|  |  |
| --- | --- |
| **PROGRAMACIÓN DIDACTICA**  **CURSO 2022/23** | |
| **ENTORNOS DE DESARROLLO** | |
| **1º CFGS TÉCNICO EN DESARROLLO DE APLICACIONES WEB**  **Informática y Comunicaciones** | |
| Interfaz de usuario gráfica  Descripción generada automáticamente con confianza media | |
| **ASPIRANTE**  **DAVID HORMIGO RAMÍREZ** | **CUERPO**  **PROFESORES DE ENSEÑANZA SECUNDARIA (590)** |
| **DNI**  **77453948E** | **ESPECIALIDAD**  **INFORMATICA (107)** |
| **NÚMERO DE ASPIRANTE**  **29003350** | **NÚMERO DE TRIBUNAL**  **7** |

Tabla de contenido

[Identificación y contexto 1](#_Toc139214621)

[Identificación del módulo profesional 1](#_Toc139214622)

[Marco legislativo completo 2](#_Toc139214623)

[Contextualización del módulo dentro de su entorno productivo 2](#_Toc139214624)

[Objetivos y Competencias 4](#_Toc139214625)

[Objetivos generales del título 4](#_Toc139214626)

[Competencias 5](#_Toc139214627)

[Resultados de aprendizaje y criterios de evaluación 7](#_Toc139214628)

[Unidades didácticas 10](#_Toc139214629)

[Contenidos básicos del módulo 10](#_Toc139214630)

[Contenidos Interdisciplinares 12](#_Toc139214631)

[Análisis y secuenciación de contenidos 12](#_Toc139214632)

[Temporalización y secuenciación 14](#_Toc139214633)

[Bloques de contenido 15](#_Toc139214634)

[Elementos organizados por unidades didácticas 16](#_Toc139214635)

[Relación entre resultados de aprendizaje, criterios de evaluación y unidades didácticas 27](#_Toc139214636)

[Metodología 28](#_Toc139214637)

[Metodología de aula 28](#_Toc139214638)

[Instrumentos de seguimiento 28](#_Toc139214639)

[Tipos de instrumento de evaluación 28](#_Toc139214640)

[Recursos didácticos y materiales 28](#_Toc139214641)

[Usos de las TIC y herramientas de trabajo colaborativo 29](#_Toc139214642)

[Organización de los espacios 29](#_Toc139214643)

[Evaluación y calificación 29](#_Toc139214644)

[Criterios de evaluación por resultados de aprendizaje 30](#_Toc139214645)

[Ponderación de cada resultado de aprendizaje en la nota del módulo 30](#_Toc139214646)

[Ponderación de los criterios de evaluación de cada resultado de aprendizaje 30](#_Toc139214647)

[Sistemas y criterios de recuperación y mejora de la calificación 32](#_Toc139214648)

[Autoevaluación 33](#_Toc139214649)

[De la programación didáctica 33](#_Toc139214650)

[De la labor docente 33](#_Toc139214651)

[Atención a la diversidad 33](#_Toc139214652)

[Atención a los alumnos con necesidades específicas de apoyo educativo 33](#_Toc139214653)

[Atención a los diferentes ritmos de aprendizaje 33](#_Toc139214654)

[Actividades complementarias y extraescolares 33](#_Toc139214655)

[Bibliografía 34](#_Toc139214656)

[Anexos i](#_Toc139214657)

[Anexo I. Indicadores de evaluación de la actividad docente i](#_Toc139214658)

# Identificación y contexto

## Identificación del módulo profesional

El presente documento desarrolla la programación didáctica para el módulo profesional de Entornos de Desarrollo.

|  |  |
| --- | --- |
| **Familia Profesional** | Informática y Comunicaciones |
| **Nivel** | Ciclo Formativo de Grado Superior |
| **Título** | Desarrollo de aplicaciones web |
| **Nombre del módulo** | Entornos de desarrollo |
| **Código** | 0487 |
| **Equivalencia en créditos ECTS** | 6 |
| **Curso** | Primero |
| **Horas totales** | 96 |
| **Horas semanales** | 3 |
| **Referente en la Clasificación Internacional Normalizada de la educación** | Nivel 5 |

### Unidades de competencia y cualificaciones profesionales

Según el Real Decreto 686/2010, el título tiene incluye las siguientes cualificaciones profesionales y unidades de competencia:

1. Cualificaciones profesionales completas:

Desarrollo de aplicaciones con tecnologías Web **IFC154\_3** que comprende las siguientes unidades de competencia:

**UC0491\_3** Desarrollar elementos software en el entorno cliente.

**UC0492\_3** Desarrollar elementos software en el entorno servidor.

**UC0493\_3** Implementar, verificar y documentar aplicaciones web en entornos internet, intranet y extranet.

1. Cualificaciones profesionales incompletas:
   1. Programación en lenguajes estructurados de aplicaciones de gestión **IFC155\_3**.

**UC0223\_3** Configurar y explotar sistemas informáticos.

**UC0226\_3** Programar bases de datos relacionales.

* 1. Programación con lenguajes orientados a objetos y bases de datos relacionales **IFC 080\_3**.

**UC0223\_3** Configurar y explotar sistemas informáticos.

**UC0226\_3** Programar bases de datos relacionales.

Aunque, según la orden de 16 de junio de 2011, el módulo profesional de Entornos de Desarrollo no está asociado directamente a cualificaciones profesionales este si contribuye a alcanzar las unidades de competencias **UC0491\_3**, **UC0492\_3** y **UC0493\_3**, ya que posibilita el manejo efectivo de entornos y herramientas para el desarrollo de software, la gestión de documentación y la implementación de pruebas de software

## Marco legislativo completo

**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.

**Decreto 334/2009 de 22 de septiembre**, por el que se regulan los centros integrados de formación profesional en la Comunidad Autónoma de Andalucía (BOJA 05-10-2009).

**Decreto 382/2010 de 13 de octubre**, por el que se modifica el Decreto 334/2009, de 22 de septiembre, por el que se regulan los centros integrados de Formación Profesional en la Comunidad Autónoma de Andalucía, para adecuarlo a la Directiva 2006/123/CE del Parlamento Europeo y del Consejo, de 12 de diciembre de 2006, relativa a los servicios en el mercado interior (BOJA 28-10-2010).

**Real Decreto 1147/2011, de 29 de julio**, por el que se establece la ordenación general de la formación profesional del sistema educativo (BOE 30/07/2011)

**Ley Orgánica 3/2022, de 31 de marzo**, de ordenación e integración de la Formación Profesional (BOE 01-04-2022).

**Real Decreto 686/2010, de 20 de mayo**, por el que se establece el título de Técnico Superior en Desarrollo de Aplicaciones Web y se fijan sus enseñanzas mínimas.

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

## Contextualización del módulo dentro de su entorno productivo

### Contextualización del Centro

En el desarrollo de la presente programación didáctica se ha tenido en cuenta las características particulares del centro y el entorno económico-social de la zona donde se sitúa.

El Centro Publico Integrado de Formación Profesional Alan Turing se encuentra ubicado en el Parque Tecnología de Andalucía (Málaga TechPark), uno de los principales polos de trabajo en el sector de las nuevas tecnologías en Andalucía. Esta ubicación marca las relaciones del centro con el tejido productivo de sus alrededores y establece sinergias importantísimas con empresas punteras en las nuevas tecnologías, influyendo notablemente en la planificación de las enseñanza y la adecuación de los contenidos.

Como CPFIP, las enseñanzas impartidas en el centro son exclusivamente de Formación Profesional en la familias de informatica y comunicaciones y en la familia de electricidad y electrónica. Dentro de la familia de informatica y comunicaciones, se imparten el CFGM de Técnico en Sistemas Microinformáticos y Redes, el CFGS de Técnico Superior en Administración de Sistemas Informáticos en Red, Técnico Superior en Desarrollo de Aplicaciones Multiplataforma, Técnico Superior en Desarrollo de Aplicaciones Web, Curso de Especialización en Inteligencia Artificial y Big Data, Curso de Especialización en Ciberseguridad en Entornos de las Tecnologías de la Información y el Curso de Especialización en Desarrollo de videojuegos y realidad virtual. Esta amplia oferta de enseñanzas especializadas y su ubicación en pleno Málaga TechPark atrae alumnado de toda la geografía andaluza.

Dicho parque empresarial se encuentra ubicado en el distrito de Campanillas de la ciudad de Málaga. En este distrito podemos encontrar desde barrios muy desfavorecidos como son Los Asperones a otros acomodados como El Brillante. Esto, junto con las características intrínsecas del Málaga TechPark, tiene como consecuencia una ampliación en la diversidad socioeconómica del alumnado que recibe el centro.

