

**Programación del Módulo Profesional**

**“Entornos de Desarrollo”**

**Del Ciclo Formativo de Grado Superior**

**“Desarrollo de Aplicaciones Web”**

## Contenido

<b>1</b>	<b>OBJETIVOS GENERALES DEL MÓDULO PROFESIONAL.....</b>	<b>3</b>
<b>2</b>	<b>UNIDADES DE COMPETENCIA ASOCIADAS AL MÓDULO PROFESIONAL .....</b>	<b>4</b>
<b>3</b>	<b>CONTENIDOS Y SECUENCIACIÓN.....</b>	<b>4</b>
<b>4</b>	<b>CONTENIDOS MÍNIMOS.....</b>	<b>8</b>
<b>5</b>	<b>METODOLOGÍA Y ESTRATEGIAS DIDÁCTICAS .....</b>	<b>8</b>
<b>6</b>	<b>RESULTADOS DE APRENDIZAJE Y CRITERIOS DE EVALUACIÓN.....</b>	<b>8</b>
<b>7</b>	<b>PROCEDIMIENTOS E INSTRUMENTOS DE EVALUACIÓN.....</b>	<b>10</b>
<b>8</b>	<b>LOS CRITERIOS DE CALIFICACIÓN .....</b>	<b>11</b>
<b>9</b>	<b>RECURSOS DIDÁCTICOS .....</b>	<b>13</b>
<b>10</b>	<b>ATENCIÓN A LA DIVERSIDAD Y ADAPTACIONES CURRICULARES.....</b>	<b>15</b>
<b>11</b>	<b>UTILIZACIÓN DE LAS TIC.....</b>	<b>15</b>
<b>12</b>	<b>ACTIVIDADES DE RECUPERACIÓN DE MÓDULOS PROFESIONALES PENDIENTES.....</b>	<b>16</b>
<b>13</b>	<b>PROCEDIMIENTO PARA QUE EL ALUMNADO Y SUS FAMILIAS CONOZCAN LOS OBJETIVOS, LOS CONTENIDOS, LOS CRITERIOS DE EVALUACIÓN, LOS CRITERIOS DE CALIFICACIÓN, LOS PROCEDIMIENTOS Y LOS INSTRUMENTOS DE EVALUACIÓN. ....</b>	<b>16</b>
<b>14</b>	<b>ACTIVIDADES COMPLEMENTARIAS Y EXTRAESCOLARES.....</b>	<b>17</b>
<b>15</b>	<b>REFERENCIAS.....</b>	<b>17</b>
<b>16</b>	<b>INDICADORES PARA LA EVALUACIÓN DE LA PRÁCTICA DOCENTE.....</b>	<b>17</b>
16.1	ENCUESTA PARA EVALUAR AL PROFESOR.....	18
16.2	AUTOEVALUACIÓN DE LA PRÁCTICA DOCENTE.....	19
<b>17</b>	<b>ADAPTACIONES CON MOTIVO DEL ESCENARIO EN FASE II (COVID-19). ....</b>	<b>19</b>

## 1 Objetivos generales del módulo profesional

Los objetivos generales del ciclo formativo, así como las competencias profesionales, personales y sociales del título, se establecen en el Real Decreto 686/2010.

En concreto, este módulo contribuye a los siguientes objetivos:

- Instalar y configurar módulos y complementos, evaluando su funcionalidad, para gestionar entornos de desarrollo.
- Seleccionar y emplear lenguajes, herramientas y librerías, interpretando las especificaciones para desarrollar aplicaciones multiplataforma con acceso a bases de datos.
- Emplear herramientas de desarrollo, lenguajes y componentes visuales, siguiendo las especificaciones y verificando interactividad y usabilidad, para desarrollar interfaces gráficos de usuario en aplicaciones multiplataforma.
- Seleccionar y emplear técnicas, motores y entornos de desarrollo, evaluando sus posibilidades, para participar en el desarrollo de juegos y aplicaciones en el ámbito del entretenimiento.
- Seleccionar y emplear técnicas, lenguajes y entornos de desarrollo, evaluando sus posibilidades, para desarrollar aplicaciones en teléfonos, PDA y otros dispositivos móviles.
- Verificar los componentes software desarrollados, analizando las especificaciones, para completar un plan de pruebas.

Además, contribuye a la adquisición de las siguientes competencias:

- Gestionar entornos de desarrollo adaptando su configuración en cada caso para permitir el desarrollo y despliegue de aplicaciones.
- Desarrollar aplicaciones implementando un sistema completo de formularios e informes que permitan gestionar de forma integral la información almacenada.
- Desarrollar interfaces gráficos de usuario interactivos y con la usabilidad adecuada, empleando componentes visuales estándar o implementando componentes visuales específicos.
- Participar en el desarrollo de juegos y aplicaciones en el ámbito del entretenimiento y la educación empleando técnicas, motores y entornos de desarrollo específicos.
- Desarrollar aplicaciones para teléfonos, PDA y otros dispositivos móviles empleando técnicas y entornos de desarrollo específicos.

