del alumnado, constituyen los ejes prioritarios en la planificación de procesos de enseñanza y aprendizaje. Los centros, departamentos y profesorado deberán tener en cuenta dicho entorno y las posibilidades de desarrollo de éste a la hora de establecer las programaciones de cada uno de los módulos profesionales y del ciclo formativo en su conjunto. Esta programación se ha diseñado teniendo en cuenta los principios y contenidos del **Plan de Centro** y del **Proyecto Educativo** del IES Aguadulce.

2.1. CFGS de Desarrollo de Aplicaciones Multiplataforma en el contexto del IES Aguadulce.

El IES Aguadulce imparte el ciclo formativo de Desarrollo de Aplicaciones Multiplataforma y, por tanto, el módulo de Programación. Se imparte en las modalidades presencial y a distancia, lo que significa una gran ventaja por lo que supone de sinergias positivas entre las dos modalidades, tanto a nivel de desarrollo y producción de materiales didácticos como de experiencia docente del profesorado del departamento de informática.

2.2. Características del Grupo.

En base a la normativa vigente, el grupo lo forman, como máximo, 20 alumnos y alumnas. La experiencia nos muestra que el factor más determinante en el proceso de enseñanza-aprendizaje es la heterogeneidad del grupo, podemos destacar los siguientes grupos:

- Alumnado que procede del ciclo de grado medio Sistemas Microinformáticos y Redes que se imparte en el propio centro IES Aguadulce, así como en centros de localidades cercanas (Vícar, Almería). Este alumnado tiene un buen bagaje tecnológico a nivel informático, pero suele presentar lagunas propias de no haber cursado bachillerato.
- Alumnado procedente de bachillerato, fundamentalmente del propio IES Aguadulce, aunque también de otros centros. Su contexto es prácticamente el contrario al grupo anterior.
- Alumnado procedente de grados universitarios, finalizados o no. No suele ser el grupo más numeroso, pero casi siempre existe.
- Existen otros grupos más minoritarios, como alumnado que lleva tiempo sin estar matriculado en ningún tipo de estudios y procede del mundo laboral o del desempleo, que procede de otros ciclos formativos de la familia de informática o de otras familias profesionales, etc.

Aunque las procedencias del alumnado ya vemos que son heterogéneas, las motivaciones suelen ser bastante coincidentes y se pueden resumir en dos: la **incorporación al mercado laboral con una cualificación** y la **continuación de estudios** (universidad, otros ciclos formativos, oposiciones, cuerpos y fuerzas de seguridad del estado, etc.)

Obviamente la heterogeneidad (diversidad de procedencias e intereses) introduce un factor de dificultad para el docente y su programación, pero al mismo tiempo representa una oportunidad de enriquecimiento mutuo de los diferentes subgrupos dentro del grupo.

3. Competencias y Objetivos Generales del Título.

La competencia general del título consiste en:

Desarrollar, implantar, documentar y mantener aplicaciones informáticas multiplataforma, utilizando tecnologías y entornos de desarrollo específicos, garantizando el acceso a los datos de forma segura y cumpliendo los criterios de «usabilidad» y calidad exigidas en los estándares establecidos.

Las competencias profesionales, personales y sociales de este título 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 bases de datos, interpretando su diseño lógico y verificando integridad, consistencia, seguridad y accesibilidad de los datos.

d) Gestionar entornos de desarrollo adaptando su configuración en cada caso para permitir el desarrollo y despliegue de aplicaciones.

e) Desarrollar aplicaciones multiplataforma con acceso a bases de datos utilizando lenguajes, librerías y herramientas adecuados a las especificaciones.

f) Desarrollar aplicaciones implementando un sistema completo de formularios e informes que permitan gestionar de forma integral la información almacenada.

g) Integrar contenidos gráficos y componentes multimedia en aplicaciones multiplataforma, empleando herramientas específicas y cumpliendo los requerimientos establecidos.

h) Desarrollar interfaces gráficos de usuario interactivos y con la usabilidad adecuada, empleando componentes visuales estándar o implementando componentes visuales específicos.

i) Participar en el desarrollo de juegos y aplicaciones en el ámbito del entretenimiento y la educación empleando técnicas, motores y entornos de desarrollo específicos.

j) Desarrollar aplicaciones para teléfonos, PDA y otros dispositivos móviles empleando técnicas y entornos de desarrollo específicos.

k) Crear ayudas generales y sensibles al contexto, empleando herramientas específicas e integrándolas en sus correspondientes aplicaciones.

l) Crear tutoriales, manuales de usuario, de instalación, de configuración y de administración, empleando herramientas específicas.

m) Empaquetar aplicaciones para su distribución preparando paquetes auto instalables con asistentes incorporados.

n) Desarrollar aplicaciones multiproceso y multihilo empleando librerías y técnicas de programación específicas.

ñ) Desarrollar aplicaciones capaces de ofrecer servicios en red empleando mecanismos de comunicación.

o) Participar en la implantación de sistemas ERP-CRM evaluando la utilidad de cada uno de sus módulos.

- p) Gestionar la información almacenada en sistemas ERP-CRM garantizando su integridad.
- q) Desarrollar componentes personalizados para un sistema ERP-CRM atendiendo a los requerimientos.
- r) Realizar planes de pruebas verificando el funcionamiento de los componentes software desarrollados, según las especificaciones.
- s) Desplegar y distribuir aplicaciones en distintos ámbitos de implantación verificando su comportamiento y realizando las modificaciones necesarias.
- t) Establecer vías eficaces de relación profesional y comunicación con sus superiores, compañeros y subordinados, respetando la autonomía y competencias de las distintas personas.**
- u) Liderar situaciones colectivas que se puedan producir, mediando en conflictos personales y laborales, contribuyendo al establecimiento de un ambiente de trabajo agradable, actuando en todo momento de forma respetuosa y tolerante.
- v) Gestionar su carrera profesional, analizando las oportunidades de empleo, autoempleo y de aprendizaje.
- w) Mantener el espíritu de innovación y actualización en el ámbito de su trabajo para adaptarse a los cambios tecnológicos y organizativos de su entorno profesional.**
- x) Crear y gestionar una pequeña empresa, realizando un estudio de viabilidad de productos, de planificación de la producción y de comercialización.
- y) Participar de forma activa en la vida económica, social y cultural, con una actitud crítica y responsable.

Según normativa, la formación de este módulo contribuye a alcanzar las competencias (a), (e), (f), (i), (j), (t) y (w) del ciclo formativo.

Los objetivos generales de este ciclo son los siguientes:

- a) Ajustar la configuración lógica del sistema analizando las necesidades y criterios establecidos para configurar y explotar sistemas informáticos.
- b) Identificar las necesidades de seguridad analizando vulnerabilidades y verificando el plan preestablecido para aplicar técnicas y procedimientos relacionados con la seguridad en el sistema.
- c) Interpretar el diseño lógico de bases de datos, analizando y cumpliendo las especificaciones relativas a su aplicación, para gestionar bases de datos.
- d) Instalar y configurar módulos y complementos, evaluando su funcionalidad, para gestionar entornos de desarrollo.
- e) Seleccionar y emplear lenguajes, herramientas y librerías, interpretando las especificaciones para desarrollar aplicaciones multiplataforma con acceso a bases de datos.**
- f) Gestionar la información almacenada, planificando e implementando sistemas de formularios e informes para desarrollar aplicaciones de gestión.
- g) Seleccionar y utilizar herramientas específicas, lenguajes y librerías, evaluando sus posibilidades y siguiendo un manual de estilo, para manipular e integrar en aplicaciones multiplataforma contenidos gráficos y componentes multimedia.
- h) Emplear herramientas de desarrollo, lenguajes y componentes visuales, siguiendo las especificaciones y verificando interactividad y usabilidad, para desarrollar interfaces gráficos de usuario en aplicaciones multiplataforma.

i) Seleccionar y emplear técnicas, motores y entornos de desarrollo, evaluando sus posibilidades, para participar en el desarrollo de juegos y aplicaciones en el ámbito del entretenimiento.

j) Seleccionar y emplear técnicas, lenguajes y entornos de desarrollo, evaluando sus posibilidades, para desarrollar aplicaciones en teléfonos, PDA y otros dispositivos móviles.

k) Valorar y emplear herramientas específicas, atendiendo a la estructura de los contenidos, para crear ayudas generales y sensibles al contexto.

l) Valorar y emplear herramientas específicas, atendiendo a la estructura de los contenidos, para crear tutoriales, manuales de usuario y otros documentos asociados a una aplicación.

m) Seleccionar y emplear técnicas y herramientas, evaluando la utilidad de los asistentes de instalación generados, para empaquetar aplicaciones.

n) Analizar y aplicar técnicas y librerías específicas, simulando diferentes escenarios, para desarrollar aplicaciones capaces de ofrecer servicios en red.

