

PROGRAMACIÓN DIDÁCTICA

Módulo: Programación

Curso: 1º

Ciclo Formativo de Grado Superior

Desarrollo de Aplicaciones Multiplataforma

Autor: Víctor de Juan

Especialidad: Informática

Firma: _____ 19 de marzo de 2025

Índice general

1. Introducción	4
1. Resumen de conceptos	7
2. Contextualización	9
1. Contexto del módulo dentro del ciclo	9
1.1. Competencias personales y para la empleabilidad a adquirir con el módulo	9
2. Contexto	12
3. Resultados de aprendizaje y criterios de evaluación	15
1. Principios de la evaluación	22
4. Saberes básicos y temporalización	24
1. Saberes básicos	24
2. Temporalización	26
5. Desarrollo de las Unidades Didácticas	27
1. Unidades Didácticas	27
1.1. UD1: Introducción a la Programación	27
1.2. UD2: Básicos de programación estructurada	28
1.3. UD3: Estructuras de control	30
1.4. UD4: Programación modular	31
1.5. UD5: Estructuras de almacenamiento en C	32
1.6. UD6: Cambio de paradigma: La Orientación a Objetos	33
1.7. UD7: Colecciones de datos	35
1.8. UD8: POO Intermedia	37
1.9. UD9: POO Avanzada	39
1.10. UD10:Control y manejo de excepciones:	41
1.11. UD11: Lectura y escritura de información	42

1.12.	UD12: Gestión de bases de datos relacionales	43
1.13.	UD13: Mantenimiento de la persistencia de los objetos:	44
6.	Metodología didáctica	45
1.	Principios metodológicos	45
2.	Recursos	46
3.	Medidas de atención a la diversidad para alumnos con necesidad es- pecífica de apoyo educativo	47
7.	Evaluación y calificación	48
1.	Características de la evaluación	48
2.	Procedimientos de evaluación y criterios de calificación	49
3.	Proceso de evaluación continua y calificación en la evaluación final or- dinaria	50
4.	Proceso de evaluación para alumnos a los que no se puede aplicar la evaluación continua (pérdida del derecho a la evaluación continua) . . .	50
5.	Proceso de evaluación y calificación en la evaluación final extraordinaria	50
6.	Medidas para alumnos con necesidad específica de apoyo educativo (2 ptos)	50
7.	Procedimiento de evaluación para alumnos con el módulo pendiente . .	50
7.1.	Ordinaria	50
7.2.	Extraordinaria	50
8.	Calendario de evaluaciones parciales, final ordinaria y final extraordinaria	50

Para el 18-03: Revisar legislación en total

Para el 18-03: Revisar legislación que citar en cada apartado. Ver magister

1 Introducción

La presente programación didáctica corresponde al módulo profesional de Programación, perteneciente al primer curso del Ciclo Formativo de Grado Superior (CFGS) en Desarrollo de Aplicaciones Multiplataforma (DAM).

El módulo de Programación es fundamental dentro del ciclo formativo, ya que establece las bases conceptuales y procedimentales necesarias para desarrollar aplicaciones multiplataforma.

En cuanto a su planteamiento didáctico, se parte de la premisa de que todo el alumnado comienza desde cero, independientemente de los conocimientos previos que pudiera tener. Es posible que algunos estudiantes lleguen al ciclo con cierta experiencia en programación adquirida de forma autodidacta o en estudios anteriores, pero en muchos casos estos aprendizajes responden a enfoques parciales, carentes de rigor o orientados únicamente a que funcione como sea. En este módulo se persigue un conocimiento sólido, bien estructurado y con fundamentos teóricos y prácticos robustos. Nuestro objetivo es que el alumnado adquiera no solo la capacidad de programar, sino de hacerlo con criterio profesional, buenas prácticas y capacidad de análisis y resolución de problemas, sentando así las bases para el resto de módulos del ciclo.

La metodología empleada busca atender la diversidad del grupo y alinearse con las oportunidades y limitaciones del contexto socioeconómico local, destacando un enfoque activo y colaborativo del aprendizaje, con énfasis en el desarrollo de proyectos reales y la resolución práctica de problemas mediante el uso de herramientas tecnológicas profesionales.

Esta programación didáctica se elabora como un documento técnico-pedagógico que establece los objetivos, contenidos, metodología, criterios e instrumentos de evaluación, y medidas de atención a la diversidad que se aplicarán a lo largo del curso. Su finalidad es servir como guía estructurada para garantizar una enseñanza coherente, rigurosa y adaptada a las necesidades reales del alumnado y del entorno. Asimismo,

este documento pretende ser un referente tanto para el profesorado como para el propio equipo docente del ciclo, favoreciendo la coordinación, la reflexión compartida y la mejora continua del proceso de enseñanza-aprendizaje.

Para garantizar la coherencia con el marco institucional y normativo del centro, esta programación didáctica se elabora teniendo en cuenta diversos documentos clave que rigen la organización, funcionamiento y orientación educativa del instituto. Entre ellos se incluyen:

- El Proyecto Educativo del Centro (PEC), que define los principios, valores y objetivos educativos generales.
- La Programación General Anual (PGA), donde se concretan los objetivos operativos y las actuaciones planificadas para el curso académico.
- El Plan de Acción Tutorial (PAT), especialmente relevante para la coordinación con los tutores en lo relativo al seguimiento académico y personal del alumnado.
- El Plan de Atención a la Diversidad (PAD), que establece las directrices para la inclusión y atención específica al alumnado con necesidades educativas especiales o específicas.
- El Plan de Convivencia, que orienta la gestión de la convivencia escolar y el desarrollo de un clima positivo en el aula.
- El Plan Digital de Centro, que proporciona el marco para la integración de las tecnologías digitales en la práctica docente.
- Los currículos oficiales de los ciclos formativos, aprobados por la Comunidad de Madrid, que definen los resultados de aprendizaje, criterios de evaluación y contenidos del módulo. Todos estos documentos sirven de referencia para asegurar que la práctica docente en este módulo se alinea con las directrices generales del centro y con la normativa educativa vigente.

A continuación, destacamos el marco normativo específico, tanto autonómico como estatal que vertebra esta programación didáctica.