- Realizar planes de pruebas verificando el funcionamiento de los componentes software desarrollados, según las especificaciones.
- Establecer vías eficaces de relación profesional y comunicación con sus superiores, compañeros y subordinados, respetando la autonomía y competencias de las distintas personas.
- Gestionar su carrera profesional, analizando las oportunidades de empleo, autoempleo y de aprendizaje.
- 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 Unidades de competencia asociadas al módulo profesional

El Anexo V B del Real Decreto 686/2010 no asocia, para su acreditación, al módulo profesional ninguna unidad de competencia.

## 3 Contenidos y secuenciación

Los contenidos del módulo vienen especificados en el DECRETO 1/2011, de 13 de enero, por el que se establece para la Comunidad de Madrid el currículo del ciclo formativo de grado superior correspondiente al título de Técnico Superior en Desarrollo de Aplicaciones Web.

La duración del módulo a lo largo del año es de 90 horas, impartido en 3 horas semanales. En el presente curso las horas reales disponibles para docencia serán 90 (30 en el 1<sup>er</sup> trimestre + 28 en el 2<sup>o</sup> + 32 en el 3<sup>o</sup>):

Se estiman 10 horas para evaluación, por lo que se prevé la siguiente distribución de tiempos:

<b>PRIMERA EVALUACIÓN</b>	<b>30 horas</b>
Presentación del módulo	1
U.T. 1. El programa informático.	6
U.T. 2. La ingeniería de software.	7
U.T. 3. Del código fuente al ejecutable.	10
U.T. 4. Entornos de desarrollo: instalación y uso.	3
Evaluación y recuperación	3
<b>SEGUNDA EVALUACIÓN</b>	<b>28 horas</b>
U.T. 4. Entornos de desarrollo: instalación y uso. (cont.)	6
U.T. 5. Gestión de la configuración.	8

U.T. 6. Introducción al UML.	2
U.T. 7. UML y los diagramas de clases.	5
U.T. 8. UML y los diagramas de comportamiento.	4
Evaluación y recuperación	3
<b>TERCERA EVALUACIÓN</b>	<b>32 horas</b>
U.T. 8. UML y los diagramas de comportamiento.	8
U.T. 9. Pruebas del software.	8
U.T. 10. Automatización de pruebas y depuración de programas	7
U.T. 11. Refactorización y documentación	5
Evaluación	2
<b>EVALUACIÓN FINAL</b>	<b>2</b>

A continuación se especificarán los contenidos de cada unidad de trabajo.

<b>U.T. 1. El programa informático</b>	<b>6 horas</b>
<b>Contenidos</b>	
<ul style="list-style-type: none"> <li>- Concepto de programa informático             <ul style="list-style-type: none"> <li>• Concepto</li> <li>• Elementos de un programa</li> </ul> </li> <li>- Componentes de un ordenador             <ul style="list-style-type: none"> <li>• Hardware vs. software</li> <li>• Máquinas virtuales</li> </ul> </li> <li>- <i>Estructura funcional de un ordenador</i></li> <li>- Los lenguajes de programación             <ul style="list-style-type: none"> <li>• Niveles de los lenguajes de programación</li> <li>• Ensambladores, compiladores e intérpretes</li> <li>• Áreas de aplicación de los diferentes lenguajes</li> <li>• Clasificación de los lenguajes de programación</li> </ul> </li> </ul>	

<b>U.T. 2. La ingeniería de software</b>	<b>7 horas</b>
<b>Contenidos</b>	
<ul style="list-style-type: none"> <li>- <i>La ingeniería de software</i> <ul style="list-style-type: none"> <li>• <i>Concepto de ingeniería de software</i></li> <li>• <i>Parámetros de calidad del software</i></li> <li>• <i>Reutilización del software</i></li> </ul> </li> <li>- Metodología de desarrollo             <ul style="list-style-type: none"> <li>• Ciclos de vida</li> <li>• <i>Metodologías de desarrollo Ágil</i></li> </ul> </li> <li>- <i>Herramientas CASE</i></li> </ul>	

<b>U.T. 3. Del código fuente al ejecutable</b>	<b>10 horas</b>
<b>Contenidos</b>	
<ul style="list-style-type: none"> <li>- Fichero fuente, objeto y ejecutable</li> </ul>	

- Librerías estáticas y dinámicas
- Fases de la compilación
  - El preprocesado
  - El compilado
  - El ensamblado
  - El enlazado
- Construcción de software: tratamiento de dependencias