ñ) Analizar y aplicar técnicas y librerías de programación, evaluando su funcionalidad para desarrollar aplicaciones multiproceso y multihilo.

o) Reconocer la estructura de los sistemas ERP-CRM, identificando la utilidad de cada uno de sus módulos, para participar en su implantación.

p) Realizar consultas, analizando y evaluando su alcance, para gestionar la información almacenada en sistemas ERP-CRM.

q) Seleccionar y emplear lenguajes y herramientas, atendiendo a los requerimientos, para desarrollar componentes personalizados en sistemas ERP-CRM.

r) Verificar los componentes software desarrollados, analizando las especificaciones, para completar un plan de pruebas.

s) Establecer procedimientos, verificando su funcionalidad, para desplegar y distribuir aplicaciones.

t) Describir los roles de cada uno de los componentes del grupo de trabajo, identificando en cada caso la responsabilidad asociada, para establecer las relaciones profesionales más convenientes.

u) Identificar formas de intervención ante conflictos de tipo personal y laboral, teniendo en cuenta las decisiones más convenientes, para garantizar un entorno de trabajo satisfactorio.

v) Identificar y valorar las oportunidades de promoción profesional y de aprendizaje, analizando el contexto del sector, para elegir el itinerario laboral y formativo más conveniente.

w) Identificar los cambios tecnológicos, organizativos, económicos y laborales en su actividad, analizando sus implicaciones en el ámbito de trabajo, para mantener el espíritu de innovación.

x) Reconocer las oportunidades de negocio, identificando y analizando demandas del mercado para crear y gestionar una pequeña empresa.

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

Según normativa, la formación de este módulo contribuye a alcanzar los objetivos generales e), j), q), w) del ciclo formativo.

4. Unidades de Competencia y Resultados de Aprendizaje.

El módulo profesional de Programación **no se relaciona con ninguna unidad de competencia completa**. En cuanto a los resultados de aprendizaje, la Orden de 16 de Junio de 2011 establece los siguientes resultados de aprendizaje (RA) y sus correspondientes criterios de evaluación (CE):

- RA01. Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.
- RA02. Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.
- RA03. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.
- RA04. Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.
- RA05. Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librería de clases.
- RA06. Escribe programas que manipulen información seleccionando y utilizando tipos avanzados de datos.
- RA07. Desarrolla programas aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.
- RA08. Utiliza bases de datos orientadas a objetos, analizando sus características y aplicando técnicas para mantener la persistencia de la información.
- RA09. Gestiona información almacenada en bases de datos relacionales manteniendo la integridad y consistencia de los datos.

(Ver los CE en la [Orden de 16 de Junio de 2011, BOJA del 1 de agosto de 2011](#))

5. Unidades de Trabajo.

5.1. *Contenidos y Secuencia de las Unidades de Trabajo.*

Los contenidos básicos del módulo se especifican en la Orden de 16 de junio. Estos contenidos se han agrupado en **10 unidades de trabajo** para su tratamiento.

A continuación, se detalla dicha agrupación:

UT01. Introducción a la programación.

- Programas: buscando una solución.
- Concepto de algoritmo y paradigma de programación.
- Generalidades de los lenguajes de programación: tipos de lenguajes de programación.
- Introducción al lenguaje de programación Java e historia del mismo. Conceptos básicos sobre Java y su composición. JDK y JRE.
- Introducción a los entornos integrados de desarrollo (IDE). Tipos de IDE, comerciales y libres.
- Instalación y utilización de NetBeans como IDE de desarrollo. Creación de proyectos. estructura y componentes del IDE.
- Creación de mi primer programa.
- Variables e identificadores.
- Los tipos de datos.
- Literales de los tipos primitivos.
- Operadores y expresiones. Operadores de asignación y aritméticos. Expresiones aritméticas. Operadores de relación y lógicos. Expresiones lógicas. Operador ternario. Operadores de bits. Operaciones básicas con cadenas.
- Conversiones de tipo.
- Entrada y salida por el terminal.
- Comentarios.

UT02. Uso de estructuras de control.

- Sentencias y bloques.
- Estructuras condicionales. Condicionales en Java: if/else y switch.
- Estructuras de repetición. Bucles en Java: for, for/in, while, do-while.
- Estructuras de salto incondicional.
- Prueba de programas.
- Depuración de programas.

UT03. Utilización de objetos.

- Clases y Objetos. Características de los objetos. Concepto de clase e instancia de objeto. Concepto de propiedad y método. Interacción entre objetos.
- Utilización de objetos en Java. Declaración de objetos, instanciación y uso. Acceso a los miembros de una clase.
- Propiedades de la clase. Propiedades estáticas o de clase.
- Métodos. Parámetros y retorno de valores. Métodos constructores de clase. Operador this. Utilización de métodos.
- Captura de excepciones lanzadas por métodos de objetos.
- Bibliotecas de objetos (paquetes). Concepto de paquete. Empaquetado y jerarquía de clases. Utilización e importación de paquetes y bibliotecas.
- Ampliación de uso de la entrada/salida por terminal.
- Documentación del código. Uso y consulta de la documentación tipo *JavaDoc*.

UT04. Cadenas de caracteres y arrays.

- Estructuras de almacenamiento estáticas.
- Cadenas de caracteres. Operaciones avanzadas con cadenas de caracteres.
- Expresiones regulares.
- Arrays unidimensionales y multidimensionales. Uso. Iteración sobre un array.
- Operaciones sobre arrays: ordenación, búsqueda, búsqueda binaria, rellenado, copia total y parcial.

UT05. Introducción a la programación orientada a objetos. Desarrollo de clases.

- Fundamentos de la programación orientada a objetos. Características y beneficios.
- Concepto de clase e instancia de objeto. Concepto de propiedad y método. Interacción entre objetos.
- Utilización de objetos en Java. Declaración de clases, instanciación y uso. Estructura y miembros de una clase.
- Estructura y miembros de una clase. Propiedades estáticas o de clase.
- Métodos. Parámetros y retorno de valores. Métodos constructores de clase. Operador this. Métodos estáticos o de clase vs. métodos de instancia.
- Encapsulación, control de acceso y visibilidad.
- Utilización de los métodos y atributos de una clase.
- Constructores.
- Lanzamiento de excepciones desde métodos.
- Uso de paquetes y bibliotecas incluidas en el JRE, y bibliotecas de terceros.
- Ampliación de uso de la entrada/salida por terminal.
- Documentación del código. Creación de documentación tipo *JavaDoc*.

UT06. Profundización en la programación orientada a objetos. Utilización avanzada de clases.

- Control de acceso y visibilidad (pública, privada, protegida o de paquete). Modificadores de acceso a variables y métodos.
- Concepto de paquete. Empaquetado y jerarquía de clases. Creación de paquetes. Importación de paquetes.
- Copia de objetos.
- Sobrecarga de métodos.
- Herencia y Composición. Relaciones entre clases. Subclases y superclases.
- Composición a nivel práctico. Clases anidadas o internas.
- Herencia a nivel práctico. Sintaxis y acceso. Sobrescritura y sobrecarga de métodos heredados. Constructores heredados. Clase Object. Herencia múltiple.
- Clases abstractas.
- Interfaces.
- Polimorfismo. Ligadura dinámica.