Marco normativo vigente

Estatal

- **Ley Orgánica 3/2020**, de 29 de diciembre, por la que se modifica la **Ley Orgánica 2/2006** de Educación. Establece el marco educativo actual, orientado a un enfoque competencial y por resultados de aprendizaje.
- **Ley Orgánica 3/2022** Ley Orgánica 3/2022, de 31 de marzo, de Ordenación e Integración de la Formación Profesional.
- **Real Decreto 659/2023** De 18 de julio, por el que se desarrolla la ordenación del Sistema de Formación Profesional.
- **Real Decreto 500/2024** Norma reciente que actualiza el marco curricular y organizativo de la Formación Profesional a nivel estatal, en línea con los principios de la LOMLOE y la nueva ley de FP.
- **Real Decreto 450/2010** de 16 de abril, establece el título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma y **Real Decreto 405/2023**, de 29 de mayo, actualiza dicho título y fija las enseñanzas mínimas del ciclo.
 - El [Real Decreto 405/2023](#), en su anexo, establece el currículo básico del módulo 0485 Programación, con los resultados de aprendizaje, criterios de evaluación y contenidos mínimos que se deben cumplir a nivel estatal. Esta programación se ajusta íntegramente a dichos mínimos curriculares.
 - El [Real Decreto 450/2010](#) mantiene el perfil profesional, competencia general, entorno profesional, listado de módulos profesionales, espacio y equipamientos, profesorado, accesos y vinculación a otros estudios, y correspondencia de módulos profesionales con las unidades de competencia.

Autonómica

Para todas las enseñanzas de formación profesional:

- **Decreto 63/2019** Decreto 63/2019, de 16 de julio, por el que se regula la ordenación y organización de la formación profesional de la Comunidad de Madrid (pendiente de publicación de nuevo decreto que lo derogará).
- **Orden 893/2022** (modificada por **Orden 3413/2022**, de 21 de abril, por la que se regulan los procedimientos relacionados con la organización, la matrícula, la evaluación y acreditación académica de las enseñanzas de formación profesional del sistema educativo en la Comunidad de Madrid.

Modificada por Orden de 15 de noviembre de 2022 (pendiente de publicación de nueva orden que la derogará).

- **Resolución de 4 de junio de 2024**, de la Dirección General de Educación Secundaria, Formación Profesional y Régimen Especial, por la que se dictan instrucciones sobre la ordenación y la organización de los grados D y E de formación profesional en el curso académico 2024-2025.
- **Decreto 103/2024**, de 13 de noviembre, del Consejo de Gobierno, por el que se modifican setenta y seis decretos por los que se establecen para la Comunidad de Madrid planes de estudios de ciclos formativos de grado superior.
- **Decreto 3/2011**, de 13 de enero (BOCM 31/01/2011), que establece el currículo del ciclo DAM en Madrid, basado en el anterior [Real Decreto 450/2010](#). Aunque está en proceso de actualización según la nueva ley de FP, sigue vigente como referencia para la organización del ciclo (distribución horaria, módulos profesionales, etc.). Este decreto enfatiza que el currículo oficial debe concretarse en programaciones didácticas adaptadas a cada centro, permitiendo adecuaciones particulares en función de los recursos, sin suprimir objetivos ni la competencia general del título.

1.1 Resumen de conceptos

A continuación, destacamos una breve definición de algunos conceptos fundamentales que servirán para concretar el trabajo que debe realizarse a lo largo del curso.

Estos son los elementos que encontramos a los largo de la legislación, organizados de más general a más concreto.

- **Competencia general del título:** Describe las funciones profesionales del perfil profesional del título.
- **Competencias personales y para la empleabilidad:** Describen el conjunto de conocimientos y destrezas que permiten responder a los requerimientos del sector productivo del título.
- **Objetivos:** Son los logros que el estudiante alcanzará al finalizar el ciclo, como resultado de las experiencias de enseñanza-aprendizaje intencionalmente planificadas.
- **Estándares de competencia:** constituyen la unidad básica para el diseño de la formación y para la acreditación de competencias profesionales adquiridas por experiencia laboral u otras vías no formales o informales.
- **Resultados de aprendizaje:** Son lo que se espera que un estudiante sea capaz de hacer una vez finalizado el proceso de aprendizaje. Son como los objetivos a alcanzar en cada módulo del ciclo. Son el eje vertebrador de la programación.
- **Criterios de Evaluación:** Describen los aspectos, en extensión, de los resultados de aprendizaje. Permiten conocer si se han alcanzado y con qué nivel de logro los resultados de aprendizaje.
- **Saberes Básicos:** Son los conocimientos (conceptos, procedimientos, habilidades, destrezas y actitudes) asociados al desempeño profesional que, mediante las actividades de enseñanza-aprendizaje, permiten alcanzar los resultados de aprendizaje.
- **Metodología Didáctica:** el conjunto de estrategias, procedimientos y acciones organizadas y planificadas por el profesorado, con la finalidad de posibilitar el aprendizaje del alumnado y el logro de los resultados de aprendizaje.

2 Contextualización

2.1 Contexto del módulo dentro del ciclo

Referencia al Real Decreto de enseñanzas mínimas: El módulo Programación tiene una duración mínima de 135 horas según el [Real Decreto 405/2023](#), lo que en el currículo de Madrid ([Decreto 3/2011](#)) se concreta en 270 horas distribuidas en el primer curso (aprox. 8 horas semanales). Esta distribución horaria y curricular es la base para la temporalización y secuenciación de unidades didácticas que se detalla más adelante (ver 4). Simplemente, aquí destacamos que se incluye un periodo de Formación en Centros de Trabajo (FCT) entre la segunda y la tercera evaluación.

2.1.1.1 Competencias personales y para la empleabilidad a adquirir con el módulo

El [Real Decreto 450/2010](#) establece las competencias que deben ser alcanzadas al completar el Ciclo Formativo y, por otro lado, detalla cuáles son las competencias que trabaja cada uno de los módulos que componen el ciclo. El decreto [Real Decreto 405/2023](#) que actualiza el [Real Decreto 450/2010](#) no modifica estas competencias.

Competencias del ciclo

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 móviles, tabletas y otros dispositivos inteligentes 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 vidas 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.