<b>U.T. 4. Entornos de desarrollo: instalación y uso</b>	<b>9 horas</b>
<b>Contenidos</b>	
<ul style="list-style-type: none"> <li>- Concepto de entorno de desarrollo</li> <li>- Funciones de un entorno de desarrollo</li> <li>- Entornos de desarrollo libres y comerciales</li> <li>- Instalación de un entorno de desarrollo</li> <li>- Uso básico de un entorno de desarrollo</li> <li>- Gestión de módulos y <i>plug-ins</i></li> </ul>	

<b>U.T. 5. Gestión de la configuración</b>	<b>8 horas</b>
<b>Contenidos</b>	
<ul style="list-style-type: none"> <li>- <i>Necesidad del Software de Gestión de la Configuración</i></li> <li>- La arquitectura cliente/servidor del SCM. El repositorio                     <ul style="list-style-type: none"> <li>• Copias de trabajo y manipulación de archivos</li> </ul> </li> <li>- El control de versiones                     <ul style="list-style-type: none"> <li>• Evolución de los ficheros</li> <li>• Tags</li> <li>• Propagación de cambios y fusión de variantes</li> </ul> </li> <li>- <i>El control de la configuración</i></li> <li>- Utilización práctica de un SCM                     <ul style="list-style-type: none"> <li>• Comandos principales</li> <li>• La gestión de la configuración en los IDE</li> </ul> </li> </ul>	

<b>U.T. 6. Introducción al UML</b>	<b>2 horas</b>
<b>Contenidos</b>	
<ul style="list-style-type: none"> <li>- <i>Representación de problemas del mundo real: la abstracción</i></li> <li>- El paradigma de la orientación a objetos</li> <li>- <i>El Lenguaje Unificado de Modelado</i> <ul style="list-style-type: none"> <li>• <i>Concepto de UML</i></li> <li>• <i>Diagramas en UML</i></li> <li>• <i>Utilización de UML en la metodología RUP</i></li> </ul> </li> </ul>	

<b>U.T. 7. UML y los diagramas de clases</b>	<b>5 horas</b>
<b>Contenidos</b>	
<ul style="list-style-type: none"> <li>- Propósito de los diagramas de clases</li> <li>- Notación de los diagramas de clases                     <ul style="list-style-type: none"> <li>• Clases, atributos y operaciones</li> <li>• Relaciones entre clases</li> <li>• Clases abstractas e interfaces</li> <li>• Objetos</li> </ul> </li> </ul>	

<ul style="list-style-type: none"> <li>• Aspectos avanzados</li> <li>- Los diagramas de paquetes</li> <li>- Los diagramas de clases en los IDE             <ul style="list-style-type: none"> <li>• Elaboración de diagramas de clases en un IDE</li> <li>• Generación automática de código</li> <li>• Ingeniería inversa</li> </ul> </li> </ul>
--

<b>U.T. 8. UML y los diagramas de comportamiento</b>	<b>12 horas</b>
<b>Contenidos</b>	
<ul style="list-style-type: none"> <li>- Diagramas de casos de uso             <ul style="list-style-type: none"> <li>• Concepto de caso de uso</li> <li>• Notación de los diagramas de casos de uso</li> </ul> </li> <li>- Diagramas de interacción             <ul style="list-style-type: none"> <li>• Propósito de los diagramas de interacción</li> <li>• Notación de los diagramas de secuencia</li> <li>• Notación de los diagramas de colaboración</li> </ul> </li> <li>- Diagramas de estado             <ul style="list-style-type: none"> <li>• Propósito de los diagramas de estado</li> <li>• Notación de los diagramas de estado</li> </ul> </li> <li>- Diagramas de actividades             <ul style="list-style-type: none"> <li>• Propósito de los diagramas de actividades</li> <li>• Notación de los diagramas de actividades</li> </ul> </li> <li>- Los diagramas de comportamiento en los IDE</li> </ul>	

<b>U.T. 9. Pruebas del software</b>	<b>8 horas</b>
<b>Contenidos</b>	
<ul style="list-style-type: none"> <li>- Pruebas en el proceso de desarrollo de software             <ul style="list-style-type: none"> <li>• Planificación de pruebas a lo largo del ciclo de desarrollo</li> <li>• Técnicas comunes de pruebas de unidad</li> <li>• Estrategias para las pruebas de integración</li> <li>• Técnicas comunes de pruebas de sistema</li> </ul> </li> <li>- Documentación de las pruebas</li> </ul>	