UT07. Estructuras de datos complejas.

- Clases y métodos genéricos.
- Colecciones. Conjuntos, listas. Pares de clave/valor (tablas hash, mapas o arrays asociativos).
- Operaciones sobre tipos de datos complejos.
- Iteradores.
- Algoritmos.

UT08. Almacenamiento externo de datos (I). Archivos.

- Concepto de flujos de datos. Flujos e datos predefinidos.
- Tipos de flujos de datos. Clases asociadas a los flujos de datos.
- Acceso, creación y modificación de ficheros binarios y en modo texto. Tipos de acceso a ficheros: secuencial o aleatorio.
- Utilización de los sistemas de ficheros: creación de directorios, listados de archivos, etc.
- Serialización y persistencia de datos.
- Concepto de XML. Estructura de un documento XML. Especificación de documentos XML. DTD y XSD. Tratamiento y almacenamiento de documentos XML.

UT09. Almacenamiento externo de datos (II). Bases de datos.

- Introducción a las bases de datos relacionales en Java. Instalación y creación del esquema de base de datos y sus tablas.
- Conectores y drivers. JDBC para acceso a la base de datos.
- Establecimiento de conexión con la base de datos.
- Operaciones CRUD (crear, leer, actualizar y borrar) con la base de datos a través de JDBC. Selección de registro y parametrización de consultas
- Introducción a las bases de datos orientadas a objetos. Bases de datos orientadas a objetos puras y objeto-relacionales.

UT10. Desarrollo de interfaces de usuario.

- Introducción a las interfaces de usuario. Interfaces gráficas AWT, Swing y JavaFX.
- Asistentes para creación de interfaces.
- Contenedores y elementos gráficos utilizados en el diseño de interfaces.
- Gestión de eventos de interacción con el entorno gráfico.

5.2. Relación de Resultados de Aprendizaje con las Unidades de Trabajo Propuestas.

A continuación, se detalla la relación de cada una de las unidades de la programación con los RA establecidos en la legislación vigente:

UT01. Introducción a la programación.

- RA01. Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.

UT02. Uso de estructuras de control.

- RA03. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.

UT03. Utilización de objetos.

- RA02. Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.
- RA03. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.
- RA05. Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases.

UT04. Cadenas de caracteres y arrays.

- RA06. Escribe programas que manipulen información seleccionando y utilizando tipos avanzados de datos.

UT05. Introducción a la programación orientada a objetos. Desarrollo de clases.

- RA04. Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.
- RA03. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.
- RA07. Desarrolla programas aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.

UT06. Profundización en la programación orientada a objetos. Utilización avanzada de clases.

- RA04. Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.
- RA07. Desarrolla programas aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.

UT07. Estructuras de datos complejas.

- RA06. Escribe programas que manipulen información seleccionando y utilizando tipos avanzados de datos.

UT08. Almacenamiento de datos (I). Archivos.

- RA05. Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librería de clases.

UT09. Almacenamiento de datos (II). Bases de datos.

- RA08. Utiliza bases de datos orientadas a objetos, analizando sus características y aplicando técnicas para mantener la persistencia de la información.
- RA09. Gestiona información almacenada en bases de datos relacionales manteniendo la integridad y consistencia de los datos.

UT10. Desarrollo de interfaces de usuario.

- RA05. Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librería de clases.

5.3. Temporalización.

La temporalización prevista para las unidades de trabajo se presenta en la siguiente tabla resumen:

- **Primer trimestre:**
 - UT01. Introducción a la programación.
 - UT02. Uso de estructuras de control.
 - UT03. Utilización de objetos.
 - UT04. Cadenas de caracteres y arrays.
- **Segundo trimestre:**
 - UT04. Cadenas de caracteres y arrays.
 - UT05. Introducción a la programación orientada a objetos. Desarrollo de clases.
 - UT06. Profundización en la programación orientada a objetos. Utilización avanzada de clases.
 - UT07. Estructuras de datos complejas.
- **Tercer trimestre:**
 - UT08. Almacenamiento de datos (I). Archivos.
 - UT09. Almacenamiento de datos (II). Bases de datos.
 - UT10. Desarrollo de interfaces de usuario.

En cualquier caso, esta temporalización está condicionada al alumnado que integre el grupo, por lo que esta programación tiene una naturaleza dinámica que se irá adaptando a la evolución en el proceso de enseñanza-aprendizaje de los alumnos y alumnas del módulo. El alumnado que parte de cero en informática/programación es natural que necesite más tiempo del planificado en un principio para abordar las unidades 1, 2 3 y 4, lo normal es que se recupere ese tiempo una vez que se han superado dichas unidades.

6. Metodología.

Según la **Orden de 16 de junio de 2011**, por la que se desarrolla el currículo correspondiente al título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma en Andalucía, 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.

6.1. *Utilización del aula virtual como apoyo a la docencia*

A lo largo del curso se utilizará el Aula Virtual como apoyo a la docencia reglada. Se fomentará un mayor uso conforme el alumnado vaya promocionando de curso. En general, su utilización responderá a las siguientes pautas:

- Se definirá la estructura del curso en unidades, temas, secciones, etc.
- Se procurará que el desarrollo de los contenidos del curso esté disponible en el Aula Virtual, sobre todo en los niveles en los que no se disponga de un libro de texto o materiales de referencia.
- Se proporcionarán recursos educativos para el tratamiento de los contenidos programados (documentos explicativos, materiales audiovisuales, cuestionarios, actividades resueltas, recursos de refuerzo y de ampliación, modelos de pruebas, etc.).
- Se podrán establecer tareas y otras actividades de evaluación cuya entrega quede registrada en el Aula Virtual.

6.2. Estrategias metodológicas en el aula

La **metodología didáctica** que se va a seguir para llevar a cabo el proceso de enseñanza-aprendizaje se regirá bajo las siguientes pautas, aunque éstas podrán sufrir las excepciones que se requieran en función de las necesidades del momento:

1. Se realiza una **breve introducción general** a la utilidad de las herramientas, metodologías, conceptos, procesos, etc. que se van a emplear.
2. Se comienza a trabajar con algunos **ejemplos básicos** que vayan usando poco a poco esas nuevas herramientas, comenzando con los casos más elementales.
3. Mediante el planteamiento de ciertos problemas y situaciones, se intenta **crear la necesidad de nuevas herramientas más sofisticadas** para que el propio alumnado intuya que sería útil la existencia de esa herramienta o concepto para mejorar el modo en que se podría resolver el problema planteado. De este modo el propio alumnado está de alguna manera anticipando lo que va a venir a continuación.
4. Se **introduce ese nuevo concepto o herramienta** en los ejemplos que se están desarrollando para **facilitar la resolución de un problema**, provocando que sea el propio alumnado quien lo pida o necesite.
5. Se procura que el alumnado **experimente con esas nuevas herramientas o conceptos de manera intuitiva e incompleta** para resolver nuevos problemas más complejos y así **descubra la necesidad de una mayor sistematización y documentación** en la explicación de esas nuevas herramientas y conceptos. De este modo el alumnado se encontrará **motivado y preparado para asimilar y aplicar esos nuevos conocimientos sin que le parezcan ajenos, extraños o innecesarios**.
6. Se lleva a cabo una **exposición más sistemática y demostrativa** de esos conceptos o herramientas para **sistematizar, completar y documentar** lo que el alumnado ya ha aprendido a utilizar de manera práctica e intuitiva.
7. Se plantean diversos problemas y ejercicios donde el alumnado podrá **poner en práctica todo lo aprendido tanto de manera intuitiva como sistemática**.

Todo el proceso de enseñanza-aprendizaje estará guiado por la realización práctica de los contenidos de cada unidad, introduciendo siempre los conceptos teóricos por medio de prácticas guiadas que darán paso posteriormente a prácticas autónomas.