Competencias del módulo:

A continuación listamos las competencias que, de acuerdo con el [Real Decreto 450/2010](#), se deben obtener en la consecución del módulo que nos compete:

- e) Desarrollar aplicaciones multiplataforma con acceso a bases de datos utilizando lenguajes, librerías y herramientas adecuados a las especificaciones.
- j) Desarrollar aplicaciones para teléfonos móviles, tabletas y otros dispositivos inteligentes empleando técnicas y entornos de desarrollo específicos.
- t) Establecer vidas eficaces de relación profesional y comunicación con sus superiores, compañeros y subordinados, respetando la autonomía y competencias de las distintas personas.
- 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.

2.2 Contexto

Se imparte en un instituto público situado en el sureste de Madrid, en un distrito urbano que combina áreas residenciales consolidadas con zonas en expansión y crecimiento demográfico. El tejido socioeconómico local está compuesto principalmente por pequeños comercios familiares, servicios comunitarios, centros educativos, entidades sociales y culturales, así como establecimientos de hostelería y restauración con presencia creciente debido a la ampliación reciente de zonas residenciales, con una limitada presencia de pequeñas empresas tecnológicas.

El centro educativo dispone de instalaciones adecuadas y equipadas tecnológicamente, lo que favorece un entorno propicio para el desarrollo de competencias prácticas y teóricas relacionadas con la informática y las comunicaciones. Además, la ubicación del instituto facilita ciertas oportunidades de colaboración con empresas locales para realizar prácticas formativas, aunque dichas oportunidades son limitadas

debido a la ausencia de grandes parques empresariales o grandes centros tecnológicos en la zona. No obstante, la comunicación en transporte público con centros tecnológicos situados en el norte de Madrid es muy buena, lo que permite contar con empresas excepcionales para la realización de la Formación en Centros de Trabajo (de aquí en adelante, FCT).

Se trata de un centro grande, que imparte enseñanzas de Educación Secundaria Obligatoria (ESO), Bachillerato, y Ciclos Formativos de Grado Básico, Medio y Superior. Además de todos los ciclos de la familia profesional de Informática y Comunicaciones, el instituto cuenta con ciclos de otras dos familias profesionales: Imagen Personal (estética) y Transporte y Mantenimiento de Vehículos (automoción). Esta diversidad formativa favorece el intercambio interdisciplinar, el enriquecimiento de la vida académica y la participación activa del alumnado en proyectos colaborativos que trascienden las fronteras de cada especialidad.

El proyecto educativo del centro está firmemente comprometido con la inclusión, la innovación metodológica y la orientación profesional del alumnado. A través de un enfoque centrado en la adquisición de competencias, el centro promueve el uso responsable de la tecnología, la colaboración interdisciplinar y el pensamiento crítico, con el objetivo de formar profesionales técnicos altamente capacitados y socialmente comprometidos.

El grupo destinatario de esta programación didáctica está formado por 29 estudiantes con edades comprendidas mayoritariamente entre los 18 y 22 años, con algún alumno rondando los 30 años. Este grupo destaca por la heterogeneidad de su alumnado, incluyendo estudiantes procedentes tanto del Bachillerato como del ámbito laboral o de otros ciclos formativos, fundamentalmente de Sistemas Microinformáticos y Redes (SMR). Dentro del grupo, se han identificado necesidades educativas especiales (NEE) en tres estudiantes: uno de ellos presenta dislexia, lo que afecta a su velocidad y precisión lectora, así como a la comprensión de textos escritos; otro presenta un Trastorno por Déficit de Atención (TDA) sin hiperactividad, que se manifiesta en dificultades para mantener la concentración de forma sostenida; y el tercero mues-

tra rasgos compatibles con un Trastorno del Espectro Autista (TEA) de nivel 1, con especial sensibilidad a los cambios de rutina y dificultades en la interacción social espontánea, lo que implica la adopción de estrategias metodológicas flexibles, atención individualizada y medidas específicas de adaptación curricular, asegurando así una adecuada inclusión educativa y el éxito académico de todos los estudiantes. En este sentido, se aplicarán los principios del Diseño Universal para el Aprendizaje (DUA), que permiten flexibilizar los métodos, materiales y formas de participación y evaluación, facilitando así el acceso equitativo a los contenidos del módulo para todo el alumnado.

3 Resultados de aprendizaje y criterios de evaluación

A continuación, presentamos el elemento vertebrador de la práctica docente de esta programación: los Resultados de Aprendizaje (RAs) definidos en el [Real Decreto 450/2010](#). Estos son los aprendizajes que debe adquirir un alumno al terminar el módulo.

Los RAs han sido diseñados para que sean una guía práctica para desarrollar progresivamente las competencias del módulo, actuando como objetivos concretos y evaluables en la formación del alumnado. Cada competencia queda cubierta y desarrollada a través de uno o varios resultados de aprendizaje, garantizando así una adquisición integral y sistemática de los conocimientos y habilidades definidos en el módulo de Programación.

La [Ley Orgánica 3/2022](#) establece que debe trabajarse parte del Currículo en las FCTs, otorgando libertad al centro y al departamento para la concreción de cuáles y en qué porcentaje. No obstante, sí existe la recomendación por parte de la Consejería de no dejar un RA completo para las FCTs. Además, el centro ha establecido el criterio de que no más del 10 % del peso de un RA puede corresponder a la FCT.

En este caso, solamente un RA (el 3) será parcialmente calificado en la FCTs. En el apartado de calificación se concretarán los instrumentos y ponderaciones exactos.

En la siguiente tabla se presentan los RAs indicando, para cada uno, tanto su ponderación respecto de la calificación total como la ponderación de cada Criterios de Evaluación (CE) dentro del RA. La suma de las ponderaciones de los CEs suma 100 % para cada RA, así como la ponderación de los RAs suma 100 %.

RA	Criterio de evaluación	%
RA1 (10 %): Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.	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 (20 %): Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.	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.	

RA	Criterio de evaluación	%
RA3 (30 %): Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje. <i>Este es el RA que será parcialmente evaluado en las FCTs</i>	a) Se ha escrito y probado código que haga uso de estructuras de selección.	5 %
	b) Se han utilizado estructuras de repetición.	5 %
	c) Se han reconocido las posibilidades de las sentencias de salto.	5 %
	d) Se ha escrito código utilizando control de excepciones.	
	e) Se han creado programas ejecutables utilizando diferentes estructuras de control.	25 %
	f) Se han probado y depurado los programas.	25 %
	g) Se ha comentado y documentado el código.	25 %
	h) Se han creado excepciones.	
	i) Se han utilizado aserciones para la detección y corrección de errores durante la fase de desarrollo.	
RA4 (30 %): Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.	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 creado constructores.	
	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 creado y utilizado conjuntos y librerías de clases.	