<b>U.T. 10. Automatización de pruebas y depuración de programas</b>	<b>7 horas</b>
<b>Contenidos</b>	
<ul style="list-style-type: none"> <li>- Herramientas integradas en los IDE para realización de pruebas             <ul style="list-style-type: none"> <li>• Automatización de pruebas unitarias</li> </ul> </li> <li>- La depuración de programas             <ul style="list-style-type: none"> <li>• Concepto de depuración</li> <li>• Puntos de ruptura y examinadores de variables</li> <li>• La depuración en los IDE</li> </ul> </li> </ul>	

<b>U.T. 11. Refactorización y documentación</b>	<b>5 horas</b>
<b>Contenidos</b>	
<ul style="list-style-type: none"> <li>- Refactorización             <ul style="list-style-type: none"> <li>• Concepto de refactorización</li> <li>• Patrones de refactorización más usuales</li> <li>• Metodologías TDD</li> </ul> </li> </ul>	

- La refactorización en los IDE
- Documentación del software
  - Documentación interna y externa
  - La documentación en los IDE
- Analizadores de código

#### **4 Contenidos mínimos**

Los contenidos mínimos que deben alcanzar los alumnos en el módulo tienen su referencia en el Real Decreto 686/2010, y en concreto en los criterios de evaluación de los correspondientes resultados de aprendizaje, que marcan los niveles de consecución aceptable de dichos resultados.

Los alumnos deben ser capaces de resolver cuestiones teóricas y prácticas que indiquen que han conseguido los resultados de aprendizaje. Para ello deben demostrar que han adquirido los conocimientos, destrezas y habilidades desarrolladas en cada unidad de trabajo, sobre los cuales se sustentarán esos resultados. Los contenidos indicados en letra cursiva no se considerarán contenidos mínimos del módulo profesional; el resto sí se considerarán mínimos.

#### **5 Metodología y estrategias didácticas**

La metodología buscará conseguir un aprendizaje significativo, en el que el alumno pueda ir construyendo sus aprendizajes sobre la base de sus conocimientos anteriores. Se utilizará una metodología activa, en la que los discentes sean protagonistas de su propio proceso de enseñanza/aprendizaje.

#### **6 Resultados de aprendizaje y Criterios de evaluación**

Son los que recoge el Real Decreto 686/2010, por el que se establece el título y se fijan sus enseñanzas mínimas. Se enumeran a continuación los resultados 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.**

*Criterios de evaluación:*

- a) Se ha reconocido la relación de los programas con los componentes del sistema informático: memoria, procesador, periféricos, entre otros.
- b) Se han identificado las fases de desarrollo de una aplicación informática.
- 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.
- e) Se han clasificado los lenguajes de programación.
- f) Se ha evaluado la funcionalidad ofrecida por las herramientas utilizadas en programación.



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

*Criterios de evaluación:*

- 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.

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

*Criterios de evaluación:*

- a) Se han identificado los diferentes tipos de pruebas.
- b) Se han definido casos de prueba.
- 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.
- f) Se han efectuado pruebas unitarias de clases y funciones.
- g) Se han implementado pruebas automáticas.
- h) Se han documentado las incidencias detectadas.

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

*Criterios de evaluación:*

- a) Se han identificado los patrones de refactorización más usuales.
- b) Se han elaborado las pruebas asociadas a la refactorización.
- 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.
- e) Se han aplicado patrones de refactorización con las herramientas que proporciona el entorno de desarrollo.
- f) Se ha realizado el control de versiones integrado en el entorno de desarrollo.
- g) 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:*

- 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.

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

*Criterios de evaluación:*

- 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 ha interpretado el significado de diagramas de actividades.
- f) Se han elaborado diagramas de actividades sencillos.
- g) Se han interpretado diagramas de estados.
- h) Se han planteado diagramas de estados sencillos.

## **7 Procedimientos e instrumentos de evaluación**

En consonancia con el art. 24.2 de la Orden 2694/2009, se celebrará una sesión de evaluación por cada trimestre de formación en el centro educativo; la última, tendrá la consideración de evaluación final ordinaria. Las fechas de las mismas son las fijadas por el Claustro de profesores al inicio de curso (con las modificaciones que a este respecto pudieran ser aprobadas posteriormente, por este mismo órgano).

La evaluación se realizará agrupando las unidades temáticas por evaluaciones. En el primer curso se considerarán tres evaluaciones parciales más la final ordinaria. Para cada módulo, la calificación de la tercera evaluación parcial de primer curso no aparecerá en ningún acta de evaluación parcial, puesto que el acta que se publique será la de la evaluación final ordinaria; no obstante, dicha evaluación constará como otra más a efectos de la calificación del módulo formativo.

Los instrumentos de evaluación serán:

- **Actividades de enseñanza/aprendizaje:** Se realizará un control durante cada evaluación para realizar un seguimiento de los procesos de enseñanza/aprendizaje.