De lo anterior se concluye que el alumnado tendrá que realizar trabajos y actividades, algunos de los cuales serán de carácter obligatorio, y otros de carácter opcional. Se procurará que existan prácticas individuales, pero también grupales, ya que tan importante es alcanzar los resultados de aprendizaje del módulo, como lo es alcanzar las soft skills (competencias personales y sociales).

Se dispone de un ordenador personal para cada alumno o alumna. Ese ordenador puede compartirse con otro alumnado durante el turno de tarde, lo que les obligará a co-responsabilizarse de sus equipos.

El alumnado es responsable de salvaguardar las actividades y trabajos que va realizando a lo largo de todo el curso. El profesorado solicitará al alumnado las actividades y trabajos realizados para su posterior evaluación. El alumnado deberá utilizar algún soporte de almacenamiento propio (memoria USB, conexión con la nube o similar) para salvaguardar el trabajo de cara a continuar con él en sesiones posteriores o para la entrega al profesorado. De esta forma se evita que el alumnado pierda el trabajo si el equipo en el que trabaja habitualmente se estropea o si otro alumno u otra alumna lo borra accidentalmente.

Se fomentará la realización de trabajos de investigación en los que habrá que contrastar informaciones de diferentes fuentes (material proporcionado en clase, documentación técnica, y búsquedas por Internet). También se fomentará el intercambio de información entre los miembros del grupo.

Tanto para la realización de actividades como para la realización de trabajos de investigación, se dejará tiempo suficiente en clase para su realización. Se hará un seguimiento en clase del trabajo del alumnado, con el fin de evaluar apropiadamente la adquisición de los resultados de aprendizaje. Así mismo, se fomentará que el alumnado haga exposiciones de uno o varios de los trabajos que haya realizado con el mismo fin anterior. Se incluirán exposiciones orales para mejorar la destreza comunicativa.

Los trabajos o actividades en grupo se fundamentarán en los principios del **aprendizaje cooperativo**, buscando la implicación en el desarrollo de la actividad práctica o trabajo, sin perjuicio de la distribución de roles entre los participantes.

A modo de síntesis y sin perjuicio del necesario rigor conceptual, se tendrá siempre presente la consideración de que **lo importante es desarrollar las capacidades para abordar realizaciones prácticas similares a aquellas que se va a tener que afrontar en la vida profesional**, una vez concluida la etapa formativa.

7. Evaluación.

La [Orden de 29 de septiembre de 2010](#), por la que se regula la **evaluación, certificación, acreditación y titulación académica** del alumnado que cursa enseñanzas de formación profesional inicial que forma parte del sistema educativo en la Comunidad Autónoma de Andalucía, establece en su Artículo 2, Apartado 5, que el departamento de familia profesional, a través del equipo educativo de cada uno de los ciclos formativos, desarrollará el currículo mediante la elaboración de las correspondientes programaciones didácticas de los módulos profesionales. Su elaboración se realizará siguiendo las directrices marcadas en el Proyecto Educativo del Centro, especialmente en lo referente a los procedimientos y criterios de evaluación comunes para las enseñanzas de formación profesional inicial.

A tal efecto, el **Proyecto Educativo**, en su apartado de Evaluación, recoge los puntos principales sobre el proceso de evaluación en los ciclos formativos de la familia de Informática y Comunicaciones. El proceso de evaluación explicitado en esta programación se ajusta a esos puntos, que a continuación son desarrollados de manera concreta para este módulo.

7.1. Criterios de Evaluación.

A continuación, se indican los criterios de evaluación (a partir de ahora CE) asociados a los diferentes resultados de aprendizaje (a partir de ahora RA) según normativa.

Resultado de aprendizaje	Criterios de evaluación
RA01) Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.	<ul style="list-style-type: none">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.
RA02) Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.	<ul style="list-style-type: none">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.
RA03) Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.	<ul style="list-style-type: none">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

Resultado de aprendizaje	Criterios de evaluación
	<p>utilizando diferentes estructuras de control.</p> <p>f) Se han probado y depurado los programas.</p> <p>g) Se ha comentado y documentado el código.</p>
RA04) Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.	<p>a) Se ha reconocido la sintaxis, estructura y componentes típicos de una clase.</p> <p>b) Se han definido clases.</p> <p>c) Se han definido propiedades y métodos.</p> <p>d) Se han creado constructores.</p> <p>e) Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.</p> <p>f) Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros.</p> <p>g) Se han definido y utilizado clases heredadas.</p> <p>h) Se han creado y utilizado métodos estáticos.</p> <p>i) Se han definido y utilizado interfaces.</p> <p>j) Se han creado y utilizado conjuntos y librerías de clases.</p>
RA05) Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librería de clases.	<p>a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información.</p> <p>b) Se han aplicado formatos en la visualización de la información.</p> <p>c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.</p> <p>d) Se han utilizado ficheros para almacenar y recuperar información.</p> <p>e) Se han creado programas que utilicen diversos métodos de acceso al contenido de los ficheros.</p> <p>f) Se han utilizado las herramientas del entorno de desarrollo para crear interfaces gráficos de usuario simples.</p> <p>g) Se han programado controladores de eventos.</p> <p>h) Se han escrito programas que utilicen interfaces gráficos para la entrada y salida de información.</p>
RA06) Escribe programas que manipulen información seleccionando y utilizando tipos avanzados de datos.	<p>a) Se han escrito programas que utilicen arrays</p> <p>b) Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados.</p> <p>c) Se han utilizado listas para almacenar y procesar información.</p> <p>d) Se han utilizado iteradores para recorrer los elementos de las listas.</p> <p>e) Se han reconocido las características y</p>

Resultado de aprendizaje	Criterios de evaluación
	<p>ventajas de cada una de la colecciones de datos disponibles.</p> <p>f) Se han creado clases y métodos genéricos.</p> <p>g) Se han utilizado expresiones regulares en la búsqueda de patrones en cadenas de texto.</p> <p>h) Se han identificado las clases relacionadas con el tratamiento de documentos XML.</p> <p>i) Se han realizado programas que realicen manipulaciones sobre documentos XML.</p>
<p>RA07. Desarrolla programas aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.</p>	<p>a) Se han identificado los conceptos de herencia, superclase y subclase.</p> <p>b) Se han utilizado modificadores para bloquear y forzar la herencia de clases y métodos.</p> <p>c) Se ha reconocido la incidencia de los constructores en la herencia.</p> <p>d) Se han creado clases heredadas que sobrescriban la implementación de métodos de la superclase.</p> <p>e) Se han diseñado y aplicado jerarquías de clases.</p> <p>f) Se han probado y depurado las jerarquías de clases.</p> <p>g) Se han realizado programas que implementen y utilicen jerarquías de clases.</p> <p>h) Se ha comentado y documentado el código.</p>
<p>RA08) Utiliza bases de datos orientadas a objetos, analizando sus características y aplicando técnicas para mantener la persistencia de la información.</p>	<p>a) Se han identificado las características de las bases de datos orientadas a objetos.</p> <p>b) Se ha analizado su aplicación en el desarrollo de aplicaciones mediante lenguajes orientados a objetos.</p> <p>c) Se han instalado sistemas gestores de bases de datos orientados a objetos.</p> <p>d) Se han clasificado y analizado los distintos métodos soportados por los sistemas gestores para la gestión de la información almacenada.</p> <p>e) Se han creado bases de datos y las estructuras necesarias para el almacenamiento de objetos.</p> <p>f) Se han programado aplicaciones que almacenen objetos en las bases de datos creadas.</p> <p>g) Se han realizado programas para recuperar, actualizar y eliminar objetos de las bases de datos.</p> <p>h) Se han realizado programas para almacenar y gestionar tipos de datos estructurados, compuestos y relacionados.</p>

Resultado de aprendizaje	Criterios de evaluación
RA09) Gestiona información almacenada en bases de datos relacionales manteniendo la integridad y consistencia de los datos.	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 creado programas para recuperar y mostrar información almacenada en bases de datos. e) Se han efectuado borrados y modificaciones sobre la información almacenada. f) Se han creado aplicaciones que ejecuten consultas sobre bases de datos. g) Se han creado aplicaciones para posibilitar la gestión de información presente en bases de datos relacionales.