### Contextualización de los profesionales del desarrollo de aplicaciones web

Dentro del ciclo de Técnico Superior en Desarrollo de Aplicaciones Web, el módulo de Entornos de Desarrollo aporta conocimientos y competencias necesarios para el uso efectivo de las herramientas específicas para el desarrollo de software, la construcción de software fiable y la gestión de documentación de software. La evolución actual del desarrollo de software hace vital que los profesionales sean competentes en el uso de las herramientas específicas de su práctica laboral.

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

* Selección y uso de entornos integrados de desarrollo
* Elaboración de documentación externa e interna de software
* Uso de herramientas de control de versiones
* Utilización de repositorios remotos
* Diseño y ejecución de planes de prueba de software
* Implementación de pruebas unitarias automáticas de software
* Implementación de pruebas automáticas de integración
* Diseño e interpretación de diagramas de clase
* Diseño e interpretación de diagramas de comportamiento

# Objetivos y Competencias

## Objetivos generales del título

Los objetivos generales de este ciclo formativo son los siguientes:

1. Ajustar la configuración lógica analizando las necesidades y criterios establecidos para configurar y explotar sistemas informáticos.
2. Identificar las necesidades de seguridad verificando el plan preestablecido para aplicar técnicas y procedimientos relacionados.
3. Instalar módulos analizando su estructura y funcionalidad para gestionar servidores de aplicaciones.
4. Ajustar parámetros analizando la configuración para gestionar servidores de aplicaciones.
5. Interpretar el diseño lógico, verificando los parámetros establecidos para gestionar bases de datos.
6. Seleccionar lenguajes, objetos y herramientas, interpretando las especificaciones para desarrollar aplicaciones web con acceso a bases de datos.
7. Utilizar lenguajes, objetos y herramientas, interpretando las especificaciones para desarrollar aplicaciones web con acceso a bases de datos.
8. Generar componentes de acceso a datos, cumpliendo las especificaciones, para integrar contenidos en la lógica de una aplicación web.
9. Utilizar lenguajes de marcas y estándares web, asumiendo el manual de estilo, para desarrollar interfaces en aplicaciones web.
10. Emplear herramientas y lenguajes específicos, siguiendo las especificaciones, para desarrollar componentes multimedia.
11. Evaluar la interactividad, accesibilidad y usabilidad de un interfaz, verificando los criterios preestablecidos, para Integrar componentes multimedia en el interfaz de una aplicación.
12. Utilizar herramientas y lenguajes específicos, cumpliendo las especificaciones, para desarrollar e integrar componentes software en el entorno del servidor web.
13. Emplear herramientas específicas, integrando la funcionalidad entre aplicaciones, para desarrollar servicios empleables en aplicaciones web.
14. Evaluar servicios distribuidos ya desarrollados, verificando sus prestaciones y funcionalidad, para integrar servicios distribuidos en una aplicación web.
15. Verificar los componentes de software desarrollados, analizando las especificaciones, para completar el plan de pruebas.
16. Utilizar herramientas específicas, cumpliendo los estándares establecidos, para elaborar y mantener la documentación de los procesos.
17. Establecer procedimientos, verificando su funcionalidad, para desplegar y distribuir aplicaciones.
18. Programar y realizar actividades para gestionar el mantenimiento de los recursos informáticos.
19. Analizar y utilizar los recursos y oportunidades de aprendizaje relacionadas con la evolución científica, tecnológica y organizativa del sector y las tecnologías de la información y la comunicación, para mantener el espíritu de actualización y adaptarse a nuevas situaciones laborales y personales.
20. Desarrollar la creatividad y el espíritu de innovación para responder a los retos que se presentan en los procesos y organización de trabajo y de la vida personal.
21. Tomar decisiones de forma fundamentada analizando las variables implicadas, integrando saberes de distinto ámbito y aceptando los riesgos y la posibilidad de equivocación en las mismas, para afrontar y resolver distintas situaciones, problemas o contingencias.
22. Desarrollar técnicas de liderazgo, motivación, supervisión y comunicación en contextos de trabajo en grupo para facilitar la organización y coordinación de equipos de trabajo.
23. Aplicar estrategias y técnicas de comunicación adaptándose a los contenidos que se van a transmitir, la finalidad y a las características de los receptores, para asegurar la eficacia en los procesos de comunicación.
24. Evaluar situaciones de prevención de riesgos laborales y de protección ambiental, proponiendo y aplicando medidas de prevención personales y colectivas, de acuerdo a la normativa aplicable en los procesos del trabajo, para garantizar entornos seguros.
25. Identificar y proponer las acciones profesionales necesarias para dar respuesta a la accesibilidad universal y al diseño para todos.
26. Identificar y aplicar parámetros de calidad en los trabajos y actividades realizados en el proceso de aprendizaje para valorar la cultura de la evaluación y de la calidad y ser capaces de supervisar y mejorar procedimientos de gestión de calidad.
27. Utilizar procedimientos relacionados con la cultura emprendedora, empresarial y de iniciativa profesional, para realizar la gestión básica de una pequeña empresa o emprender un trabajo.
28. Reconocer sus derechos y deberes como agente activo en la sociedad, teniendo en cuenta el marco legal que regula las condiciones sociales y laborales para participar como ciudadano democrático.

El módulo de Entornos de desarrollo contribuye a alcanzar las competencias f, g, j, l, m, o y p.

## Competencias

### Competencia general

El Real Decreto 686/2010, de 20 de mayo, en su artículo 4 establece que la competencia general de este título consiste en desarrollar, implantar, y mantener aplicaciones web, con independencia del modelo empleado y utilizando tecnologías específicas, garantizando el acceso a los datos de forma segura y cumpliendo los criterios de accesibilidad, usabilidad y calidad exigidas en los estándares establecidos.

El módulo profesional de Entornos de Desarrollo contribuye a alcanzar dicha competencia general a través de formar en el uso efectivo de herramientas desarrollo y la creación de software de calidad y fiable.

### Competencias profesionales, personales y sociales

El Real Decreto 686/2010, de 20 de mayo, en su artículo 5 establece que las competencias profesionales, personales y sociales de este título son las que se relacionan a continuación:

1. Configurar y explotar sistemas informáticos, adaptando la configuración lógica del sistema según las necesidades de uso y los criterios establecidos.
2. Aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad.
3. Gestionar servidores de aplicaciones adaptando su configuración en cada caso para permitir el despliegue de aplicaciones web.
4. Gestionar bases de datos, interpretando su diseño lógico y verificando integridad, consistencia, seguridad y accesibilidad de los datos.
5. Desarrollar aplicaciones web con acceso a bases de datos utilizando lenguajes, objetos de acceso y herramientas de mapeo adecuados a las especificaciones.
6. Integrar contenidos en la lógica de una aplicación web, desarrollando componentes de acceso a datos adecuados a las especificaciones.
7. Desarrollar interfaces en aplicaciones web de acuerdo con un manual de estilo, utilizando lenguajes de marcas y estándares web.
8. Desarrollar componentes multimedia para su integración en aplicaciones web, empleando herramientas específicas y siguiendo las especificaciones establecidas.
9. Integrar componentes multimedia en el interface de una aplicación web, realizando el análisis de interactividad, accesibilidad y usabilidad de la aplicación.
10. Desarrollar e integrar componentes software en el entorno del servidor web, empleando herramientas y lenguajes específicos, para cumplir las especificaciones de la aplicación.
11. Desarrollar servicios para integrar sus funciones en otras aplicaciones web, asegurando su funcionalidad.
12. Integrar servicios y contenidos distribuidos en aplicaciones web, asegurando su funcionalidad.
13. Completar planes de pruebas verificando el funcionamiento de los componentes software desarrollados, según las especificaciones.
14. Elaborar y mantener la documentación de los procesos de desarrollo, utilizando herramientas de generación de documentación y control de versiones.
15. Desplegar y distribuir aplicaciones web en distintos ámbitos de implantación, verificando su comportamiento y realizando modificaciones.
16. Gestionar y/o realizar el mantenimiento de los recursos de su área en función de las cargas de trabajo y el plan de mantenimiento.
17. Adaptarse a las nuevas situaciones laborales, manteniendo actualizados los conocimientos científicos, técnicos y tecnológicos relativos a su entorno profesional, gestionando su formación y los recursos existentes en el aprendizaje a lo largo de la vida y utilizando las tecnologías de la información y la comunicación.
18. Resolver situaciones, problemas o contingencias con iniciativa y autonomía en el ámbito de su competencia, con creatividad, innovación y espíritu de mejora en el trabajo personal y en el de los miembros del equipo.
19. Organizar y coordinar equipos de trabajo, supervisando el desarrollo del mismo, con responsabilidad, manteniendo relaciones fluidas y asumiendo el liderazgo, así como, aportando soluciones a los conflictos grupales que se presentan.
20. Comunicarse con sus iguales, superiores, clientes y personas bajo su responsabilidad utilizando vías eficaces de comunicación, transmitiendo la información o conocimientos adecuados, y respetando la autonomía y competencia de las personas que intervienen en el ámbito de su trabajo.
21. Generar entornos seguros en el desarrollo de su trabajo y el de su equipo, supervisando y aplicando los procedimientos de prevención de riesgos laborales y ambientales de acuerdo con lo establecido por la normativa y los objetivos de la empresa.
22. Supervisar y aplicar procedimientos de gestión de calidad, de accesibilidad universal y de diseño para todos, en las actividades profesionales incluidas en los procesos de producción o prestación de servicios.
23. Realizar la gestión básica para la creación y funcionamiento de una pequeña empresa y tener iniciativa en su actividad profesional con sentido de la responsabilidad social.
24. Ejercer sus derechos y cumplir con las obligaciones derivadas de su actividad profesional, de acuerdo con lo establecido en la legislación vigente, participando activamente en la vida económica, social y cultural.