En cada evaluación, además, los alumnos realizarán un trabajo que habrán de entregar el profesor:

- El de la primera evaluación versará sobre el lenguaje de programación Python, el más popular según el índice PYPL. Lo realizarán en equipos de tres. Utilizarán la

herramienta Collaboratory de Google para redactar un cuaderno describiendo algunos aspectos significativos del lenguaje.

- En la segunda evaluación los alumnos, también en grupos de tres, deberán realizar un trabajo acerca de alguna cuestión no tratada en clase pero relacionada con las unidades didácticas de la asignatura. El texto habrá de ser original, y tener una extensión mínima de 5 folios, con interlineado 1,5 y tamaño de fuente 12. Los autores deberán hacer una exposición del tema ante sus compañeros de un máximo de 15 minutos de duración, para lo que tendrán que preparar una presentación. A continuación, se les podrán realizar algunas preguntas sobre la materia.

Posibles contenidos a tratar en los trabajos serán:

- Técnicas de priorización de requisitos del software
  - Herramientas software para Scrum
  - Kanban
  - Lean Software Development
  - Maven
  - Gradle
  - Netbeans vs Eclipse
  - Mercurial
  - Diagramas UML de despliegue, de componentes, de estructuras compuestas, de tiempo, generales de interacción.
  - Plugins de Eclipse para UML
- En la tercera evaluación deberán realizar diferentes tipos de diagramas UML utilizando la herramienta StarUML.
- **Prueba específica de evaluación:** Será un examen que abarcará todos los contenidos impartidos durante la evaluación; se realizará cerca del final de la misma.
  - **Actitud:** Se tomará en consideración el interés que el alumno muestre por este módulo profesional y sus tecnologías asociadas. Un aspecto importante a tener en cuenta será la asistencia del alumno a las clases.

## 8 Los criterios de calificación

La evaluación del módulo se hará conforme a la siguiente tabla.

Convocatoria	Peso Actividades de enseñanza/aprendizaje	Peso prueba específica de evaluación	Recuperación	Recuperación final	Nota Final del módulo
Evaluación 1	Control: 20% Trabajo: 20%	60%	Sí	Sí	Media aritmética de las evaluaciones
Evaluación 2	Control: 20% Trabajo: 20%	60%	Sí	Sí	
Evaluación 3	Control: 20% Trabajo: 20%	60%	No	Sí	

La calificación de cada evaluación parcial se obtendrá del siguiente modo:

- **Actividades de enseñanza/aprendizaje:** Serán evaluadas con un valor numérico comprendido entre 0 y 10. Si el alumno, sin causa justificada, no realiza un control, se le valorará con un 0; si es por causa justificada, el control no le computará, por lo que su nota de evaluación se obtendrá de los otros instrumentos de evaluación, adaptando proporcionalmente sus pesos.
- **Prueba específica de evaluación:** Tendrá una nota numérica entre 0 y 10. Si el alumno no realiza la prueba, constará como “No Evaluado” en la evaluación; si es por causa justificada, la podrá realizar el día que se le indique tras su reincorporación al centro.

Como resultado de la aplicación de los porcentajes indicados anteriormente se obtendrá una nota con una precisión de un decimal, que se redondeará al entero más cercano; no obstante, en los futuros cálculos en los que se utilicen estos resultados del alumno, se empleará la nota previa al redondeo.

A efectos de redondeo, los decimales inferiores a 0,5 se redondearán al entero más bajo. Los iguales o superiores a 0,5 al entero más alto. Esta regla tiene dos excepciones: la franja entre 4 y 5 se redondeará siempre a 4 y las notas inferiores a 1 se redondearán a 1.

Las faltas de ortografía cometidas en todo tipo de escritos (ejercicios, prácticas, exámenes, etc.) se penalizarán, hasta un máximo de un punto, con arreglo al siguiente baremo:

- Cada error en el empleo de las grafías: 0,2 puntos.
- Cada error de acentuación o puntuación: 0,1 puntos

Será necesario, para que el alumno pueda aprobar la evaluación, que haya obtenido en la prueba específica al menos un 4. En caso contrario su nota máxima será un 4,0 independientemente del resultado de la ponderación anterior.

Los alumnos que no alcancen evaluación positiva en alguna de las dos primeras evaluaciones habrán de presentarse a la prueba de recuperación que se convoque durante la siguiente evaluación para examinarse de todos los contenidos de la evaluación pendiente de aprobar. La nota final de esa parte del módulo será la obtenida en la prueba (con un decimal de precisión).