7.2. Instrumentos de Evaluación.

Entendiendo como Instrumentos de Evaluación aquellos recursos que permiten la recogida o registro de información sobre el desarrollo del aprendizaje por parte del alumnado, se determina que para el presente módulo se tendrán en cuenta los siguientes instrumentos:

- **Tareas y actividades obligatorias de carácter práctico** (conocidas a partir de ahora como “tareas” o “prácticas”) que el alumnado deberá realizar para demostrar la adquisición de los resultados de aprendizaje. En algunos casos estas tareas se realizarán durante una sesión de clase con un tiempo límite predefinido. En otros casos, para tareas más extensas, éstas se realizarán tanto en el aula como en casa, a lo largo de uno o varios días, debiendo ser entregadas en un determinado plazo.
- **Cuestionarios teórico/prácticos** (conocidos a partir de ahora como “cuestionarios”) que el alumnado deberá realizar también para demostrar la adquisición de los resultados de aprendizaje.
- **Tareas de ampliación (voluntarias)**, que el alumnado podrá realizar para ampliar y reforzar sus conocimientos en un área concreta.

En relación con lo anterior:

- **Cada actividad o instrumento de evaluación que se realice recogerá los RA cubiertos por la unidad que se pretenden evaluar**, ya sea total o parcialmente, detallando los CE específicos que se evaluarán para cada RA.
- El conjunto de actividades prácticas a realizar, dependerá de la evolución del alumnado y de la unidad que se esté trabajando en cada momento. No obstante, **se procurará desarrollar, siempre que sea posible, una actividad o tarea práctica por unidad**, versando dicha actividad sobre los RA de la unidad en curso o recién finalizada y, dada la naturaleza acumulativa de la materia, parte de los RA/CE cubiertos por unidades anteriores.

- Cada actividad o trabajo práctico tendrá un **plazo de entrega** fuera del cual, dicha actividad no puede ser entregada por el alumnado, salvo circunstancias concretas: enfermedad o circunstancias personales justificables que hayan impedido que el alumno o la alumna realice la actividad en el periodo previsto. Si se dieran dichas circunstancias concretas, el alumnado debería justificar apropiadamente el motivo de manera documental. Esas actividades podrán realizarse en clase y, en algunos casos, fuera de clase. Para cada actividad concreta se indicará cómo se va a proceder. En cualquier caso, el profesorado siempre se reserva la posibilidad de exigir, si lo considera necesario, una defensa del trabajo haciendo que el alumno o alumna conteste a algunas preguntas sobre la realización de la práctica o bien que complete la práctica con algunos detalles adicionales basándose en lo que se ha presentado.
- Las actividades podrán tener un diseño **individual o grupal**, aunque lo habitual será el trabajo individual.
- Normalmente se procurará realizar al menos **un cuestionario o trabajo teórico/práctico por unidad**. En cualquier caso, como mínimo, se realizará **al menos un cuestionario teórico/práctico por trimestre**, en cuyo caso se englobaría los RA y CE cubiertos por todas las unidades impartidas en ese trimestre.
- En el caso de los **cuestionarios teórico/prácticos**, se ofrecerá al alumno/a la oportunidad de intentar **recuperar o subir nota al menos una vez en cada trimestre**. De esta manera la evaluación del proceso de recuperación está incluida en el **procedimiento de evaluación continua** a lo largo del curso y no tratando las unidades como elementos aislados que se abandonan una vez impartidas.
- A efectos de cómputo de la calificación final, **una tarea (voluntaria u obligatoria) no entregada** computará como **cero (0)**. De igual forma, un **cuestionario teórico/práctico no realizado** computará como **cero (0)**.
- **En caso de comprobarse que el alumno/a no ha realizado su trabajo (cuestionario, tarea, etc.) de manera legítima o fraudulenta** (copia de otro compañero/a, falsificación de resultados, plagio de otras fuentes, etc.), tendrá una puntuación de **un cero (0) en dicho trabajo**. Para asegurarse de esto, el profesorado podrá hacer las comprobaciones y preguntas que estime convenientes, pudiendo exigir, si lo considera necesario, una defensa de su trabajo delante de él. Este tipo de comportamientos fraudulentos **no estarán exentos de otras medidas disciplinarias** que se puedan acometer en función de la gravedad del acto realizado.
- En el caso de las **tareas o actividades prácticas**, dada la naturaleza acumulativa de los conocimientos y habilidades que se han de adquirir en este módulo, si no se supera algún CE evaluado anteriormente, **algunas de las tareas prácticas de unidades posteriores permitirán recuperar todos o parte de los RA/CE no superados en tareas anteriores** con una calificación máxima de 5 en esos CE (en cada caso, se avisará apropiadamente y con antelación, qué prácticas permiten recuperar RA/CE no superadas aún). Si se desea mejorar esa calificación práctica a efectos de obtener una mayor nota global en el módulo, habrá que acudir al periodo de **recuperación y mejora de calificaciones** del mes de junio para poder hacerlo. De esta manera también se permite llevar a cabo una **evaluación y recuperación continua** en las tareas prácticas sin que dé lugar a una acumulación excesiva de actividades de evaluación a lo largo del curso pues se aprovechan tareas que evalúan habilidades más complejas para comprobar si se han adquirido destrezas más simples. Esta forma de poder recuperar las tareas prácticas a lo largo del curso hace posible que se lleve a cabo una evaluación realmente continua, pues no se dejan atrás las unidades impartidas, sino que siguen presentes en las siguientes al considerarse “implícitas” en algunas de las unidades posteriores.
- **El conjunto de las actividades de evaluación realizadas para cada unidad permitirá discernir si los RA cubiertos por esa unidad han sido superados o no**. Algunos CE de esos RA serán evaluados mediante tareas prácticas y otros mediante cuestionarios.

- En el caso de aquellos CE que sean evaluados mediante cuestionarios, será necesario superarlos los CE a través del cuestionario asociado a una unidad (o a varias, si se ha decidido hacer un único cuestionario para varias unidades).
- Del mismo modo, para los CE evaluados mediante tareas, será necesario superarlos a través de una tarea práctica. Como se ha visto en el punto anterior, dada la naturaleza acumulativa de esta materia, será posible superar RA y CE no superados en unidades anteriores mediante actividades de unidades posteriores.
- Para que el alumnado esté siempre apropiadamente informado, **se detallará, en cada actividad que se desarrolle, qué RA/CE pueden superarse (o recuperarse) con ella.** Se informará de ello anticipadamente, al planificarse la actividad, así como en el propio enunciado o texto de la actividad.

7.3. Criterios de calificación.

La siguiente tabla muestra la **ponderación de resultados de aprendizaje (RA) y criterios de evaluación (CE)**, así como su **relación con las unidades y los diferentes instrumentos de evaluación (IE)**. Dicha tabla es un referente inicial y puede actualizarse para adaptarse al contexto y a las características particulares del grupo o al progreso del curso.