A las competencias e, f, h, j, k, m y n se contribuye desde el módulo de Entornos de Desarrollo.

.

# Resultados de aprendizaje y criterios de evaluación

Según la Orden del 16 de julio de 2011, los resultados de aprendizaje del módulo de **Entornos de Desarrollo** son:

1. Reconoce los elementos y herramientas que intervienen en el desarrollo de un programa informático, analizando sus características y las fases en las que actúan hasta llegar a su puesta en funcionamiento.

**Criterios de evaluación:**

1. Se ha reconocido la relación de los programas con los componentes del sistema informático, memoria, procesador, periféricos, entre otros.
2. Se han clasificado los lenguajes de programación.
3. Se han diferenciado los conceptos de código fuente, objeto y ejecutable.
4. Se han reconocido las características de la generación de código intermedio para su ejecución en máquinas virtuales.
5. Se ha evaluado la funcionalidad ofrecida por las herramientas utilizadas en programación.
6. Se ha diferenciado el funcionamiento de los distintos tipos de traductores de lenguajes ante el código fuente de un programa.
7. Se han identificado las fases de desarrollo de una aplicación informática.

2. Evalúa entornos integrados de desarrollo analizando sus características para editar código fuente y generar ejecutable.

**Criterios de evaluación:**

1. Se han instalado entornos de desarrollo, propietarios y libres.
2. Se han añadido y eliminado módulos en el entorno de desarrollo.
3. Se ha personalizado y automatizado el entorno de desarrollo.
4. Se ha configurado el sistema de actualización del entorno de desarrollo.
5. Se han generado ejecutables a partir de código fuente de diferentes lenguajes en un mismo entorno de desarrollo.
6. Se han generado ejecutables a partir de un mismo código fuente con varios entornos de desarrollo.
7. Se han identificado las características comunes y específicas de diversos entornos de desarrollo.
8. Se han identificado las funciones más usuales de las herramientas CASE para el desarrollo, prueba y documentación de código.

3. Verifica el funcionamiento de programas diseñando y realizando pruebas.

**Criterios de evaluación:**

1. Se han identificado los diferentes tipos de pruebas.
2. Se han definido casos de prueba.
3. Se han identificado las herramientas de depuración y prueba de aplicaciones ofrecidas por el entorno de desarrollo.
4. Se han utilizado herramientas de depuración para definir puntos de ruptura y seguimiento.
5. Se han utilizado las herramientas de depuración para examinar y modificar el comportamiento de un programa en tiempo de ejecución.
6. Se ha documentado el plan de pruebas.
7. Se han efectuado pruebas unitarias de clases y funciones.
8. Se han efectuado pruebas de integración, de sistema y de aceptación.
9. Se han implementado pruebas automáticas.
10. Se han documentado las incidencias detectadas.
11. Se han aplicado normas de calidad a los procedimientos de desarrollo de software.
12. Se han realizado medidas de calidad sobre el software desarrollado.

4. Optimiza código empleando las herramientas disponibles en el entorno de desarrollo.

**Criterios de evaluación:**

1. Se han identificado los patrones de refactorización más usuales.
2. Se han elaborado las pruebas asociadas a la refactorización.
3. Se ha revisado el código fuente usando un analizador de código.
4. Se han identificado las posibilidades de configuración de un analizador de código.
5. Se han aplicado patrones de refactorización con las herramientas que proporciona el entorno de desarrollo.
6. Se ha realizado el control de versiones integrado en el entorno de desarrollo.
7. Se ha documentado el código fuente mediante comentarios.
8. Se han utilizado herramientas del entorno de desarrollo para documentar los procesos, datos y eventos.
9. Se han utilizado herramientas del entorno de desarrollo para documentar las clases.

5. Genera diagramas de clases valorando su importancia en el desarrollo de aplicaciones y empleando las herramientas disponibles en el entorno.

**Criterios de evaluación:**

1. Se han identificado los conceptos básicos de la programación orientada a objetos.
2. Se ha instalado el módulo del entorno integrado de desarrollo que permite la utilización de diagramas de clases.
3. Se han identificado las herramientas para la elaboración de diagramas de clases.
4. Se ha interpretado el significado de diagramas de clases.
5. Se han trazado diagramas de clases a partir de las especificaciones de las mismas.
6. Se ha generado código a partir de un diagrama de clases.
7. Se ha generado un diagrama de clases mediante ingeniería inversa.

6. Genera diagramas de comportamiento valorando su importancia en el desarrollo de aplicaciones y empleando las herramientas disponibles en el entorno.

**Criterios de evaluación:**

1. Se han identificado los distintos tipos de diagramas de comportamiento.
2. Se ha reconocido el significado de los diagramas de casos de uso.
3. Se han interpretado diagramas de interacción.
4. Se han elaborado diagramas de interacción sencillos.
5. Se han interpretado diagramas de estados.
6. Se han planteado diagramas de estados sencillos.
7. Se ha interpretado el significado de diagramas de actividades.
8. Se han elaborado diagramas de actividades sencillos.

# Unidades didácticas

## Contenidos básicos del módulo

Según la Orden del 16 de julio de 2011 los contenidos del módulo profesional de Entornos de Desarrollo son los siguientes:

* Reconocimiento de elementos del desarrollo de software:
  + Conceptos de programa informático y de aplicación informática.
  + Concepto de lenguaje de programación.
  + Tipos de lenguajes de programación.
  + Características de los lenguajes más difundidos.
  + Código fuente, código objeto y código ejecutable; máquinas virtuales.
  + Proceso de obtención de código ejecutable a partir del código fuente; herramientas implicadas.
    - Traductores de lenguajes.
    - Depuradores.
  + Fases del desarrollo de una aplicación, análisis, diseño, codificación, pruebas, documentación, explotación y mantenimiento, entre otras.
* Evaluación de entornos integrados de desarrollo:
  + Funciones de un entorno de desarrollo.
  + Componentes de un entorno de desarrollo.
  + Instalación de un entorno de desarrollo.
  + Mecanismo de actualización de un entorno de desarrollo.
  + Uso básico de un entorno de desarrollo.
    - Edición de programas.
    - Generación de ejecutables.
  + Entornos de desarrollo libres y comerciales más usuales.
  + Uso de herramientas CASE en el desarrollo de software.
* Diseño y realización de pruebas:
  + Tipos de pruebas, funcionales, estructurales, regresión.
  + Procedimientos y casos de prueba.
  + Pruebas de Código, cubrimiento, valores límite, clases de equivalencia.
  + Herramientas de depuración de código.
  + Planificación de Pruebas.
    - Pruebas unitarias; herramientas.
    - Pruebas de integración.
    - Pruebas del sistema.
    - Pruebas de aceptación.
    - Automatización de pruebas.
  + Calidad del software.
    - Normas y certificaciones.
    - Medidas de calidad del software.
* Optimización y documentación:
  + Refactorización.
    - Concepto y limitaciones.
    - Patrones de refactorización más usuales.
    - Refactorización y pruebas.
    - Herramientas de ayuda a la refactorización.
  + Control de versiones.
    - Concepto y características.
    - Tipos.
    - Herramientas.
    - Repositorio.
  + Documentación.
    - Uso de comentarios.
    - Alternativas.
    - Herramientas.
* Elaboración de diagramas de clases:
  + Notación de los diagramas de clases.
    - Clases. Atributos, métodos y visibilidad.
    - Objetos. Instanciación.
    - Relaciones. Herencia, composición, agregación, asociación y uso.
  + Herramientas para la elaboración de diagramas de clases. Instalación.
  + Generación de código a partir de diagramas de clases.
  + Generación de diagramas de clases a partir de código.
* Elaboración de diagramas de comportamiento:
  + Tipos. Campo de aplicación.
  + Diagramas de casos de uso. Actores, escenario, relación de comunicación.
  + Diagramas de interacción.
    - Diagramas de secuencia. Línea de vida de un objeto, activación, envío de mensajes.
    - Diagramas de colaboración. Objetos, mensajes.
    - Diagramas de estados. Estados, eventos, señales, transiciones.
    - Diagramas de actividades. Actividades, transiciones, decisiones y combinaciones.