Llegada la convocatoria ordinaria de junio, los alumnos que no hayan alcanzado evaluación positiva en alguna de las evaluaciones habrán de presentarse a una prueba de recuperación final para examinarse de todos los contenidos de la evaluación pendiente de aprobar. La nota final de esa parte del módulo será la obtenida en la prueba de junio (con un decimal de precisión). Si algún alumno debiera presentarse en la convocatoria ordinaria de junio a dos o las tres evaluaciones del módulo, el procedimiento será el ya citado, aplicado a cada una por separado.

La calificación final del módulo en la convocatoria ordinaria consistirá en la media aritmética de las calificaciones obtenidas en las tres evaluaciones parciales, siempre que estén todas aprobadas; en caso contrario tendrá como valor máximo 4.

### **Convocatoria extraordinaria**

Los alumnos que acudan a convocatoria extraordinaria realizarán un examen. Es de aplicación el art. 25.3 de la Orden 2694/2009, que, sobre este asunto dice que *«se llevarán a cabo actividades de recuperación; cuando ello no sea posible se programará una prueba que se convocará antes de que se celebre la sesión de evaluación extraordinaria. Dicha prueba tendrá como referentes los criterios de evaluación mínimos incluidos en las programaciones didácticas y el informe previsto en el apartado 1.c de este artículo»* (nótese que este informe es el que se entrega a cada alumno que tiene módulos pendientes tras la evaluación final ordinaria).

El examen será calificado con un valor numérico comprendido entre 0 y 10, y se considerará aprobado si este valor es mayor o igual a 5.

Se aplicarán los mismos criterios ortográficos y de redondeo establecidos anteriormente.

## **9 Recursos didácticos**

No se usará libro de texto, aportando el profesor parte de los apuntes y recomendando el uso de algunos libros de los citados a continuación, así como de determinadas páginas de Internet.

### **Bibliografía**

- CASADO, C. (2012). *“Entornos de desarrollo”*. Madrid. Ra-Ma

- DOAR, M. (2005). *“Practical Development Environments”*. O’Reilly Media
- *“Workbench User Guide”*, Contenidos de ayuda de Eclipse.
- *“Java Development User Guide”*, Contenidos de ayuda de Eclipse.
- WILLIAMS, L. (2006): *White-Box Testing*.
- Arquitectura e Integración de Sistemas Software (2012-13): *Tema 10. Pruebas de Integración*. Departamento de Lenguajes y Sistemas Informáticos. Universidad de Sevilla.
- DEXTER, M. (2008): *Eclipse and Java: Using the Debugger*.
- FOWLER, M. (1999): *Refactorización: Mejorando el diseño de código existente*. Addison-Wesley
- KICILLOF, N.: *Desarrollo guiado por pruebas y refactorización*.
- CID MÍLLARA, B. y otros: *Refactorización en Eclipse*.
- BARNES, D. y KÖLLING, M. (2012): *Objects First with Java: A Practical Introduction Using BlueJ. 5th Edition*. Prentice Hall / Pearson Education.
- MASON, M. (2010): *Pragmatic Guide to Subversion*. The Pragmatic Programmers, LLC. USA
- MASON, M. (2006): *Pragmatic Version Control Using Subversion (2<sup>nd</sup> Edition)*. The Pragmatic Programmers, LLC. USA
- COLLINS-SUSSMAN, B. y otros (2006): *Version Control with Subversion*.
- PILONE, D.; PITMAN, N. (2005): *UML 2.0 in a Nutshell*. O’Reilly
- HAMILTON, K.; MILES, R. (2006): *Learning UML 2.0*. O’Reilly
- FOWLER, M.; SCOTT, K (1999). *“UML gota a gota”*. México, Addison Wesley
- PIATTINI, M. G., y GARCÍA, F. O. (2002): *“Calidad en el Desarrollo y Mantenimiento del Software”*. Madrid. Ra-Ma.
- FAIRLEY, R. (1987): *Ingeniería de Software*. México. McGraw-Hill
- Apuntes del módulo profesional “Entornos de Desarrollo” del I.E.S. Julio Verne.

Direcciones URL

- Wikipedia: “Software”; “Software development process”; “Eclipse (software)”; “Software Testing”; “Orientación a Objetos”.
- VOGEL, L.: Tutoriales Vogella: <http://www.vogella.com/tutorials>
- <http://refactoring.com>
- Sitio web de la asignatura “Entornos de Programación” del Grado en Ingeniería Informática por la Universidad Politécnica de Madrid (<http://lml.ls.fi.upm.es/ep/>)
- Scrum Manager Body of Knowledge (<http://www.scrummanager.net/>): *Pruebas de Integración*.
- <http://modeling-languages.com/herramientas-para-uml/>
- [http://en.wikipedia.org/wiki/Comparison\\_of\\_Unified\\_Modeling\\_Language\\_tools](http://en.wikipedia.org/wiki/Comparison_of_Unified_Modeling_Language_tools)
- Página web del CPR de Avilés sobre la unidad didáctica “El ordenador y su funcionamiento” del módulo “Operaciones auxiliares con tecnologías de la información y la comunicación” ([http://www.cpraviles.com/materiales/pcpi/PCPI/index028b.html?page\\_id=52](http://www.cpraviles.com/materiales/pcpi/PCPI/index028b.html?page_id=52))

Además, los alumnos deberán disponer de un cuaderno donde irán recogiendo ordenadamente su información de las clases.