Resultados de Aprendizaje	Porcentaje	Criterios de Evaluación	Ponderación por CE	IE	Unidades
1. Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.	7%	a) <i>Se han identificado los bloques que componen la estructura de un programa informático.</i>	0,25%	Tareas	UT01
				Cuestionarios	
		b) <i>Se han creado proyectos de desarrollo de aplicaciones.</i>	0,25%	Tareas	
				Cuestionarios	
		c) <i>Se han utilizado entornos integrados de desarrollo.</i>	0,25%	Tareas	
				Cuestionarios	
		d) <i>Se han identificado los distintos tipos de variables y la utilidad específica de cada uno.</i>	1,00%	Tareas	
				Cuestionarios	
		e) <i>Se ha modificado el código de un programa para crear y utilizar variables.</i>	1,50%	Tareas	
				Cuestionarios	
2. Escribe y prueba programas sencillos, reconociendo y aplicando los fun-	10%	f) <i>Se han creado y utilizado constantes y literales.</i>	1,00%	Tareas	UT03
				Cuestionarios	
		g) <i>Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje.</i>	1,50%	Tareas	
				Cuestionarios	
		h) <i>Se ha comprobado el funcionamiento de las conversiones de tipos explícitas e implícitas.</i>	1,00%	Tareas	
				Cuestionarios	
		i) <i>Se han introducido comentarios en el código.</i>	0,25%	Tareas	
				Cuestionarios	
		a) <i>Se han identificado los fundamentos de la programación orientada a objetos.</i>	0,50%	Tareas	
				Cuestionarios	
		b) <i>Se han escrito programas simples.</i>	0,25%	Tareas	
				Cuestionarios	

damentos de la programación orientada a objetos.		c) <i>Se han instanciado objetos a partir de clases pre-definidas.</i>	1,50%	Tareas Cuestionarios	
		d) <i>Se han utilizado métodos y propiedades de los objetos.</i>	2,50%	Tareas Cuestionarios	
		e) <i>Se han escrito llamadas a métodos estáticos.</i>	1,50%	Tareas Cuestionarios	
		f) <i>Se han utilizado parámetros en la llamada a métodos.</i>	2,00%	Tareas Cuestionarios	
		g) <i>Se han incorporado y utilizado librerías de objetos.</i>	0,50%	Tareas Cuestionarios	
		h) <i>Se han utilizado constructores.</i>	1,00%	Tareas Cuestionarios	
		i) <i>Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples.</i>	0,25%	Tareas Cuestionarios	
3. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.	14%	a) <i>Se ha escrito y probado código que haga uso de estructuras de selección.</i>	2,00%	Tareas Cuestionarios	UT02
		b) <i>Se han utilizado estructuras de repetición.</i>	3,00%	Tareas Cuestionarios	
		c) <i>Se han reconocido las posibilidades de las sentencias de salto.</i>	0,50%	Tareas Cuestionarios	
		d) <i>Se ha escrito código utilizando control de excepciones.</i>	3,00%	Tareas Cuestionarios	UT03/ UT05
		e) <i>Se han creado programas ejecutables utilizando diferentes estructuras de control.</i>	3,00%	Tareas Cuestionarios	UT02
		f) <i>Se han probado y depurado los programas.</i>	2,00%	Tareas Cuestionarios	
		g) <i>Se ha comentado y documentado el código.</i>	0,50%	Tareas Cuestionarios	UT02
4. Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.	17%	a) <i>Se ha reconocido la sintaxis, estructura y componentes típicos de una clase.</i>	1,00%	Tareas Cuestionarios	UT05
		b) <i>Se han definido clases.</i>	3,00%	Tareas Cuestionarios	
		c) <i>Se han definido propiedades y métodos.</i>	2,00%	Tareas Cuestionarios	
		d) <i>Se han creado constructores.</i>	2,00%	Tareas Cuestionarios	
		e) <i>Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.</i>	1,00%	Tareas Cuestionarios	
		f) <i>Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros.</i>	2,00%	Tareas Cuestionarios	
					UT05

		g) <i>Se han definido y utilizado clases heredadas.</i>	2,00%	Tareas Cuestionarios	UT06
		h) <i>Se han creado y utilizado métodos estáticos.</i>	2,00%	Tareas Cuestionarios	UT05
		i) <i>Se han definido y utilizado interfaces.</i>	1,00%	Tareas Cuestionarios	UT06
		j) <i>Se han creado y utilizado conjuntos y librerías de clases.</i>	1,00%	Tareas Cuestionarios	
5. Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases	12%	a) <i>Se ha utilizado la consola para realizar operaciones de entrada y salida de información.</i>	1,00%	Tareas Cuestionarios	UT03
		b) <i>Se han aplicado formatos en la visualización de la información.</i>	1,00%	Tareas Cuestionarios	
		c) <i>Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.</i>	1,00%	Tareas Cuestionarios	
		d) <i>Se han utilizado ficheros para almacenar y recuperar información.</i>	3,00%	Tareas Cuestionarios	UT08
		e) <i>Se han creado programas que utilicen diversos métodos de acceso al contenido de los ficheros.</i>	3,00%	Tareas Cuestionarios	
		f) <i>Se han utilizado las herramientas del entorno de desarrollo para crear interfaces gráficos de usuario simples.</i>	1,00%	Tareas Cuestionarios	UT10
		g) <i>Se han programado controladores de eventos.</i>	1,00%	Tareas Cuestionarios	
		h) <i>Se han escrito programas que utilicen interfaces gráficos para la entrada y salida de información.</i>	1,00%	Tareas Cuestionarios	
6. Escribe programas que manipulen información seleccionando y utilizando tipos avanzados de datos.	25%	a) <i>Se han escrito programas que utilicen arrays.</i>	7,00%	Tareas Cuestionarios	UT04
		b) <i>Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados.</i>	2,00%	Tareas Cuestionarios	UT07
		c) <i>Se han utilizado listas para almacenar y procesar información.</i>	3,00%	Tareas Cuestionarios	UT07
		d) <i>Se han utilizado iteradores para recorrer los elementos de las listas.</i>	2,50%	Tareas Cuestionarios	
		e) <i>Se han reconocido las características y ventajas de cada una de las colecciones de datos disponibles.</i>	2,50%	Tareas Cuestionarios	
		f) <i>Se han creado clases y métodos genéricos.</i>	2,50%	Tareas Cuestionarios	
		g) <i>Se han utilizado expresiones regulares en la búsqueda de patrones en ca-</i>	5,00%	Tareas Cuestionarios	UT04

		<i>denas de texto.</i>			
		<i>h) Se han identificado las clases relacionadas con el tratamiento de documentos XML.</i>	0,25%	Tareas Cuestionarios	UT08
		<i>i) Se han realizado programas que realicen manipulaciones sobre documentos XML.</i>	0,25%	Tareas Cuestionarios	
7. Desarrolla programas aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.	12%	<i>a) Se han identificado los conceptos de herencia, superclase y subclase.</i>	0,50%	Tareas Cuestionarios	UT06
		<i>b) Se han utilizado modificadores para bloquear y forzar la herencia de clases y métodos.</i>	1,50%	Tareas Cuestionarios	
		<i>c) Se ha reconocido la incidencia de los constructores en la herencia.</i>	2,00%	Tareas Cuestionarios	
		<i>d) Se han creado clases heredadas que sobrescriban la implementación de métodos de la superclase.</i>	2,00%	Tareas Cuestionarios	
		<i>e) Se han diseñado y aplicado jerarquías de clases.</i>	1,00%	Tareas Cuestionarios	
		<i>f) Se han probado y depurado las jerarquías de clases.</i>	0,25%	Tareas Cuestionarios	
		<i>g) Se han realizado programas que implementen y utilicen jerarquías de clases.</i>	1,75%	Tareas Cuestionarios	
		<i>h) Se ha comentado y documentado el código.</i>	3,00%	Tareas Cuestionarios	UT05/ UT06
		<i>a) Se han identificado las características de las bases de datos orientadas a objetos.</i>	0,10%	Tareas Cuestionarios	UT09
		<i>b) Se ha analizado su aplicación en el desarrollo de aplicaciones mediante lenguajes orientados a objetos.</i>	0,10%	Tareas Cuestionarios	
		<i>c) Se han instalado sistemas gestores de bases de datos orientados a objetos.</i>	0,10%	Tareas Cuestionarios	
		<i>d) Se han clasificado y analizado los distintos métodos soportados por los sistemas gestores para la gestión de la información almacenada.</i>	0,10%	Tareas Cuestionarios	
		<i>e) Se han creado bases de datos y las estructuras necesarias para el almacenamiento de objetos.</i>	0,10%	Tareas Cuestionarios	
		<i>f) Se han programado aplicaciones que almacenen objetos en las bases de</i>	0,10%	Tareas Cuestionarios	