## Contenidos Interdisciplinares

El módulo profesional de Entornos de Desarrollo está íntimamente ligado al módulo profesional de Programación (0485) y es vital realizar una coordinación para la consecución de objetivos y el desarrollo de competencias. Ambos módulos comparten ciertos contenidos como son:

* Entornos integrados de desarrollo
  + Definición y tipos. Entornos comerciales y de Software libre.
  + Instalación y descripción de entornos integrados de desarrollo.
  + Creación de proyectos. Estructura y componentes.
* Entornos de desarrollo para programación orientada a objetos.
  + Entornos específicos.
  + Plugins de integración en entornos genéricos.
* Depuración de programas.
  + El depurador como herramienta de control de errores.
  + Documentación de programas.
* Documentación interna, comentarios.
* Documentación externa, diagramas de clases, requisitos, guías, etc.

Es por ello por lo que ciertos contenidos de este módulo han sido secuenciados de tal manera que ambos módulos se beneficien. De esta manera las UD 4 y 5 se han planificado para ser impartidas durante la primera evaluación. A su vez, las unidades 6 y 7 serán trabajadas al principio de la segunda evaluación en sincronía con la

A su vez, se apoya en el módulo profesional de Sistemas Informáticos para la creación de contenedores y maquinas virtuales para la creación de entornos locales de desarrollo.

Finalmente, los contenidos impartidos en el módulo de Lenguajes de Marcas y Sistemas de Información sobre el uso de HTML y CSS se ha tenido en cuenta a la hora de planificar la UD 11.

## Temporalización y secuenciación

Según el calendario escolar oficial, para el curso 2022/2023 se disponen de 102 horas lectivas en la localidad de Málaga para el módulo de Entornos de Desarrollo. En la Tabla 1 se puede ver la distribución horaria de cada unidad didáctica y su temporalización por evaluaciones.

|  |  |  |
| --- | --- | --- |
| **Evaluación** | **Unidad didáctica** | **Horas** |
| **Primera** | 1. Introducción al desarrollo de software | 6 Horas |
| 2. Instalación y uso de entornos integrados de desarrollo | 9 Horas |
| 3. Ciclo de vida del software. Metodologías | 3 Horas |
| 4. Defectos de software y depuración | 9 Horas |
| 5. Control de versiones, repositorios remotos y colaboración | 12 Horas |
|  | | 39 Horas |
| **Segunda** | 6. Diagramas de comportamiento | 9 Horas |
| 7. Diagramas de clase | 9 Horas |
| 8. Introducción a las pruebas de software | 6 Horas |
| 9. Diseño e implementación de pruebas unitarias | 15 Horas |
|  | | 39 Horas |
| **Tercera** | 10. Diseño e implementación de pruebas de integración | 15 Horas |
| 11. Documentación de software | 3 Horas |
| 12. Código limpio y refactorización | 6 Horas |
|  | | 24 Horas |

Tabla . Temporalización y secuenciación UD

## Bloques de contenido

Las unidades didácticas están divididas en bloques conceptuales como se puede ver en la Tabla 2. Es importante señalar que estos bloques no se corresponden a las evaluaciones del curso y que obedecen a una agrupación de contenidos afines.

|  |  |
| --- | --- |
| **Bloque** | **Unidades** |
| El desarrollo de software y sus herramientas | UD 1. Introducción al desarrollo de Software  UD 2. Instalación y uso de entornos de desarrollo integrado  UD 3. Ciclo de vida del software. Metodologías  UD 5. Control de versiones, repositorios remotos y colaboración |
| Documentación y requisitos de software | UD 6. Diseño orientado a objetos. Diagramas de comportamiento  UD 10. Diseño orientado a objetos. Diagramas estructurales  UD 11. Documentación de software |
| Calidad, pruebas y refactorización | UD 4. Defectos de software y depuración  UD 8. Introducción a las pruebas de software  UD 9. Diseño e implementación de pruebas unitarias  UD 9. Diseño e implementación de pruebas de integración  UD 12. Código limpio y refactorización de código |

Tabla . Bloques conceptuales del módulo

## Elementos organizados por unidades didácticas

### Unidad Didáctica 1. Introducción al desarrollo del software

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Resultados Aprendizaje** | | | | | | **CPPyS** | | **OG** |
| 1. Reconoce los elementos y herramientas que intervienen en el desarrollo de un programa informático, analizando sus características y las fases en las que actúan hasta llegar a su puesta en funcionamiento | | | | | |  | |  |
| **Objetivos de aprendizaje** | | | | | | | | |
| 1. Entender los conceptos básicos del proceso de desarrollo de software 2. Conocer las diferencias y características de los lenguajes de programación 3. Entender el proceso de compilación y sus peculiaridades en función del lenguaje 4. Reconocer los elementos implicados en el proceso de desarrollo de software | | | | | | | | |
| **Contenidos** | | | | | | | | |
| ¿Qué es el desarrollo del software?  Lenguajes de programación compilados e interpretados  Ejecutar un programa en Python  Proceso de compilación en C  Proceso de compilación en Java. Java Development Toolkit y Apache Maven | | | | | | | | |
| **Actividades** | | | | | | | | |
| F.Metod. | **Tipo** | **Actividad** | **Objetivos** | **Actores** | | | **Tiempo** | |
| Explicit. | Ev. Ini | Cuestionario de conocimientos previos. |  | Alum  Prof | | | ½ h. | |
|  | Des. | Impartición UD 1 |  | Alumnos  Profesor. | | | 3 h. | |
|  | Des. | Como realizar un mapa conceptual |  | Profesor-Alumnos | | | ½ h. | |
|  | Des. | Clasificar lenguajes de programación |  | Alumnos | | | ½ h | |
|  |  |  |  |  | | |  | |
|  |  |  |  |  | | |  | |
|  |  |  |  |  | | |  | |
|  |  |  |  |  | | |  | |
|  |  |  |  |  | | |  | |
|  |  |  |  |  | | |  | |
| Evaluación | | | | | | | | |
| Criterios de Evaluación | | | | | | | % Cal. | |
|  | | | | | | |  | |
|  | | | | | | |  | |
|  | | | | | | |  | |
| Actividades de Evaluación | | | | | | | | |
| CE | Actividad | | | | Instrumento | | % Cal | |
|  | Clasificar lenguajes de programación | | | | Rubrica | | 20% | |
|  | Prueba escrita UD 1 | | | | Solucionario | | 40% | |

### Unidad Didáctica 2. Instalación y uso de entornos de desarrollo integrado

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Resultados Aprendizaje** | | | **CPPyS** | | **Objetivos Generales** | | |
| 2 | | |  | |  | | |
| Objetivos de aprendizaje | | | | | | | |
| 1. Seleccionar Entornos de Desarrollo en función de las necesidades y características del lenguaje de programación 2. Configurar el entorno de desarrollo para el uso eficiente en el proceso de desarrollo | | | | | | | |
| Contenidos | | | | | | | |
| ¿Qué es un IDE?  IDEs de propósito General: Visual Studio Code  IDEs especializados. IntelliJ IDEA  Configuración y personalización de un IDE  Instalación y uso de extensiones de un IDE | | | | | | | |
| **Actividades** | | | | | | | |
| F.Metod. | | Tipo | Actividad | Objetivos | Actores | | Tiempo |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
| Evaluación | | | | | | | |
| Criterios de Evaluación | | | | | | | % Cal. |
|  | | | | | | |  |
|  | | | | | | |  |
|  | | | | | | |  |
| Actividades de Evaluación | | | | | | | |
| CE | Actividad | | | | | Instrumento | % Cal |
|  | Instalación Visual Studio Code | | | | |  |  |
|  | Instalación IntelliJ IDEA | | | | |  |  |
|  | Temas y fuentes en Visual Studio | | | | |  |  |
|  | Instalar extensiones para Visual Studio | | | | |  |  |
|  | Generar ejecutable Java desde Visual Studio Code | | | | |  |  |