## 10 Atención a la diversidad y adaptaciones curriculares

En el caso en que este módulo sea cursado por un alumno con discapacidad, se realizará la adaptación de las actividades de formación, los criterios y los procedimientos de evaluación necesarios, de modo que se garantice su accesibilidad a las pruebas de evaluación; esta adaptación en ningún caso supondrá la supresión de objetivos, o resultados de aprendizaje que afecten a la competencia general del título. La adaptación curricular se archivará en el expediente del alumno.

## 11 Utilización de las TIC

Al menos habrá un PC en el aula para cada dos alumnos, y otro para el profesor. En cada puesto informático del aula (incluyendo el PC del profesor) se utilizará el sistema operativo Linux o Windows, ya sea instalado en la máquina física o en máquinas virtuales. Todos estos ordenadores estarán conectados a una red local, y tendrán acceso controlado a la red Internet.

Los alumnos utilizarán, entre otras aplicaciones, el IDE Eclipse.

Con bastante frecuencia para la explicación de los contenidos el profesor se ayudará de un cañón vídeo-proyector. Además, Internet será una fuente fundamental para la obtención de información.

Los alumnos, para la realización del trabajo de la primera evaluación, harán uso de la plataforma Colaboratory de Google, a la que accederán utilizando la cuenta de Gmail que el instituto les proporciona a partir de la plataforma Google Suite for Education. Con esa herramienta pueden programar y ejecutar código Python en el navegador, sin necesidad de ninguna instalación ni configuración. Y, tal como dice el tutorial de Google, *“los cuadernos de Colab te permiten combinar código ejecutable y texto enriquecido en un mismo documento, además de imágenes, HTML, LaTeX y mucho más”*, por lo que son una herramienta fantástica para explicar y probar código fuente y realizar tutoriales.

## **12 Actividades de recuperación de módulos profesionales pendientes**

Los alumnos que promocionen al segundo curso con este módulo profesional pendiente realizarán un examen en febrero, correspondiente a la convocatoria ordinaria. Si no aprueban en febrero realizarán un examen en junio, correspondiente a la convocatoria extraordinaria.

Los alumnos podrán recibir clases de recuperación siempre y cuando la organización del centro lo permita.

En ambas convocatorias, el examen abarcará todos los contenidos del módulo profesional, salvo que para la convocatoria extraordinaria el alumno no haya recibido clases de recuperación, en cuyo caso es de aplicación el art. 25.3 de la Orden 2694/2009, que, sobre el examen a realizar, dice que *«tendrá como referentes los criterios de evaluación mínimos incluidos en las programaciones didácticas»* (y el informe que, en su caso, se entrega a cada alumno que tiene módulos pendientes tras la evaluación final ordinaria).

En ambas convocatorias, el examen será calificado con un valor numérico comprendido entre 0 y 10, y se considerará aprobado si este valor es mayor o igual a 5.

Se aplicarán los mismos criterios ortográficos y de redondeo establecidos anteriormente.

## **13 Procedimiento para que el alumnado y sus familias conozcan los objetivos, los contenidos, los criterios de evaluación, los criterios de calificación, los procedimientos y los instrumentos de evaluación.**

La presente programación se publicará en la página web del centro ([www.iesjovellanos.org](http://www.iesjovellanos.org)).



## 14 Actividades complementarias y extraescolares

Se prevé una visita a las IV Jornadas 3D que organiza el I.E.S. Palomeras Vallecas, cuya celebración, a falta de confirmación, será en la semana del 30 de marzo al 2 de abril.

## 15 Referencias

- **Real Decreto 686/2010, de 16 de abril**, por el que se establece el título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma y se fijan sus enseñanzas mínimas.
- **DECRETO 1/2011, de 13 de enero**, del Consejo de Gobierno, por el que se establece para la Comunidad de Madrid el currículo del ciclo formativo de grado superior correspondiente al título de Técnico Superior en Desarrollo de Aplicaciones Web.
- **Orden 2694/2009, de 9 de junio**, por la que se regula el acceso, la matriculación, el proceso de evaluación y la acreditación académica de los alumnos que cursen en la Comunidad de Madrid la modalidad presencial de la formación profesional del sistema educativo establecida en la Ley Orgánica 2/2006, de 3 de mayo, de Educación (BOCM Lunes 22 de Junio de 2009).(Modificada por la Orden 11783/2012, de 11 de diciembre – BOCM de 04/01/2013).
- **Proyecto Educativo de Centro**. IES Gaspar Melchor de Jovellanos, Fuenlabrada.