RA	Criterio de evaluación	%
RA5 (30 %): Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases.	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 utilizado las herramientas del entorno de desarrollo para crear interfaces gráficos de usuario simples.	
	g) Se han programado controladores de eventos.	
	h) Se han escrito programas que utilicen interfaces gráficos para la entrada y salida de información.	
RA6 (30 %): Escribe programas que manipulen información seleccionando y utilizando tipos avanzados de datos.	a) Se han escrito programas que utilicen matrices (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.	

RA	Criterio de evaluación	%
	h) Se han identificado las clases relacionadas con el tratamiento de documentos escritos en diferentes lenguajes de intercambio de datos.	
	i) Se han realizado programas que realicen manipulaciones sobre documentos escritos en diferentes lenguajes de intercambio de datos.	
	j) Se han utilizado operaciones agregadas para el manejo de información almacenada en colecciones.	
RA7 (30 %): Desarrolla programas aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.	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 sobrescriben 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.	
	i) Se han identificado y evaluado los escenarios de uso de interfaces.	
	j) Se han identificado y evaluado los escenarios de utilización de la herencia y la composición.	

RA	Criterio de evaluación	%
RA8 (30 %): Utiliza bases de datos orientadas a objetos, analizando sus características y aplicando técnicas para mantener la persistencia de la información.	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 (30 %): Gestiona información almacenada en bases de datos 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.	
	b) Se han programado conexiones con bases de datos.	
	c) Se ha escrito un 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.	

RA	Criterio de evaluación	%
	f) Se han creado aplicaciones que muestren la información almacenada en bases de datos.	
	g) Se han creado aplicaciones para gestionar la información presente en bases de datos.	

3.1 Principios de la evaluación

Para considerar que el alumno ha superado el módulo deberá tener conseguidos todos los resultados de aprendizaje. Cada resultado de aprendizaje será calificado con una nota de 0 a 10, como se detallará más adelante. Aquí solamente nos centramos en los principios que rigen la evaluación de estos resultados de aprendizaje sin entrar en cómo serán exactamente calificados.

El carácter competencial de la [Ley Orgánica 3/2020](#) y la [Ley Orgánica 3/2022](#) anima a dar la oportunidad al alumno a seguir mejorando y demostrar que ha adquirido los resultados de aprendizaje. Por este motivo, la evaluación será continua a lo largo del curso.

Además, la evaluación del alumnado tendrá un carácter criterial, ateniéndose siempre a los criterios de evaluación vinculados a cada resultado de aprendizaje del módulo. Cabe mencionar que no será necesario tener una calificación aprobada en cada criterio de evaluación de un resultado de aprendizaje para que dicho resultado de aprendizaje pueda considerarse adquirido gracias a todos los demás criterios de evaluación.

Por otro lado, la evaluación tendrá un carácter formativo de cara a que el alumno pueda aprender de sus errores y seguir perfeccionando los puntos fuertes sobre los que se sostiene su aprendizaje.

Por último, la evaluación estará caracterizada por los principios del Diseño Universal del Aprendizaje (DUA) atendiendo a la diversidad del alumnado con diferentes formas de evaluación que les permitan demostrar los aprendizajes. En esta línea, merece la pena destacar que se pueden realizar evaluaciones no calificables mediante la retroalimentación oral en las sesiones de clase.

Estos principios generales serán concretados más adelante (ver sección ??) pero era necesario declararlos aquí puesto que los Resultados de Aprendizaje con sus Criterios de Evaluación son el elemento vertebrador de esta Programación Didáctica. A continuación, trataremos de los contenidos y saberes básicos que serán los que se utilizarán para aplicar los criterios de evaluación y serán el medio para la consecución

de los resultados de aprendizaje.

4 Saberes básicos y temporalización

4.1 Saberes básicos

A continuación, se presentan los saberes básicos definidos en [Decreto 3/2011](#) (concreción autonómica del [Real Decreto 450/2010](#) y de [Real Decreto 405/2023](#)).

Estos saberes básicos serán el medio con el que conseguir los aprendizajes concretos definidos por los Resultados de Aprendizaje que, a su vez, son los que garantizan la consecución de las competencias de módulo permitiendo, de esta manera, conseguir adquirir todas las competencias de ciclo.

Estos son los contenidos que se tratarán en cada una de las Unidades Didácticas.

- *Introducción a la programación:* Datos, algoritmos y programas. Paradigmas de programación. Lenguajes de programación. Herramientas y entornos para el desarrollo de programas. Errores y calidad de los programas.
- *Introducción a la orientación a objetos:* Clases. Atributos, métodos y visibilidad. Objetos. Estado, comportamiento e identidad. Mensajes. Encapsulado. Visibilidad. Relaciones entre clases. Principios básicos de la orientación a objetos.
- *Identificación de los elementos de un programa informático:* Estructura y bloques fundamentales. Identificadores. Palabras reservadas. Variables. Declaración, inicialización y utilización. Almacenamiento en memoria. Tipos de datos. Literales. Constantes. Operadores y expresiones. Precedencia de operadores. Conversiones de tipo. Implícitas y explícitas (casting). Comentarios.
- *Utilización de objetos:* Características de los objetos. Constructores. Instanciación de objetos. Declaración y creación. Utilización de métodos. Parámetros y valores de retorno. Utilización de propiedades. Utilización de métodos estáticos. Almacenamiento en memoria. Tipos básicos vs objetos. Destrucción de objetos y liberación de memoria.
- *Uso de estructuras de control:* Estructuras de selección. Estructuras de repetición. Estructuras de salto.