### Unidad Didáctica 3. Ciclo de vida del software. Metodologías

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Resultados Aprendizaje | | | CPPyS | | Objetivos Generales | | |
|  | | |  | |  | | |
| Objetivos de aprendizaje | | | | | | | |
| Analizar la necesidad de una ingeniería del software  Identificar las diferentes fases que componen el ciclo de desarrollo de software  Comparar metodologías de desarrollo de software  Usar herramientas de gestión de tareas | | | | | | | |
| Contenidos | | | | | | | |
| La necesidad de una ingeniería del software  Ciclo de vida del software. Fases  Metodologías no agiles: En Cascada, En espiral  Metodologías agiles: Scrum, Kanban, Programación Extrema  Herramientas de gestión de proyectos ágiles: Trello | | | | | | | |
| Actividades | | | | | | | |
| F.Metod. | | Tipo | Actividad | Objetivos | Actores | | Tiempo |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
| Evaluación | | | | | | | |
| Criterios de Evaluación | | | | | | | % Cal. |
|  | | | | | | |  |
|  | | | | | | |  |
|  | | | | | | |  |
| Actividades de Evaluación | | | | | | | |
| CE | Actividad | | | | | Instrumento | % Cal |
|  |  | | | | |  |  |
|  | Prueba escrita UD 1 | | | | | Solucionario | 40 |

### Unidad Didáctica 4. Defectos de software y depuración

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Resultados Aprendizaje** | | | **CPPyS** | | **Objetivos Generales** | | |
|  | | |  | |  | | |
| **Objetivos de aprendizaje** | | | | | | | |
|  | | | | | | | |
| **Contenidos** | | | | | | | |
| El concepto de “bug”  Errores en tiempo de compilación  Uso del linter para el análisis de errores de compilación  Errores en tiempo de ejecución  Uso del depurador. Puntos de ruptura, edición de variables, control del flujo | | | | | | | |
| Actividades | | | | | | | |
| F.Metod. | | Tipo | Actividad | Objetivos | Actores | | Tiempo |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
| Evaluación | | | | | | | |
| Criterios de Evaluación | | | | | | | % Cal. |
|  | | | | | | |  |
|  | | | | | | |  |
|  | | | | | | |  |
| Actividades de Evaluación | | | | | | | |
| CE | Actividad | | | | | Instrumento | % Cal |
|  |  | | | | |  |  |
|  | Prueba escrita UD 1 | | | | | Solucionario | 40 |

### Unidad Didáctica 5. Control de versiones, repositorios remoto y colaboración

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Resultados Aprendizaje | | | CPPyS | | Objetivos Generales | | |
|  | | |  | |  | | |
| Objetivos de aprendizaje | | | | | | | |
|  | | | | | | | |
| Contenidos | | | | | | | |
| La importancia del control de versiones  Instalación de git  Uso de git como herramienta de líneas de comandos  Uso de git desde el IDE  El desarrollo cómo tarea colaborativa  Repositorios Remotos: GitHub  Flujos de trabajo más comunes: Trunk y GitFlow | | | | | | | |
| Actividades | | | | | | | |
| F.Metod. | | Tipo | Actividad | Objetivos | Actores | | Tiempo |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
| Evaluación | | | | | | | |
| Criterios de Evaluación | | | | | | | % Cal. |
|  | | | | | | |  |
|  | | | | | | |  |
|  | | | | | | |  |
| Actividades de Evaluación | | | | | | | |
| CE | Actividad | | | | | Instrumento | % Cal |
|  |  | | | | |  |  |
|  | Prueba escrita UD 1 | | | | | Solucionario | 40 |

### Unidad Didáctica 6. Diseño orientado a objetos. Diagramas de comportamiento

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Resultados Aprendizaje** | | | **CPPyS** | | **Objetivos Generales** | | |
|  | | |  | |  | | |
| Objetivos de aprendizaje | | | | | | | |
|  | | | | | | | |
| Contenidos | | | | | | | |
| Elementos dinámicos del sistema  Diagramas de casos de uso  Diagramas de secuencia  Diagramas de actividad  Diagramas de estado | | | | | | | |
| Actividades | | | | | | | |
| F.Metod. | | Tipo | Actividad | Objetivos | Actores | | Tiempo |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
| Evaluación | | | | | | | |
| Criterios de Evaluación | | | | | | | % Cal. |
|  | | | | | | |  |
|  | | | | | | |  |
|  | | | | | | |  |
| Actividades de Evaluación | | | | | | | |
| CE | Actividad | | | | | Instrumento | % Cal |
|  |  | | | | |  |  |
|  | Prueba escrita UD 1 | | | | | Solucionario | 40 |

### Unidad Didáctica 7. Diseño orientado a objetos. Diagramas estructurales

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Resultados Aprendizaje | | | CPPyS | | Objetivos Generales | | |
|  | | |  | |  | | |
| Objetivos de aprendizaje | | | | | | | |
|  | | | | | | | |
| Contenidos | | | | | | | |
| Conceptos y ventajas del diseño orientado a objetos  Diagramas de clases.  Elementos de un diagrama de clases: clases, atributos, métodos  Relaciones en un diagrama de clases: herencia, asociación, composición y agregación  Elaboración de un diagrama de clases a partir de requisitos  Escribir código a partir de un diagrama de clases  Herramientas para la realización de diagrams de clases | | | | | | | |
| Actividades | | | | | | | |
| F.Metod. | | Tipo | Actividad | Objetivos | Actores | | Tiempo |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
| Evaluación | | | | | | | |
| Criterios de Evaluación | | | | | | | % Cal. |
|  | | | | | | |  |
|  | | | | | | |  |
|  | | | | | | |  |
| Actividades de Evaluación | | | | | | | |
| CE | Actividad | | | | | Instrumento | % Cal |
|  |  | | | | |  |  |
|  | Prueba escrita UD 1 | | | | | Solucionario | 40 |

### Unidad Didáctica 8. Introducción a las pruebas de software

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Resultados Aprendizaje | | | CPPyS | | Objetivos Generales | | |
|  | | |  | |  | | |
| Objetivos de aprendizaje | | | | | | | |
|  | | | | | | | |
| Contenidos | | | | | | | |
| El problema de la verificación de software  Planificación de pruebas  Tipos de prueba: Caja blanca, caja negra  Pruebas de código: Valores limite, cubrimiento, clases de equivalencia  Calidad de software  Procedimientos y casos de prueba | | | | | | | |
| Actividades | | | | | | | |
| F.Metod. | | Tipo | Actividad | Objetivos | Actores | | Tiempo |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
| Evaluación | | | | | | | |
| Criterios de Evaluación | | | | | | | % Cal. |
|  | | | | | | |  |
|  | | | | | | |  |
|  | | | | | | |  |
| Actividades de Evaluación | | | | | | | |
| CE | Actividad | | | | | Instrumento | % Cal |
|  |  | | | | |  |  |
|  | Prueba escrita UD 1 | | | | | Solucionario | 40 |

### Unidad Didáctica 9. Diseño e implementación de pruebas unitarias

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Resultados Aprendizaje | | | CPPyS | | Objetivos Generales | | |
|  | | |  | |  | | |
| Objetivos de aprendizaje | | | | | | | |
|  | | | | | | | |
| Contenidos | | | | | | | |
| Herramientas para el desarrollo de pruebas unitarias. Junit 5  Incorporar JUnit5 a un proyecto Java  Organización de clases de test  Ejecución de tests desde el IDE  Generación de código de test desde el IDE  Anotaciones de JUnit5  Aserciones en JUnit5  Diseño de test unitarios: el patrón AAA  Tests unitarios con dependencias entre clases: Mocking y Test-Doubles  Incorporar Mockito a un proyecto Java  Creación de mocks de clase y stubs de métodos  Verificando interacciones con Mockito | | | | | | | |
| Actividades | | | | | | | |
| F.Metod. | | Tipo | Actividad | Objetivos | Actores | | Tiempo |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
| Evaluación | | | | | | | |
| Criterios de Evaluación | | | | | | | % Cal. |
|  | | | | | | |  |
|  | | | | | | |  |
|  | | | | | | |  |
| Actividades de Evaluación | | | | | | | |
| CE | Actividad | | | | | Instrumento | % Cal |
|  |  | | | | |  |  |
|  | Prueba escrita UD 1 | | | | | Solucionario | 40 |