		<i>datos creadas.</i>			
		<i>g) Se han realizado programas para recuperar, actualizar y eliminar objetos de las bases de datos.</i>	0,20%	Tareas	
				Cuestionarios	
		<i>h) Se han realizado programas para almacenar y gestionar tipos de datos estructurados, compuestos y relacionados.</i>	0,20%	Tareas	
				Cuestionarios	
9. Gestiona información almacenada en bases de datos relacionales manteniendo la integridad y consistencia de los datos	2%	<i>a) Se han identificado las características y métodos de acceso a sistemas gestores de bases de datos relacionales.</i>	0,25%	Tareas	UT09
				Cuestionarios	
		<i>b) Se han programado conexiones con bases de datos.</i>	0,25%	Tareas	
				Cuestionarios	
		<i>c) Se ha escrito código para almacenar información en bases de datos.</i>	0,30%	Tareas	
				Cuestionarios	
		<i>d) Se han creado programas para recuperar y mostrar información almacenada en bases de datos.</i>	0,30%	Tareas	
				Cuestionarios	
		<i>e) Se han efectuado borrados y modificaciones sobre la información almacenada.</i>	0,30%	Tareas	
				Cuestionarios	
		<i>f) Se han creado aplicaciones que ejecuten consultas sobre bases de datos.</i>	0,30%	Tareas	
				Cuestionarios	
		<i>g) Se han creado aplicaciones para posibilitar la gestión de información presente en bases de datos relacionales.</i>	0,30%	Tareas	
				Cuestionarios	

La información sobre objetivos, contenidos y criterios de evaluación y calificación será facilitada por el profesorado de cada módulo profesional al alumnado durante el primer trimestre del curso académico.

A efectos de **cómputo de la nota final del módulo**, se tendrán en cuenta las siguientes pautas:

- **Todos los RA (para aquellos CE que han sido evaluados) deben ser superados con un grado de consecución de al menos un 5.**
- **Si no se alcanza la calificación mínima de un 5 en cada RA, la calificación máxima alcanzable al calcular la nota de una evaluación será de 4, pues no se podrá superar el módulo si no se han superado todos sus Resultados de Aprendizaje.**
- Los distintos CE en los que se descompone cada RA pueden ser evaluados con uno o dos instrumentos:
 - El impacto de los **cuestionarios** sobre un CE será de un **20%** de la ponderación de ese CE, mientras que el impacto de las **tareas prácticas** será de un **80%**.
 - En el caso de que algún CE no se evalúe con los dos instrumentos, el impacto del único instrumento que se utilice será del 100%.

- Para el **cálculo de las calificaciones correspondientes al primer, segundo y tercer trimestre**, así como la calificación final, se aplicará lo siguiente:
 - El **grado de consecución de cada RA** se calculará en función de los CE evaluados hasta el momento.
 - La **calificación trimestral** se calculará en función de los CE evaluados hasta el momento. Eso significa que al final de algunos de los trimestres, algunos de los RA habrán sido evaluados solo parcialmente (porque no todos sus CE han sido evaluados aún). **El cálculo del grado de consecución de un RA se llevará a cabo basándonos únicamente en esos CE evaluados.** Ahora bien, si alguno de los RA impartidos hasta el momento (en el grado en el que hayan sido evaluados hasta el momento) **no ha sido superado, la calificación trimestral será como máximo de un 4.**
 - Esto significa que la nota obtenida en cada evaluación es una especie de “fotografía” de cómo se encuentra el alumno o alumna justo en ese momento de curso, reflejando de la manera más fiel posible el concepto de **evaluación continua.**
 - Para el **cálculo de la calificación final** no se realizará la nota media de los trimestres, sino que se seguirá el proceso antes descrito, teniendo en cuenta la calificación obtenida en cada uno de los CE y RA a través de los diferentes instrumentos utilizados durante el curso. Si se han podido impartir y evaluar apropiadamente todos los RA/CE, se tendrán todos en cuenta. Si hay algún RA que no haya sido impartido o alguno que haya sido evaluado parcialmente (no se pudieron evaluar todos sus CE), solo se tendrá en cuenta todo aquello que haya podido ser evaluado. Recordemos una vez más que **cada RA impartido y evaluado (tanto total como parcialmente) debe obtener un grado de consecución mínimo de un 5 para poderse considerar el módulo como superado.** En caso contrario, la máxima calificación alcanzable será de un 4.
- **Redondeo de la calificación final:** cuando el alumnado tenga una nota superior a 5, las calificaciones finales que arrojen números decimales se redondearán a la unidad, eliminando la parte decimal y aproximando la unidad a la más cercana. De este modo, si la parte decimal fuera inferior a 0,500 se aproximará a la unidad inferior. Si esta fuera igual o superior a 0,500, se aproximará a la unidad superior. **Si el cálculo de la calificación final es inferior a 5, el módulo no se considerará como superado y su calificación máxima podría llegar como mucho hasta el 4.**
- La realización de **actividades de ampliación voluntarias** puede conllevar un incremento en la calificación final. En ningún caso, su no realización o mala resolución supondrá un detrimento en la evaluación. Para calcular el incremento de calificación proporcionado por las tareas de ampliación, el profesorado tendrá en cuenta los RA/CE que se evalúen en la actividad propuesta y se actualizarán en aquellos casos en los que se mejoren.

7.4. Criterios de corrección de la expresión escrita

Tal y como se establece en el Proyecto Educativo del centro, los aspectos formales de la expresión escrita serán objeto de valoración por parte de todos los departamentos didácticos en las diferentes pruebas que realice el alumnado. Esto se tendrá en cuenta especialmente en aquellos instrumentos que evalúen los CE:

- *RA1.i) Se han introducido comentarios en el código.*
- *RA3.g) Se ha comentado y documentado el código.*
- *RA5.a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información.*
- *RA5. h) Se han escrito programas que utilicen interfaces gráficos para la entrada y salida de información.*
- *RA7.h) Se ha comentado y documentado el código.*

Para valorar la consecución de estos criterios, se tendrán en cuenta los siguientes aspectos (entre otros):

- *En los programas desarrollados por el alumnado se muestra la información esperada y correcta por pantalla en un formato apropiado.*
- *Se observa una corrección ortográfica y gramatical, así como la coherencia en las expresiones lingüísticas, en los textos de los mensajes que aparezcan en pantalla para pedir información de entrada al usuario o para mostrar resultados de salida.*
- *Se evitan mensajes de entrada de datos y/o salida de resultados inapropiados, descontextualizados, insuficientes o incorrectos.*
- *El código está apropiadamente comentado. Se observa una corrección ortográfica y gramatical, así como la coherencia en las expresiones lingüísticas en los comentarios incluidos.*

Todo esto se valorará hasta en un 30% del impacto del CE sobre la calificación.

7.5. Proceso de recuperación y mejora de calificaciones

Una vez finalizado el periodo de evaluación ordinario, existe un **período final de recuperación o mejora de calificaciones**, comprendido entre la 3ª evaluación y la evaluación final, que se utilizará para intentar que el alumnado consiga alcanzar los resultados de aprendizaje no superados o mejore las competencias relacionadas con el módulo. Durante este periodo:

- El alumnado podrá volver a ser **evaluado de los RA no superados**, a través de los instrumentos de evaluación anteriormente descritos (tareas de carácter práctico y cuestionarios teórico/prácticos). Estos instrumentos de evaluación incorporarán los CE de los RA aún pendientes por superar.
- En el caso de los **cuestionarios** teórico/prácticos, se requerirá que el alumnado realice una o más pruebas donde se le vuelva a evaluar, mediante los CE asociados, aquellos RA que no lograra superar durante el curso.
- En el caso de las **tareas**, se requerirá que el alumnado realice una serie de tareas (diferentes a las realizadas durante el curso, aunque de un nivel similar de dificultad) donde se evalúen los RA no superados mediante los CE asociados.
- De la misma manera, el alumnado que desee **optar a una mejora de su calificación** podrá acogerse también a este proceso preparándose para éste instrumentos de evaluación específicos basados en los CE de los RA en los que se quiera intentar mejorar.