- *Desarrollo de clases:* Concepto de clase. Estructura y miembros de una clase. Creación de atributos. Declaración e inicialización. Creación de métodos. Declaración, argumentos y valores de retorno. Creación de constructores. Ámbito de atributos y variables. Sobrecarga de métodos. Visibilidad. Modificadores de clase, de atributos y de métodos. Paso de parámetros. Paso por valor y paso por referencia. Utilización de clases y objetos. Utilización de clases heredadas. Librerías y paquetes de clases. Utilización y creación. Documentación sobre librerías y paquetes de clases.
- *Aplicación de las estructuras de almacenamiento:* Estructuras. Arrays unidimensionales y multidimensionales: Declaración. Creación de arrays unidimensionales y multidimensionales. Inicialización Acceso a elementos. Recorridos, búsquedas y ordenaciones. Cadenas de caracteres: Declaración. Creación de cadenas de caracteres. Inicialización Operaciones. Acceso a elementos, conversiones, concatenación.
- *Utilización avanzada de clases:* Relaciones entre clases. Composición de clases. Herencia. Concepto y tipos (simple y múltiple). Superclases y subclases. Constructores y herencia. Modificadores en clases, atributos y métodos. Sobreescritura de métodos. Clases y métodos abstractos y finales. Interfaces. Clases abstractas vs. Interfaces. Polimorfismo: Concepto. Polimorfismo en tiempo de compilación (sobrecarga) y polimorfismo en tiempo de ejecución (ligadura dinámica). Comprobación estática y dinámica de tipos. Conversiones de tipos entre objetos (casting). Clases y tipos genéricos o parametrizados.
- *Control y manejo de excepciones:* Excepciones. Concepto. Jerarquías de excepciones. Manejo de excepciones: Captura de excepciones. Propagar excepciones. Lanzar excepciones. Crear clases de excepciones.
- *Colecciones de datos:* Tipos de colecciones (listas, pilas, colas, tablas). Jerarquías de colecciones. Operaciones con colecciones. Acceso a elementos y recorridos. Uso de clases y métodos genéricos.

- *Lectura y escritura de información:* Flujos (streams): Tipos de flujos. Flujos de bytes y de caracteres. Clases relativas a flujos. Jerarquías de clases. Utilización de flujos. Entrada/salida estándar: Entrada desde teclado. Salida a pantalla. Almacenamiento de información en ficheros: Ficheros de datos. Registros. Apertura y cierre de ficheros. Modos de acceso. Escritura y lectura de información en ficheros. Almacenamiento de objetos en ficheros. Persistencia. Serialización. Utilización de los sistemas de ficheros. Creación y eliminación de ficheros y directorios. Interfaces gráficas de usuario simples. Concepto de evento. Creación de controladores de eventos.
- *Gestión de bases de datos relacionales:* Interfaces de programación de acceso a bases de datos. Establecimiento de conexiones. Recuperación de información. Manipulación de la información. Ejecución de consultas sobre la base de datos.
- *Mantenimiento de la persistencia de los objetos:* Bases de datos orientadas a objetos. Características de las bases de datos orientadas a objetos. Instalación del gestor de bases de datos. Creación de bases de datos. Mecanismos de consulta. El lenguaje de consultas: sintaxis, expresiones, operadores. Recuperación, modificación y borrado de información. Tipos de datos objeto; atributos y métodos. Tipos de datos colección.

4.2 Temporalización

Relación (tabla) de unidades didácticas, situadas en el anexo. Aquí solo títulos y RAs con sus CEs

5 Desarrollo de las Unidades Didácticas

5.1 Unidades Didácticas

5.1.1.1 UD1: Introducción a la Programación

RAAs y CEs:

- RA1.a) Se han identificado los bloques que componen la estructura de un programa informático.

Saberes Básicos: *Introducción a la programación:* Datos, algoritmos y programas. Paradigmas de programación. Lenguajes de programación. Herramientas y entornos para el desarrollo de programas. Errores y calidad de los programas.

Sesiones: completar

Instrumento de calificación: test moodle con aplicaciones prácticas de esta introducción

5.1.1.2 UD2: Básicos de programación estructurada

RAAs y CEs

- RA1.b) Se han creado proyectos de desarrollo de aplicaciones.
- RA1.c) Se han utilizado entornos integrados de desarrollo.
- RA1.d) Se han identificado los distintos tipos de variables y la utilidad específica de cada uno.
- RA1.e) Se ha modificado el código de un programa para crear y utilizar variables.
- RA1.f) Se han creado y utilizado constantes y literales.
- RA1.g) Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje.
- RA1.h) Se ha comprobado el funcionamiento de las conversiones de tipo explícitas e implícitas.
- RA1.i) Se han introducido comentarios en el código.
- RA5.a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información.
- RA5.b) Se han aplicado formatos en la visualización de la información.
- RA5.c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.

Saberes básicos: *Identificación de los elementos de un programa informático:* Estructura y bloques fundamentales. Identificadores. Palabras reservadas. Variables. Declaración, inicialización y utilización. Almacenamiento en memoria. Tipos de datos. Literales. Constantes. Operadores y expresiones. Precedencia de operadores Conversiones de tipo. Implícitas y explícitas (casting). Comentarios.

Entrada Salida

Sesiones: completar

Instrumentos de calificación: Ejercicios a resolver en clase, entrega de prácticas individuales, tipo test de moodle para identificar errores

5.1.1.3 UD3: Estructuras de control

RAs y CEs:

- RA3.a) Se ha escrito y probado código que haga uso de estructuras de selección.
- RA3.b) Se han utilizado estructuras de repetición.
- RA3.c) Se han reconocido las posibilidades de las sentencias de salto.
- RA3.e) Se han creado programas ejecutables utilizando diferentes estructuras de control.
- RA3.f) Se han probado y depurado los programas.
- 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.b) Se han aplicado formatos en la visualización de la información.
- RA5.c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.

Contenidos: Estructuras de selección. Estructuras de repetición. Estructuras de salto.

Sesiones: completar

Instrumentos de calificación: Ejercicios de clase, entregas individuales de prácticas (una de selección, otra de repetición y otra combinada), tipo test para reconocer bucles y aprender a pensar [no probar si es un < o un <=], vídeo explicativo de cómo depurar con gdb

5.1.1.4 UD4: Programación modular

RAs y CEs:

- RA1.f) Se han creado y utilizado constantes y literales.
- RA1.i) Se han introducido comentarios en el código.
- RA3.e) Se han creado programas ejecutables utilizando diferentes estructuras de control.
- RA3.f) Se han probado y depurado los programas.
- 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.b) Se han aplicado formatos en la visualización de la información.
- RA5.c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.

Contenidos: Ninguno del currículo. Podríamos saltarla.