### Unidad Didáctica 10. Diseño e implementación de pruebas de integración

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Resultados Aprendizaje | | | CPPyS | | Objetivos Generales | | |
|  | | |  | |  | | |
| Objetivos de aprendizaje | | | | | | | |
|  | | | | | | | |
| Contenidos | | | | | | | |
| Herramientas para el desarrollo de pruebas de integración  Descarga e instalación de Selenium WebDriver  Elementos Web. Identificar y trabajar con objetos en el DOM  Interacciones con el navegador  Implementación de pruebas automáticas de integración con WebDriver y JUnit | | | | | | | |
| Actividades | | | | | | | |
| F.Metod. | | Tipo | Actividad | Objetivos | Actores | | Tiempo |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
| Evaluación | | | | | | | |
| Criterios de Evaluación | | | | | | | % Cal. |
|  | | | | | | |  |
|  | | | | | | |  |
|  | | | | | | |  |
| Actividades de Evaluación | | | | | | | |
| CE | Actividad | | | | | Instrumento | % Cal |
|  |  | | | | |  |  |
|  | Prueba escrita UD 1 | | | | | Solucionario | 40 |

### Unidad Didáctica 11. Documentación de software

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Resultados Aprendizaje | | | CPPyS | | Objetivos Generales | | |
|  | | |  | |  | | |
| Objetivos de aprendizaje | | | | | | | |
|  | | | | | | | |
| Contenidos | | | | | | | |
| La importancia de la documentación  Documentar requisitos de software  Uso de comentarios  Generación de documentación. JavaDoc | | | | | | | |
| Actividades | | | | | | | |
| F.Metod. | | Tipo | Actividad | Objetivos | Actores | | Tiempo |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
| Evaluación | | | | | | | |
| Criterios de Evaluación | | | | | | | % Cal. |
|  | | | | | | |  |
|  | | | | | | |  |
|  | | | | | | |  |
| Actividades de Evaluación | | | | | | | |
| CE | Actividad | | | | | Instrumento | % Cal |
|  |  | | | | |  |  |
|  | Prueba escrita UD 1 | | | | | Solucionario | 40 |

### Unidad Didáctica 12. Código limpio y refactorización de código

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Resultados Aprendizaje | | | CPPyS | | Objetivos Generales | | |
|  | | |  | |  | | |
| Objetivos de aprendizaje | | | | | | | |
|  | | | | | | | |
| Contenidos | | | | | | | |
| Definición de código limpio  Los cinco principios básicos de la programación orientada a objetos. SOLID  Patrones de diseño.  El patrón Modelo-Vista-Controlador  Code Smells  Refactorización y herramientas en el id | | | | | | | |
| Actividades | | | | | | | |
| F.Metod. | | Tipo | Actividad | Objetivos | Actores | | Tiempo |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
|  | |  |  |  |  | |  |
| Evaluación | | | | | | | |
| Criterios de Evaluación | | | | | | | % Cal. |
|  | | | | | | |  |
|  | | | | | | |  |
|  | | | | | | |  |
| Actividades de Evaluación | | | | | | | |
| CE | Actividad | | | | | Instrumento | % Cal |
|  |  | | | | |  |  |
|  | Prueba escrita UD 1 | | | | | Solucionario | 40 |

# Resultados de aprendizaje, criterios de evaluación y unidades didácticas

La Tabla 3 muestra la relación entre las unidades didácticas programadas, los criterios de evaluación y los resultados de aprendizaje del módulo profesional.

|  |  |  |  |
| --- | --- | --- | --- |
| **UD** | **# RA** | **Resultados de aprendizaje** | **Criterios de evaluación** |
| 1 | 1 | Reconoce los elementos y herramientas que intervienen en el desarrollo de un programa informático, analizando sus características y las fases en las que actúan hasta llegar a su puesta en funcionamiento. | a) Se ha reconocido la relación de los programas con los componentes del sistema informático, memoria, procesador, periféricos, entre otros.  b) Se han clasificado los lenguajes de programación.  c) Se han diferenciado los conceptos de código fuente, objeto y ejecutable.  d) Se han reconocido las características de la generación de código intermedio para su ejecución en máquinas virtuales.  f) Se ha diferenciado el funcionamiento de los distintos tipos de traductores de lenguajes ante el código fuente de un programa. |
| 2 | 2 | Evalúa entornos integrados de desarrollo analizando sus características para editar código fuente y generar ejecutable. | a) Se han instalado entornos de desarrollo, propietarios y libres.  b) Se han añadido y eliminado módulos en el entorno de desarrollo.  c) Se ha personalizado y automatizado el entorno de desarrollo.  d) Se ha configurado el sistema de actualización del entorno de desarrollo.  e) Se han generado ejecutables a partir de código fuente de diferentes lenguajes en un mismo entorno de desarrollo.  f) Se han generado ejecutables a partir de un mismo código fuente con varios entornos de desarrollo.  g) Se han identificado las características comunes y específicas de diversos entornos de desarrollo.  h) Se han identificado las funciones más usuales de las herramientas CASE para el desarrollo, prueba y documentación de código. |
| 3 | 1 | Reconoce los elementos y herramientas que intervienen en el desarrollo de un programa informático, analizando sus características y las fases en las que actúan hasta llegar a su puesta en funcionamiento. | g) Se han identificado las fases de desarrollo de una aplicación informática. |
| 4 | 3 | Verifica el funcionamiento de programas diseñando y realizando pruebas. | c) Se han identificado las herramientas de depuración y prueba de aplicaciones ofrecidas por el entorno de desarrollo.  d) Se han utilizado herramientas de depuración para definir puntos de ruptura y seguimiento.  e) Se han utilizado las herramientas de depuración para examinar y modificar el comportamiento de un programa en tiempo de ejecución. |
| 4 | Optimiza código empleando las herramientas disponibles en el entorno de desarrollo | c) Se ha revisado el código fuente usando un analizador de código.  d) Se han identificado las posibilidades de configuración de un analizador de código. |
| 5 | 4 | Optimiza código empleando las herramientas disponibles en el entorno de desarrollo | f) Se ha realizado el control de versiones integrado en el entorno de desarrollo. |
| 6 | 6 | Genera diagramas de comportamiento valorando su importancia en el desarrollo de aplicaciones y empleando las herramientas disponibles en el entorno | a) Se han identificado los distintos tipos de diagramas de comportamiento.  b) Se ha reconocido el significado de los diagramas de casos de uso.  c) Se han interpretado diagramas de interacción.  d) Se han elaborado diagramas de interacción sencillos.  e) Se han interpretado diagramas de estados.  f) Se han planteado diagramas de estados sencillos.  g) Se ha interpretado el significado de diagramas de actividades.  h) Se han elaborado diagramas de actividades sencillos. |
| 7 | 5 | Genera diagramas de clases valorando su importancia en el desarrollo de aplicaciones y empleando las herramientas disponibles en el entorno | a) Se han identificado los conceptos básicos de la programación orientada a objetos.  b) Se ha instalado el módulo del entorno integrado de desarrollo que permite la utilización de diagramas de clases.  c) Se han identificado las herramientas para la elaboración de diagramas de clases.  d) Se ha interpretado el significado de diagramas de clases.  e) Se han trazado diagramas de clases a partir de las especificaciones de las mismas.  f) Se ha generado código a partir de un diagrama de clases.  g) Se ha generado un diagrama de clases mediante ingeniería inversa. |
| 8 | 3 | Verifica el funcionamiento de programas diseñando y realizando pruebas. | a) Se han identificado los diferentes tipos de pruebas.  b) Se han definido casos de prueba.  f) Se ha documentado el plan de pruebas.  k) Se han aplicado normas de calidad a los procedimientos de desarrollo de software.  l) Se han realizado medidas de calidad sobre el software desarrollado. |
| 9 | 3 | Verifica el funcionamiento de programas diseñando y realizando pruebas. | g) Se han efectuado pruebas unitarias de clases y funciones.  i) Se han implementado pruebas automáticas.  j) Se han documentado las incidencias detectadas. |
| 10 | 3 | Verifica el funcionamiento de programas diseñando y realizando pruebas. | h) Se han efectuado pruebas de integración, de sistema y de aceptación.  i) Se han implementado pruebas automáticas.  j) Se han documentado las incidencias detectadas. |
| 11 | 4 | Optimiza código empleando las herramientas disponibles en el entorno de desarrollo | g) Se ha documentado el código fuente mediante comentarios.  h) Se han utilizado herramientas del entorno de desarrollo para documentar los procesos, datos y eventos.  i) Se han utilizado herramientas del entorno de desarrollo para documentar las clases. |
| 12 | 4 | Optimiza código empleando las herramientas disponibles en el entorno de desarrollo | a) Se han identificado los patrones de refactorización más usuales.  b) Se han elaborado las pruebas asociadas a la refactorización.  e) Se han aplicado patrones de refactorización con las herramientas que proporciona el entorno de desarrollo. |