## 16 Indicadores para la Evaluación de la Práctica Docente

El profesor, al acabar la convocatoria ordinaria, pasará a los alumnos el cuestionario titulado “Encuesta para evaluar al profesor”. A la luz de las respuestas de dicho cuestionario y de la propia reflexión, rellenará su autoevaluación con el formato de la “Autoevaluación de la práctica docente”.

Las preguntas de ambos cuestionarios se contestan con una valoración numérica entre 1 y 4, significando estos valores: Siempre, casi siempre, a veces, nunca

Ambos cuestionarios se incluyen a continuación.

### 16.1 Encuesta para evaluar al profesor

ASPECTO A EVALUAR	EVALUACIÓN
1. Demostró actualización en los temas del módulo, tanto en sus aspectos teóricos como prácticos	
2. Promovió espacios para la participación de los estudiantes en su clase	
3. Promovió el desarrollo de un pensamiento crítico constructivo	
4. Ofreció una orientación clara a las preguntas de los estudiantes	
5. Promovió que los estudiantes asumieran la responsabilidad de su propio aprendizaje	
6. El trabajo asignado por el profesor para desarrollar fuera de clase fue pertinente para el curso	
7. Retroalimentó a los alumnos respecto a su desempeño a lo largo del curso.	
8. Asistió puntualmente a las sesiones y actividades programadas.	
9. Cumplió con lo previsto en el plan del módulo profesional.	
10. Entregó oportunamente (de acuerdo con las fechas límite establecidas en el calendario académico) las notas, los resultados de la evaluación de los trabajos, los informes y exámenes.	
11. Mostró interés en atender las inquietudes de los estudiantes.	
12. Se mostró respetuoso y tolerante hacia los demás y hacia ideas divergentes.	
13. Empleó una metodología que facilitó el aprendizaje y la comprensión de los temas	
14. Favoreció la interacción con los estudiantes a través de las TIC.	
15. Favoreció la consulta permanente de los recursos educativos.	
16. Realizó una temporalización adecuada de las U.T.	
17. Informó a los alumnos sobre los contenidos, secuenciación y criterios de calificación y evaluación, al principio del curso.	
18. Puso en contexto los contenidos del módulo con respecto a la práctica profesional y a los otros módulos.	

## 16.2 Autoevaluación de la práctica docente

ASPECTO A EVALUAR	A DESTACAR	A MEJORAR	PROPUESTAS DE MEJORA PERSONAL
Temporizo correctamente las unidades didácticas			
Desarrollo los objetivos didácticos			
Manejo los contenidos de la unidad			
Realizo tareas			
Uso estrategias metodológicas			
Uso buenos recursos			
Soy claro en los criterios de evaluación			
Uso diversas herramientas de evaluación			
Planifico actividades intentando coordinarlas con los contenidos o actividades de otros módulos.			
Utilizo criterios de evaluación de las actividades que los alumnos perciben como claros y uniformes			
Los alumnos perciben que he explicado claramente al principio del curso la planificación y los criterios de evaluación.			
Pongo en contexto cada unidad temática con el desarrollo de la actividad profesional.			
Pongo en contexto las unidades temáticas con los contenidos de otros módulos u otras unidades temáticas del mismo módulo.			

## 17 Adaptaciones con motivo del Escenario en Fase II (COVID-19).

Teniendo en cuenta las **INSTRUCCIONES PARA EL DESARROLLO DEL CURSO 2020-2021 EN LA COMUNIDAD DE MADRID COMO CONSECUENCIA DEL ESCENARIO EN FASE II PROVOCADO POR CORONAVIRUS (COVID-19)**, resultamos en realizar las siguientes adaptaciones:

- Las sesiones se podrán retransmitir en línea o grabar para su posterior visionado haciendo uso de herramientas de videoconferencia y vídeos para la Mediateca de Educamadrid como recursos didácticos complementarios al Aula Virtual que ya se utiliza, así se *avanzará en aspectos del currículo no abordados en el aula de forma presencial, siempre con las adaptaciones para su desarrollo a distancia y en línea, facilitando la participación de todos los alumnos.*

- Se intensifica la comunicación con el alumnado, fundamentalmente *de las actividades que van a realizar y de los contenidos de la programación que van a ser fundamentales para su progresión y aprovechamiento futuro, así como de las adaptaciones de las programaciones didácticas que los órganos de coordinación docente competentes hayan aprobado, en especial de los criterios de calificación.*