Funciones, parámetros y valores de retorno. Paso por valor y referencia.

Sesiones: completar

Instrumentos de calificación:

5.1.1.5 UD5: Estructuras de almacenamiento en C

RAAs y CEs:

- RA1.f) Se han creado y utilizado constantes y literales.
- RA1.i) Se han introducido comentarios en el código.
- RA3.e) Se han creado programas ejecutables utilizando diferentes estructuras de control.
- RA3.f) Se han probado y depurado los programas.
- RA3.g) Se ha comentado y documentado el código.
- RA6.a) Se han escrito programas que utilicen matrices (arrays).
- RA5.a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información.
- RA5.b) Se han aplicado formatos en la visualización de la información.
- RA5.c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.

Contenidos: Estructuras. Arrays unidimensionales y multidimensionales: Declaración. Creación de arrays unidimensionales y multidimensionales. Inicialización Acceso a elementos. Recorridos, búsquedas y ordenaciones. Cadenas de caracteres: Declaración. Creación de cadenas de caracteres. Inicialización Operaciones. Acceso a elementos, conversiones, concatenación.

Opcional y extra: Gestión de la memoria dinámica en C

Sesiones: completar

Instrumentos de calificación: Registros de clase sobre la utilización del depurador (RA3f) Checklist de comentarios(RA3g) Prácticas individuales. Práctica grupal

5.1.1.6 UD6: Cambio de paradigma: La Orientación a Objetos

RAAs y CEs:

- RA1.f) Se han creado y utilizado constantes y literales.
- RA1.i) Se han introducido comentarios en el código.
- RA3.e) Se han creado programas ejecutables utilizando diferentes estructuras de control.
- RA3.f) Se han probado y depurado los programas.
- RA3.g) Se ha comentado y documentado el código.
- RA2.a) Se han identificado los fundamentos de la programación orientada a objetos.
- RA2.b) Se han escrito programas simples.
- RA2.c) Se han instanciado objetos a partir de clases predefinidas.
- RA2.d) Se han utilizado métodos y propiedades de los objetos.
- RA2.e) Se han escrito llamadas a métodos estáticos.
- RA2.f) Se han utilizado parámetros en la llamada a métodos.
- RA2.h) Se han utilizado constructores.
- RA2.i) Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples.
- RA5.a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información.
- RA5.b) Se han aplicado formatos en la visualización de la información.
- RA5.c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.

Contenidos: Clases. Atributos, métodos y visibilidad Objetos. Estado, comportamiento e identidad. Mensajes. Encapsulado. Visibilidad. Relaciones entre clases. Principios básicos de la orientación a objetos.

Utilización de objetos: Características de los objetos. Constructores. Instanciación de objetos. Declaración y creación. Utilización de métodos. Parámetros y valores de retorno. Utilización de propiedades. Utilización de métodos estáticos. Almacenamiento en memoria. Tipos básicos vs objetos. Destrucción de objetos y liberación de memoria.

Entrada/salida estándar: Entrada desde teclado. Salida a pantalla.

Sesiones: completar

Instrumentos de calificación: completar

5.1.1.7 UD7: Colecciones de datos

RAs y CEs:

- RA1.f) Se han creado y utilizado constantes y literales.
- RA1.i) Se han introducido comentarios en el código.
- RA3.e) Se han creado programas ejecutables utilizando diferentes estructuras de control.
- RA3.f) Se han probado y depurado los programas.
- 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.b) Se han aplicado formatos en la visualización de la información.
- RA5.c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.
- RA2.c) Se han instanciado objetos a partir de clases predefinidas.
- RA2.g) Se han incorporado y utilizado librerías de objetos.
- RA2.h) Se han utilizado constructores.
- RA2.i) Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples.
- RA6.a) Se han escrito programas que utilicen matrices (arrays).
- RA6.b) Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados.
- RA6.c) Se han utilizado listas para almacenar y procesar información.
- RA6.d) Se han utilizado iteradores para recorrer los elementos de las listas.
- RA6.e) Se han reconocido las características y ventajas de cada una de las colecciones de datos disponibles.

- RA6.h) Se han identificado las clases relacionadas con el tratamiento de documentos escritos en diferentes lenguajes de intercambio de datos.
- RA6.i) Se han realizado programas que realicen manipulaciones sobre documentos escritos en diferentes lenguajes de intercambio de datos.
- RA6.j) Se han utilizado operaciones agregadas para el manejo de información almacenada en colecciones.
- RA5.d) Se han utilizado ficheros para almacenar y recuperar información.
- RA5.e) Se han creado programas que utilicen diversos métodos de acceso al contenido de los ficheros.

Contenidos: *Colecciones de datos:* Tipos de colecciones (listas, pilas, colas, tablas). Jerarquías de colecciones. Operaciones con colecciones. Acceso a elementos y recorridos.

Sesiones: completar

Instrumentos de calificación: Prácticas individuales. Una práctica de JSON (RA6b,hi;RA5de)

5.1.1.8 UD8: POO Intermedia

RAAs y CEs:

- RA1.f) Se han creado y utilizado constantes y literales.
- RA1.i) Se han introducido comentarios en el código.
- RA3.e) Se han creado programas ejecutables utilizando diferentes estructuras de control.
- RA3.f) Se han probado y depurado los programas.
- 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.b) Se han aplicado formatos en la visualización de la información.
- RA5.c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.
- RA2*
- RA4.a) Se ha reconocido la sintaxis, estructura y componentes típicos de una clase.
- RA4.b) Se han definido clases.
- RA4.c) Se han definido propiedades y métodos.
- RA4.d) Se han creado constructores.
- RA4.e) Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.
- RA4.f) Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros.
- RA4.g) Se han definido y utilizado clases heredadas.
- RA4.h) Se han creado y utilizado métodos estáticos.

- RA4.i) Se han creado y utilizado conjuntos y librerías de clases.
- RA6(FALTA)
- RA7(FALTA)

Contenidos: *Utilización avanzada de clases:* Relaciones entre clases. Composición de clases. Herencia. Concepto y tipos (simple y múltiple). Superclases y subclases. Constructores y herencia. Modificadores en clases, atributos y métodos. Sobreescritura de métodos.

Flujos (streams): Tipos de flujos. Flujos de bytes y de caracteres. Clases relativas a flujos. Jerarquías de clases. Utilización de flujos. Entrada/salida estándar: Entrada desde teclado. Salida a pantalla.