Tabla . Relación RA, CE y UD

# Metodología

## Metodología de aula

### Agrupamientos del alumnado

Los agrupamientos del alumnado obedecerán a criterios pedagogicos

## Instrumentos de seguimiento

Todas las actividades entregables del curso se registrará

**Cuaderno de clase**

A

## Tipos de instrumento de evaluación

**Portafolio**

**Prueba escrita**

**Rubricas**

**Solucionario con criterios de corrección de pruebas escritas u orales**

## Recursos didácticos y materiales

Para la realización de las actividades asociadas al módulo se precisa una dotación material de equipos y materiales. Esta dotación podrá verse modificada durante el curso en función de las necesidades

### Material docente en el aula y equipos técnicos

1. Ordenador para el profesor con conexión a internet
2. Proyector
3. Pizarra blanca y rotuladores
4. Ordenadores con conexión a internet para cada alumno

### Software específico

El siguiente software será necesario para la realización de las actividades del curso:

1. Visual Studio Code
2. Docker
3. IntelliJ Idea
4. Apache Maven
5. Java Development Toolkit
6. Selenium WebDriver

## Usos de las TIC y herramientas de trabajo colaborativo

Dada la naturaleza del ciclo y del del módulo de Entornos de Desarrollo el uso de las TIC es parte fundamental de los procesos de enseñanza-aprendizaje. Se plantea el uso de Moodle Centros para la publicación de materiales, actividades y la gestión de aula virtual. Esta plataforma servirá como punto de encuentro entre profesores y alumnado. Se irán colgando los diferentes materiales propuestos en el aula, como documentos, enlaces a recursos y tareas. Sera en esta plataforma donde se realizará la entrega de las actividades escritas del alumnado, quedando un registro de la fecha de entrega y pudiendo el alumno acceder al resultado de evaluación y la calificación de cada una de ellas. Así mismo se crearán actividades en la Moodle asociadas a las practicas de taller, para que el alumno pueda ver su evaluación y calificación en las mismas. La Moodle nos aporta además un foro de interacción entre los alumnos y el profesor, pudiéndose crear hilos de debate e intercambio de información entre los alumnos, así como creando un canal de comunicación oficial entre el alumno y el profesor.

El trabajo del desarrollador es un trabajo eminente colaborativo por lo que durante el curso se fomentará el uso de herramientas donde poder trabajar de manera colaborativa:

* Diagrams.net para la realización de diagramas colaborativamente
* Trello para la planificación de tareas por parte de equipos
* Notion para la generación de documentación de manera colaborativa
* Miro para la realización de mapas mentales y lluvias de ideas
* Github para tareas de desarrollo colaborativo

## Organización de los espacios

# Evaluación y calificación

Sobre la evaluación podemos ver en el artículo 51 del Real Decreto 1147/2011 lo siguiente:

1. La evaluación del aprendizaje del alumnado de las enseñanzas de formación profesional se realizará por módulos profesionales. Los procesos de evaluación se adecuarán a las adaptaciones metodológicas de las que haya podido ser objeto el alumnado con discapacidad y se garantizará su accesibilidad a las pruebas de evaluación.

2. En todo caso, la evaluación se realizará tomando como referencia los objetivos, expresados en resultados de aprendizaje, y los criterios de evaluación de cada uno de los módulos profesionales, así como los objetivos generales del ciclo formativo o curso de especialización.

Según 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, en su artículo 2 dice:

1. La evaluación de los aprendizajes del alumnado que cursa ciclos formativos será continua y se realizará por módulos profesionales.
2. La aplicación del proceso de evaluación continua del alumnado requerirá, en la modalidad presencial, su asistencia regular a clase y su participación en las actividades programadas para los distintos módulos profesionales del ciclo formativo.
3. La evaluación del alumnado será realizada por el profesorado que imparta cada módulo profesional del ciclo formativo, de acuerdo con los resultados de aprendizaje, los criterios de evaluación y contenidos de cada módulo profesional, así como las competencias y objetivos generales del ciclo formativo asociados a los mismos. […]

## Proceso de evaluación por resultado de aprendizaje

## Ponderación de cada resultado de aprendizaje en la nota del módulo

Los seis resultados de aprendizaje del módulo han sido ponderados en su contribución a la nota del modulo como se detalla en la Tabla 1.

|  |  |  |
| --- | --- | --- |
| **#** | **Resultado de aprendizaje** | **%** |
| 1 | Reconoce los elementos y herramientas que intervienen en el desarrollo de un programa informático, analizando sus características y las fases en las que actúan hasta llegar a su puesta en funcionamiento | 10% |
| 2 | Evalúa entornos integrados de desarrollo analizando sus características para editar código fuente y generar ejecutable | 10% |
| 3 | Verifica el funcionamiento de programas diseñando y realizando pruebas | 33% |
| 4 | Optimiza código empleando las herramientas disponibles en el entorno de desarrollo | 27% |
| 5 | Genera diagramas de clases valorando su importancia en el desarrollo de aplicaciones y empleando las herramientas disponibles en el entorno | 10% |
| 6 | Genera diagramas de comportamiento valorando su importancia en el desarrollo de aplicaciones y empleando las herramientas disponibles en el entorno | 10% |

Tabla . ponderación RA en la nota del módulo

## Ponderación de los criterios de evaluación de cada resultado de aprendizaje

La Tabla 5 muestra la ponderación que tiene cada criterio de evaluación en el resultado de aprendizaje. El tanto por ciento mostrado es

|  |  |  |
| --- | --- | --- |
| **RA** | **Criterio de evaluación** | **%** |
| 1 | a)    Se ha reconocido la relación de los programas con los componentes del sistema informático, memoria, procesador, periféricos, entre otros | 1’5% |
| b)    Se han clasificado los lenguajes de programación | 1’5% |
| c)     Se han diferenciado los conceptos de código fuente, objeto y ejecutable | 1% |
| d)    Se han reconocido las características de la generación de código intermedio para su ejecución en máquinas virtuales | 1’5% |
| e) Se ha evaluado la funcionalidad ofrecida por las herramientas utilizadas en programación | 1’5% |
| f) Se ha diferenciado el funcionamiento de los distintos tipos de traductores de lenguajes ante el código fuente de un programa | 1% |
| g) Se han identificado las fases de desarrollo de una aplicación informática. | 2% |
| Total RA 1 | | 10% |
| 2 | a) Se han instalado entornos de desarrollo, propietarios y libres. | 2% |
| b) Se han añadido y eliminado módulos en el entorno de desarrollo. | 1’5% |
| c) Se ha personalizado y automatizado el entorno de desarrollo | 1’5% |
| d) Se ha configurado el sistema de actualización del entorno de desarrollo. | 1% |
| e) Se han generado ejecutables a partir de código fuente de diferentes lenguajes en un mismo entorno de desarrollo. | 1% |
| f) Se han generado ejecutables a partir de un mismo código fuente con varios entornos de desarrollo. | 1% |
| g) Se han identificado las características comunes y específicas de diversos entornos de desarrollo. | 1% |
| h) Se han identificado las funciones más usuales de las herramientas CASE para el desarrollo, prueba y documentación de código. | 1% |
| Total RA 2 | | 10% |
| 3 | a) Se han identificado los diferentes tipos de pruebas. | 1% |
| b) Se han definido casos de prueba. | 1% |
| c) Se han identificado las herramientas de depuración y prueba de aplicaciones ofrecidas por el entorno de desarrollo. | 1% |
| d) Se han utilizado herramientas de depuración para definir puntos de ruptura y seguimiento. | 1% |
| e) Se han utilizado las herramientas de depuración para examinar y modificar el comportamiento de un programa en tiempo de ejecución. | 1% |
| f) Se ha documentado el plan de pruebas. | 1% |
| h) Se han efectuado pruebas unitarias de clases y funciones. | 9% |
| g) Se han efectuado pruebas de integración, de sistema y de aceptación. | 7% |
| i) Se han implementado pruebas automáticas. | 10% |
| j) Se han documentado las incidencias detectadas. | 1% |
| k) Se han aplicado normas de calidad a los procedimientos de desarrollo de software. | 1% |
| l) Se han realizado medidas de calidad sobre el software desarrollado. | 1% |
| Total RA 3 | | 33% |
| 4 | a) Se han identificado los patrones de refactorización más usuales. | 3’5% |
| b) Se han elaborado las pruebas asociadas a la refactorización. | 3’5% |
| c) Se ha revisado el código fuente usando un analizador de código. | 1% |
| d) Se han identificado las posibilidades de configuración de un analizador de código. | 1% |
| e) Se han aplicado patrones de refactorización con las herramientas que proporciona el entorno de desarrollo. | 3% |
| f) Se ha realizado el control de versiones integrado en el entorno de desarrollo. | 10% |
| g) Se ha documentado el código fuente mediante comentarios. | 1% |
| h) Se han utilizado herramientas del entorno de desarrollo para documentar los procesos, datos y eventos. | 2% |
| i) Se han utilizado herramientas del entorno de desarrollo para documentar las clases. | 2% |
| Total RA 4 | | 27% |
| 5 | a) Se han identificado los conceptos básicos de la programación orientada a objetos. | 1% |
| b) Se ha instalado el módulo del entorno integrado de desarrollo que permite la utilización de diagramas de clases. | 1% |
| c) Se han identificado las herramientas para la elaboración de diagramas de clases. | 1% |
| d) Se ha interpretado el significado de diagramas de clases. | 2% |
| e) Se han trazado diagramas de clases a partir de las especificaciones de las mismas. | 2% |
| f) Se ha generado código a partir de un diagrama de clases. | 1% |
| g) Se ha generado un diagrama de clases mediante ingeniería inversa. | 2% |
| Total RA 5 | | 10% |
| 6 | a) Se han identificado los distintos tipos de diagramas de comportamiento. | 1,25% |
| b) Se ha reconocido el significado de los diagramas de casos de uso. | 1,25% |
| c) Se han interpretado diagramas de interacción. | 1,25% |
| d) Se han elaborado diagramas de interacción sencillos. | 1,25% |
| e) Se han interpretado diagramas de estados. | 1,25% |
| f) Se han planteado diagramas de estados sencillos. | 1,25% |
| g) Se ha interpretado el significado de diagramas de actividades. | 1,25% |
| h) Se han elaborado diagramas de actividades sencillos. | 1,25% |
| Total RA 6 | | 10% |