De acuerdo con la [Orden de 29 de septiembre de 2010](#), el carácter de la evaluación será continua, por tanto, **la asistencia del alumnado durante todo este periodo de recuperación es obligatoria para poder recuperar o mejorar la calificación**. Las faltas de asistencia durante este período por parte del alumnado que necesite recuperar dará lugar a la no superación del módulo, pues no se trata de venir un día a realizar un examen de recuperación, sino de asistir durante este período a una serie de sesiones en las que se llevarán a cabo diversas actividades de recuperación.

La **nota final después de este periodo de recuperación** se obtendrá teniendo en cuenta todo el trabajo realizado durante el curso, pero considerando las calificaciones de las partes recuperadas o mejoradas en lugar de las calificaciones obtenidas y registradas inicialmente. Por tanto, las pautas para calcular la calificación en el periodo de recuperación o mejora de nota, y poder considerar el módulo por superado o mejorado, a través de sus RA y CE, son las mismas que se han descrito para el resto del curso.

8. Orientaciones pedagógicas

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 bibliotecas** 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.

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.

9. Fomento de la Lectura.

Para el desarrollo de la **competencia en comunicación lingüística** de los centros educativos públicos se desarrollarán las siguientes actuaciones:

- **Lectura en clase** de artículos relacionados con la materia del módulo al final de las unidades de trabajo, dichas lecturas se complementarán con coloquios o debates sobre la temática de la lectura, que también podrá ser apoyada por otro material multimedia.
- Realización de trabajos de investigación comparando diversos artículos y contenidos, redactando finalmente **resúmenes y esquemas** con la información obtenida.
- Realización de **presentaciones orales** sobre algunas de las tareas realizadas fomentando así la lectura desde la oralidad, con lo que se potencia y enriquece la capacidad de expresión del alumnado y se desarrolla su capacidad de atención y comunicación.

10. Medidas previstas para la consecución de la plena igualdad entre hombres y mujeres.

Teniendo en cuenta que habitualmente la presencia de alumnas en este ciclo es mínima, no es tarea fácil llevar a cabo este punto, pues partimos de una importante desproporción de género desde la misma matriculación. No obstante, se trabaja ponderando la labor y la figura de la mujer en la sociedad como elemento indispensable para la convivencia, la mejora y la integración en la sociedad.

Las medidas que se tomarán serán las siguientes:

- **Visibilizar el papel de la mujer:** se destacarán las figuras femeninas que han contribuido en el desarrollo de la materia, en nuestro caso en Informática, y en particular en Programación.
- Utilizar el **lenguaje igualitario, inclusivo, y no sexista:** se procurará utilizar la lengua tanto oral como escrita para identificar una realidad que no excluya a las mujeres.
- **Evitar estereotipos:** se evitarán los estereotipos asociados al sexo en los enunciados de prácticas, ejercicios, tareas, etc.
- Participación en las actividades del **plan de Igualdad:** Se instará a participar en talleres, charlas, presentaciones, etc., que se realicen en el centro para la consecución de la plena igualdad entre hombres y mujeres.

11. Medidas de Atención a la Diversidad.

La atención a la diversidad hace referencia a las **adaptaciones curriculares no significativas**. Una adaptación curricular no significativa establece medidas de refuerzo o de ampliación para determinados alumnos y alumnas sin cambiar los objetivos y contenidos establecidos por el módulo. Esto habitualmente se debe a que el alumnado parte con distintos niveles de conocimientos y experiencia dependiendo de su procedencia (otros ciclos formativos de grado medio o superior de la misma u otras familias profesionales, bachilleratos tecnológicos, bachilleratos de ciencias sociales, estudios universitarios, etc.).

Por otro lado, también habrá que tener en cuenta que no todos los alumnos y alumnas superarán los objetivos establecidos con la misma facilidad pudiendo necesitar en algunos casos dichos refuerzos o ampliaciones, según los supere con dificultad o por el contrario los supere con notable facilidad y estén preparados para poder aprender algo más.

Dado que es probable que los niveles sean diferentes podemos disponer de varios recursos que se pueden emplear para atender esta diversidad, pudiéndose plantear un seguimiento individual de cada alumno o alumna a través de propuestas del tipo:

- Realización de las actividades propuestas al final de cada unidad, que siguen un orden creciente de dificultad.
- Integración del alumnado con problemas en grupos de trabajo de nivel heterogéneo para que en ningún momento se sienta discriminado. Si se crea un buen ambiente de grupo, los mismos compañeros y compañeras se ayudarán entre ellos favoreciendo el proceso de aprendizaje.
- Apoyo del profesorado cuando lo consideren necesario y en la forma que se estime oportuna.
- A través de la lectura del material complementario (libros, apuntes, ejercicios resueltos, revistas, artículos, etc.) que se encuentra en el aula.
- Realización de actividades complementarias propuestas y/o coordinadas por los profesores.
- Realizaciones de trabajos haciendo uso de la capacidad creativa y los medios y recursos con que cuenta el centro.
- Planteamiento por parte de los profesores/as de ejercicios y cuestionarios al alumnado con la consiguiente supervisión.
- Exposición de algunos de los trabajos que se van desarrollando en clase.
- Adaptación de la programación, delimitando aquellos que sean considerados como mínimo exigible según el currículo.

11.1. Tratamiento del alumnado con NEE

En relación al **tratamiento del alumnado con NEE**, se estará a lo dispuesto en los **acuerdos adoptados en la sesión de evaluación inicial** para cada caso, reflejados en el acta de la reunión del equipo educativo, o si se detectan a posteriori, en una **reunión del equipo educativo convocada al efecto**.

En caso de que se detectara algún caso de NEE tanto en la evaluación inicial como durante el resto del curso, el tutor del grupo, en colaboración con el equipo educativo y el departamento de orientación, elaborará un informe sobre las medidas a adoptar.

12. Actividades complementarias y extraescolares.

Las posibles actividades propuestas para este curso son:

- Aquellas actividades relacionadas con el **desarrollo de software** que se realicen durante las tradicionales **jornadas "Orientate"**.
- Aquellas actividades relacionadas con el **desarrollo de software** que se realicen durante la **"Feria de Orientación Académica y Profesional"**.
- Participación en **talleres y concursos** relacionados con el **desarrollo de software** que se convoquen a lo largo del curso por diversos organismos públicos o privados. Las convocatorias las pueden realizar instituciones como la Universidad de Almería, el INCIBE (Instituto Nacional para la Ciberseguridad), etc.

13. Recursos Didácticos.

Para el desarrollo normal de la clase se disponen de los siguientes recursos:

- Pizarra y accesorios.
- Ordenador de sobremesa del profesorado y de dotación TDE.
- Proyector para conectar a ordenador o recursos audiovisuales.
- Sistema de altavoces.
- Libros de consulta (no están el aula, pero en un momento dado se pueden consultar del departamento).
- Curso alojado en la plataforma Moodle Centros.
- Aula con ordenadores para alumnado.
- Conexión a Internet.
- Software de red, software de documentación (paquete ofimático), navegadores web y utilidades para el acceso a los servicios de red.
- JDK Java y entorno de desarrollo integrado NetBeans.

14. Bibliografía.

- Material del módulo de Programación de la modalidad de FP a Distancia elaborado por la Junta de Andalucía/Ministerio de Educación.
- Ceballos, F.J. Java 2 (2005). Curso de programación, 3ª edición, Madrid, Ra-Ma.
- Holzner, Steven (2000). La Biblia de Java 2, Madrid, Anaya Multimedia.
- Eckel, Bruce (2003). Piensa en Java, Madrid, Prentice-Hall.
- Froufe Quintas, Agustín y Jorge Cárdenas, Patricia (2003). J2ME: Java 2 Micro Edition: manual de usuario y tutorial, Madrid, Ra-Ma.
- Aprenda Java como si estuviera en Primero. Universidad de Deusto.
- Deitel, H.M., and Deitel P.J.: (1998). Cómo programar en Java, Madrid, Prentice-Hall.