Sesiones: completar

Instrumentos de calificación: completar

5.1.1.9 UD9: POO Avanzada

RAAs y CEs:

- RA1.f) Se han creado y utilizado constantes y literales.
- RA1.i) Se han introducido comentarios en el código.
- RA3.e) Se han creado programas ejecutables utilizando diferentes estructuras de control.
- RA3.f) Se han probado y depurado los programas.
- 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.b) Se han aplicado formatos en la visualización de la información.
- RA5.c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.
- RA2*
- RA4.a) Se ha reconocido la sintaxis, estructura y componentes típicos de una clase.
- RA4.b) Se han definido clases.
- RA4.c) Se han definido propiedades y métodos.
- RA4.d) Se han creado constructores.
- RA4.e) Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.
- RA4.f) Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros.
- RA4.g) Se han definido y utilizado clases heredadas.
- RA4.h) Se han creado y utilizado métodos estáticos.

- RA4.i) Se han creado y utilizado conjuntos y librerías de clases.
- RA6.f) Se han creado clases y métodos genéricos.
- RA7*

Contenidos: *Utilización avanzada de clases:* Clases y métodos abstractos y finales. Interfaces. Clases abstractas vs. Interfaces. Polimorfismo: Concepto. Polimorfismo en tiempo de compilación (sobrecarga) y polimorfismo en tiempo de ejecución (ligadura dinámica). Comprobación estática y dinámica de tipos. Conversiones de tipos entre objetos (casting). Clases y tipos genéricos o parametrizados.

Uso de clases y métodos genéricos. (del bloque Colecciones de datos)

completar

Sesiones: completar

Instrumentos de calificación: completar

5.1.1.10 UD10:Control y manejo de excepciones:

RAs y CEs:

- RA3.d) Se ha escrito código utilizando control de excepciones.
- RA3.h) Se han creado excepciones.
- RA3.i) Se han utilizado aserciones para la detección y corrección de errores durante la fase de desarrollo.

Contenidos: Excepciones. Concepto. Jerarquías de excepciones. Manejo de excepciones: Captura de excepciones. Propagar excepciones. Lanzar excepciones. Crear clases de excepciones.

Sesiones: completar

Instrumentos de calificación: completar

5.1.1.11 UD11: Lectura y escritura de información

RAs y CEs:

- RA5*

Contenidos: Almacenamiento de información en ficheros: Ficheros de datos. Registros. Apertura y cierre de ficheros. Modos de acceso. Escritura y lectura de información en ficheros. Almacenamiento de objetos en ficheros. Persistencia. Serialización. Utilización de los sistemas de ficheros. Creación y eliminación de ficheros y directorios. Interfaces gráficos de usuario simples. Concepto de evento. Creación de controladores de eventos.

Sesiones: completar

Instrumentos de calificación: completar

5.1.1.12 UD12: Gestión de bases de datos relacionales

RAs y CEs:

- RA8*
- RA9*

Contenidos: Interfaces de programación de acceso a bases de datos. Establecimiento de conexiones. Recuperación de información. Manipulación de la información. Ejecución de consultas sobre la base de datos.

Sesiones: completar

Instrumentos de calificación: completar

5.1.1.13 UD13: Mantenimiento de la persistencia de los objetos:

RAs y CEs: completar

Contenidos: Bases de datos orientadas a objetos. Características de las bases de datos orientadas a objetos. Instalación del gestor de bases de datos. Creación de bases de datos. Mecanismos de consulta. El lenguaje de consultas: sintaxis, expresiones, operadores. Recuperación, modificación y borrado de información. Tipos de datos objeto; atributos y métodos. Tipos de datos colección.

Sesiones: completar

Instrumentos de calificación: completar

6 Metodología didáctica

6.1 Principios metodológicos

La metodología propuesta en esta programación se fundamenta en el enfoque constructivista en el que el conocimiento se va construyendo poco a poco. Como ya se ha comentado más arriba (ver 1), partiremos de la base de que los alumnos no tienen ningún conocimiento previo para poder sentar unas sólidas bases de conocimiento que sirvan como cimiento para su futuro laboral como desarrolladores.

Además, la metodología será interactiva y contextualizada, adecuando siempre el lenguaje y las actividades a las características particulares del alumnado. Las explicaciones teóricas se abordarán mediante una metodología inductiva, la cual parte de la presentación inicial de retos o problemas prácticos para, después de haber razonado conjunta o individualmente, se ofrezca la formalización rigurosa del concepto que termine de resolver todas generalizaciones del problema. Esta metodología estimula la curiosidad natural del alumnado, favorece la implicación activa en la búsqueda de soluciones y facilita la comprensión profunda de los conceptos. Además, los conceptos que puedan elaborar los alumnos por sí mismos serán mucho más duraderos que los que sean ofrecidos por el docente y asimilados posteriormente por el alumno.

Por otro lado, el aprendizaje de destrezas profesionales demanda una participación activa del alumnado, por lo que se proporcionarán actividades variadas y eminentemente prácticas que permitan a los estudiantes entrenar y perfeccionar habilidades profesionales en contextos reales o simulados. Se alternarán tareas individuales y actividades cooperativas, fomentando la autonomía y al mismo tiempo el aprendizaje colaborativo. El alumno siempre tendrá a su disposición tantas tareas de cada tipo como necesite para poder asegurar que ha adquirido el aprendizaje.

Se aplicará el Diseño Universal para el Aprendizaje (DUA), mediante una diversidad de actividades y tareas que favorecen la participación plena de todo el alumnado, atendiendo así a las diferentes maneras de acceder al conocimiento y estilos de aprendizaje.

En cuanto a las entregas, predominan las individuales para fomentar la responsabilidad personal y la autoevaluación del alumno. Sin embargo, ocasionalmente se integrarán entregas grupales con problemas específicos que exijan una auténtica cooperación y trabajo conjunto. Incluso en actividades predominantemente individuales, se promoverán espacios de interacción grupal para que los estudiantes puedan intercambiar ideas, enriquecer su aprendizaje y practicar competencias sociales.