Tabla . Ponderación CEs por cada RA

## Ponderación de la relación entre resultados de aprendizaje, criterios de evaluación y actividades evaluables

## Mecanismos de recuperación y mejora de la calificación

Cuando el alumno o alumna no logre la superación de las deficiencias y fallos detectados y haya obtenido una valoración insuficiente en cualquiera de los conceptos evaluados, se establecerán unas actividades específicas de recuperación. Estas actividades estarán en función de los contenidos a reforzar o recuperar y las capacidades implicadas. Estas actividades pueden consistir en: resolución de cuestionarios, análisis y resolución de casos prácticos, trabajos individuales, realización de exposiciones, pruebas escritas, etc.

De la misma manera, y teniendo en cuenta los diferentes ritmos de aprendizaje que podemos tener en el aula, podemos tener alumnos y alumnas que quieran mejorar la calificación obtenida. El profesor o profesora debe incentivar este tipo de iniciativas y potenciarlas, como forma de mostrar interés por el aprendizaje. Para ello se establecerán actividades de mejora de la calificación que podrán ser del mismo tipo que las anteriores, pero con cierto grado de profundidad, según las capacidades del alumnado.

# Autoevaluación

## De la programación didáctica

## De la labor docente

Para realizar la evaluación de la práctica docente y del proceso de enseñanza-aprendizaje se realizarán las siguientes actuaciones durante el curso escolar:

### Evaluación por parte del alumnado

Durante el final de cada trimestre se distribuirán al alumnado unos formularios anónimos donde puedan valorar la actividad del profesor, la metodología, la adecuación de los contenidos o los instrumentos de evaluación. Así mismo, estos formularios darán la posibilidad de realizar aportaciones a modo de observaciones para mejorar y adecuar el proceso de resultado aprendizaje

### Evaluación por parte del propio docente

Además, al final de cada trimestre el docente también rellenará un formulario con los indicadores de consecución de diferentes aspectos importantes de la práctica docente y su nivel de consecución:

1. Metodología docente y adecuación del contenido
2. Adecuación de la evaluación
3. Atención a la diversidad
4. Fomento de la empleabilidad del alumnado
5. Desarrollo de competencias transversales

# Atención a la diversidad

## Atención a los alumnos con necesidades específicas de apoyo educativo

## Atención a los diferentes ritmos de aprendizaje

# Actividades complementarias y extraescolares

Durante el segundo trimestre del curso la realización de un taller sobre calidad y pruebas automáticas de software en la empresa de desarrollo Freepik. Esta startup malagueña tiene un fuerte componente tecnológico y colabora habitualmente con centros públicos que imparten enseñanzas de Formación Profesional en la provincia de Málaga. Dado que la finalidad de esta actividad es la de resaltar la importancia de alcanzar los resultados de aprendizaje del módulo profesional, la posibilidad de poder descubrir que ciertas competencias a las que contribuye el módulo son muy demandadas por empresas del sector productivo relacionado con el ciclo de Técnico Superior en Desarrollo de Aplicaciones Web.

# Bibliografía

Atlassian. (s.f.). *Getting Git Right*. Obtenido de Bitbucket: https://www.atlassian.com/git/tutorials

Bechtold, S. (s.f.). *JUnit5*. Obtenido de https://junit.org/junit5/

Casado Iglesias, C. (2012). *Entornos de Desarrollo.* Paracuellos del Jarama: RA-MA.

*Dangit, Git!?!* (s.f.). Obtenido de https://dangitgit.com/es

Dudler, R. (01 de 06 de 2023). *git - la guía sencilla*. Obtenido de https://rogerdudler.github.io/git-guide/index.es.html

Faber, S. (s.f.). *Mockito*. Obtenido de https://javadoc.io/doc/org.mockito/mockito-core/latest/org/mockito/Mockito.html

Foundation, E. (s.f.). *Eclipse Documentation*. Obtenido de https://www.eclipse.org/documentation/

Jetbrains. (s.f.). *IntelliJ IDEA*. Obtenido de https://www.jetbrains.com/help/idea/getting-started.html

Martin, R. C. (2008). *Clean code: a handbook of agile software craftsmanship.* Upper Saddle River, NJ: Prentice Hall.

Microsoft. (s.f.). *Visual Studio Code Docs*. Obtenido de https://code.visualstudio.com/docs

Oracle. (02 de 2004). *Javadoc Tool*. Obtenido de https://www.oracle.com/es/technical-resources/articles/java/javadoc-tool.html

Pressman, R. S. (2010). *Ingenieria del software. Un enfoque práctico.* Ciudad de México: Mc Graw Hill.

Selenium. (7 de 12 de 2021). *WebDriver documentation*. Obtenido de https://www.selenium.dev/documentation/webdriver/

Sourcemaking. (s.f.). *Refactoring*. Obtenido de https://sourcemaking.com/refactoring

Sourcemaking. (s.f.). *UML*. Obtenido de https://sourcemaking.com/uml

# Anexos

## Anexo I. Indicadores de evaluación de la actividad docente

**Indicador 1**

Diseño estrategias y programo actividades variadas y creativas que se ajustan de manera precisa a los objetivos didácticos planteados, abordando diferentes tipos de contenidos y adaptándolas de manera efectiva a las características individuales de los alumnos.

**Indicador 2**

Utilizo métodos de evaluación formativa y sumativa de manera equilibrada, diseñando instrumentos de evaluación adecuados que permiten medir de forma precisa el logro de los objetivos didácticos establecidos, considerando tanto el progreso individual de los estudiantes como el desempeño grupal.

**Indicador 3**

Implemento estrategias de enseñanza diferenciadas que atienden a la diversidad de los alumnos, adaptando los contenidos y las actividades de acuerdo con las necesidades, estilos de aprendizaje y ritmos de cada estudiante, garantizando así un entorno inclusivo y propicio para el aprendizaje de todos.

**Indicador 4**

Fomento el desarrollo de habilidades socioemocionales y competencias transversales en mis alumnos, integrando en las estrategias y actividades didácticas aspectos como el trabajo en equipo, la resolución de problemas, la comunicación efectiva y la reflexión crítica, con el objetivo de prepararlos de manera integral para enfrentar los desafíos del siglo XXI.

**Indicador 5**

Selecciono y estructuro los contenidos del curso de manera que estén alineados con las demandas y tendencias actuales del mercado laboral, integrando conceptos, habilidades y conocimientos que son relevantes y necesarios para que los estudiantes puedan enfrentar con éxito los desafíos y requerimientos del entorno profesional.