Las actividades estarán estructuradas según su función en el proceso de aprendizaje:

- **Actividades de iniciación y motivación:** Estimulan el interés inicial del alumnado.
- **Actividades de desarrollo y aprendizaje:** Introducen y profundizan en nuevos contenidos.
- **Actividades procedimentales:** Desarrollan competencias técnicas específicas.
- **Actividades de consolidación:** Refuerzan y afianzan el aprendizaje.
- **Actividades de refuerzo:** Dan apoyo adicional a estudiantes que requieren mayor atención.

6.2 Recursos

Aquí faltan muchas cosas

Finalmente, se integrarán herramientas TIC como parte esencial de la metodología educativa:

- Plataforma educativa (EducaMadrid) para gestión de contenidos y tareas, foros para la comunicación, cuestionarios interactivos, insignias digitales como reconocimiento del aprendizaje competencial, tareas colaborativas mediante wikis y glosarios interactivos para construir conocimiento compartido.
- Herramientas interactivas como Kahoot para gamificar contenidos teóricos.

- Jueces online y repositorios de código (Git) para el desarrollo práctico y colaborativo.
- Uso del ordenador del profesor en red local para entregas puntuales que deban realizarse sin conexión a internet.

6.3 Medidas de atención a la diversidad para alumnos con necesidad específica de apoyo educativo

Aquí también

Las medidas de atención a la diversidad para alumnado ACNEAE estarán enmarcadas dentro de lo previsto en el artículo 41 de la Orden 893/2022, de 21 de abril, de la Consejería de Educación, Universidades, Ciencia y Portavocía.

Desde la Programación se responde a la diversidad de capacidades, intereses, motivaciones y estilos particulares de aprender adoptando las siguientes medidas (que en ningún caso supondrán adaptaciones curriculares significativas):

7 Evaluación y calificación

7.1 Características de la evaluación

La evaluación del alumnado tendrá un carácter criterial, ateniéndose siempre a los criterios de evaluación vinculados a cada resultado de aprendizaje del módulo. Asimismo, conforme a lo establecido en el artículo 36 de la Orden 893/2022, de 21 de abril, de la Consejería de Educación, Universidades, Ciencia y Portavocía, se garantiza el carácter formativo de dicha evaluación, de modo que contribuya a la mejora continua del proceso de enseñanza-aprendizaje.

La metodología enfocada a la adquisición de competencias requiere un sistema de evaluación continua, ya que este facilita la evaluación formativa y proporciona oportunidades constantes de mejora. Sus principales características son:

- Realización periódica de actividades y pruebas evaluables, a lo largo del periodo lectivo, que:
 - Favorezcan la asimilación y el desarrollo progresivo de los contenidos y competencias del módulo.
 - Permitan al alumnado evidenciar, de forma continua, cómo va superando dificultades y adquiriendo las competencias propuestas.
- Atención a la diversidad, reconociendo que:
 - Cada estudiante cuenta con ritmos de aprendizaje, habilidades y capacidades diferentes.
 - Se emplearán procedimientos variados de evaluación para ofrecer más ocasiones de comprobar el progreso, corregir errores y mejorar.
 - Se considera la posibilidad de que el alumnado supere sus dificultades en cualquier momento del proceso.
- Requisitos fundamentales para la evaluación continua:

- Asistencia regular a clase.
- Registro documentado del desempeño de cada alumno.
- Análisis y retroalimentación sistemática:
 - Informar de los errores cometidos.
 - Proporcionar pautas de mejora antes de que el retraso en la adquisición de competencias sea irrecuperable.
- Autoevaluación docente y adaptación de la enseñanza cuando sea necesario, para optimizar el proceso formativo.

7.2 Procedimientos de evaluación y criterios de calificación

Cada PROCEDIMIENTO DE EVALUACIÓN lleva asociado un INSTRUMENTO DE CALIFICACIÓN (Listado agrupado de todo lo que se va a utilizar (que ya está en la UD)) En la Programación se presentará un listado completo y organizado de todos los que se aplicarán (pruebas escritas, rúbricas de proyectos, listas de control, registros de observación, etc.), de manera coherente con la estructura de las unidades de trabajo.

Con este planteamiento, se garantiza que la evaluación sea un proceso continuo, formativo y transparente, proporcionando al alumnado oportunidades reales de aprendizaje y mejora en consonancia con los principios de calidad e inclusión educativa.

- 7.3 Proceso de evaluación continua y calificación en la evaluación final ordinaria**
- 7.4 Proceso de evaluación para alumnos a los que no se puede aplicar la evaluación continua (pérdida del derecho a la evaluación continua)**
- 7.5 Proceso de evaluación y calificación en la evaluación final extraordinaria**
- 7.6 Medidas para alumnos con necesidad específica de apoyo educativo (2 ptos)**
- 7.7 Procedimiento de evaluación para alumnos con el módulo pendiente**
 - 7.7.7.1 Ordinaria**
 - 7.7.7.2 Extraordinaria**
- 7.8 Calendario de evaluaciones parciales, final ordinaria y final extraordinaria**

Índice alfabético

Competencia general del título, 8	(DUA), 45
Competencias personales y para la empleabilidad, 8	Estándares de competencia, 8
Criterios de Evaluación, 8	Metodología Didáctica, 8
Criterios de Evaluación (CE), 15	Objetivos, 8
Diseño Universal del Aprendizaje (DUA), 22	Resultados de Aprendizaje, 8, 15
Diseño Universal para el Aprendizaje	Saberes Básicos, 8, 24

Lista de TODOs

- Para el 18-03: Revisar legislación en total
- Para el 18-03: Revisar legislación que citar en cada apartado. Ver magister
- Relación (tabla) de unidades didácticas, situadas en el anexo. Aquí solo títulos y RAs con sus CEs
- completar
- completar
- completar
- completar
- completar
- completar
- completar
- completar

- completar
- completar
- completar
- completar
- completar
- completar
- completar
- completar
- completar
- completar
- completar
- completar
- Aquí faltan muchas cosas
- Aquí también
- Cada PROCEDIMIENTO DE EVALUACIÓN lleva asociado un INSTRUMENTO DE CALIFICACIÓN (Listado agrupado de todo lo que se va a utilizar (que ya está en la UD)) En la Programación se presentará un listado completo y organizado de todos los que se aplicarán (pruebas escritas, rúbricas de proyectos, listas de control, registros de observación, etc.), de manera coherente con la estructura de las unidades de trabajo.