

ENTOROS DE DESARROLLO

(CÓDIGO: 0487)

Álvaro González Sotillo

17 de septiembre de 2025

Índice

1. Cómo serán las clases	1
2. Materiales	2
3. Entrega de Trabajos	2
4. Normas	2
5. Cómo será la evaluación	3
6. Actividades	3
7. Contenidos	3
8. Criterios de evaluación	6
9. Distribución de RA en unidades de trabajo	8
10. Referencias	8

1. Cómo serán las clases

- Teoría
 - Basada en apuntes
 - Y un aula virtual
- Ejercicios
 - Se realizan en clase o en casa
 - Se ponen en común al día siguiente
- Práctica
 - Máquinas virtuales
 - Posiblemente, una nube
- Trabajos
 - Expondremos algunos en clase

2. Materiales

- Memoria USB
- Correo electrónico
- Acceso a Internet fuera del aula

3. Entrega de Trabajos

- Via Moodle
 - Nuestro curso es <https://aulavirtual3.educa.madrid.org/ies.alonsodeavellan.alcala>
 - El curso es accesible incluso sin usuario, inicialmente
- Se utilizará Microsoft Office (, **DOCX**, **PPTX**)
 - Opcionalmente, **PDF** o LibreOffice (**ODT**, **ODP**)
 - También es desable entregas *online* (Canvas, Figlet...)
- Se tendrá en cuenta
 - La corrección técnica de los trabajos
 - La fecha de entrega
 - Expresión, sintaxis, ortografía
 - La apariencia profesional

4. Normas

- Retrasos y faltas
- Uso de los ordenadores
 - No pueden utilizarse para tareas distintas de las encargadas por el profesor
 - Se respetará a otros alumnos
- Móviles
 - No.

4.1. Averías de los ordenadores

- Los problemas se comunican al profesor en cuanto se detectan
- Se deben hacer copias de seguridad para no perder los datos de los discos
 - Pen Drive
 - Disco Externo
 - Correos enviados a uno mismo
 - Copias en los ordenadores de otros compañeros
- Norma fundamental:

Si se pierde porque no hay copia, es que no era importante

5. Cómo será la evaluación

- Las notas de las evaluaciones (1^a, 2^a, 3^a) no son realmente importantes
- Solo interesa la nota de la evaluación final
- Basado en *Resultados de aprendizaje* (RA)
 - Cada RA supone un porcentaje de la nota final
 - Cada prueba (examen, trabajo) indicará que RA evalúa, en qué porcentaje
 - Se necesita aprobar cada RA para aprobar el módulo

6. Actividades

- Trabajos
- Exámenes
- Examen final evaluación ordinaria
 - Con los RA no superados
- Examen evaluación extraordinaria
 - Incluirá todos los RA. La nota del examen será la nota del módulo.
- Entrega de trabajos
 - Individuales, o por parejas si se comparte ordenador
 - Un trabajo entregado fuera de plazo tiene una nota máxima de 6

7. Contenidos

Según el Decreto 3/2011, de 13 de enero
Desarrollo de software:

- Concepto de programa informático. Instrucciones y datos.
- Ejecución de programas en ordenadores:
 - Datos y programas.
 - Hardware vs. software.
 - Estructura funcional de un ordenador: procesador, memoria.
 - Tipos de software. BIOS. Sistema. Aplicaciones.
 - Código fuente, código objeto y código ejecutable; máquinas virtuales.
- Lenguajes de programación:
 - Tipos de lenguajes de programación.
 - Características de los lenguajes más difundidos.
- Introducción a la ingeniería del software:
 - Proceso software y ciclo de vida del software.
 - Fases del desarrollo de una aplicación: análisis, diseño, codificación, pruebas, documentación, explotación y mantenimiento, entre otras.
 - Modelos de proceso de desarrollo software (cascada, iterativo, evolutivo).
 - Metodologías de desarrollo software. Características. Técnicas. Objetivos. Tipos de metodologías:
 - Herramientas CASE (Computer Aided Software Engineering).

-
- Proceso de obtención de código ejecutable a partir del código fuente; herramientas implicadas (editores, compiladores, enlazadores, etcétera).
 - Errores en el desarrollo de programas.
 - Importancia de la reutilización de código.

Instalación y uso de entornos de desarrollo:

- Funciones de un entorno de desarrollo.
- Tipos de entornos de desarrollo. Entornos de desarrollo libres y propietarios. Características.
- Instalación de un entorno de desarrollo.
- Uso básico de un entorno de desarrollo:
 - Uso de herramientas y asistentes.
 - Creación de proyectos.
 - Incorporación de elementos a proyectos.
 - Edición de programas. Sintaxis y formateo de código.
 - Compilación de programas. Detección de errores.
 - Generación de ejecutables.
 - Ejecución de programas.
 - Paneles y vistas.
 - Importación y exportación de ficheros.
 - Personalización.
 - Acceso a documentación.
 - Instalación y desinstalación de aplicaciones, módulos y plugins adicionales.
 - Configuración de actualizaciones.
 - Automatización de tareas.

Diseño y realización de pruebas:

- Pruebas en el proceso de desarrollo de software:
 - Planificación de pruebas a lo largo del ciclo de desarrollo.
 - Tipos de pruebas: funcionales, estructurales, regresión, caja negra.
 - Procedimientos y casos de prueba.
- Pruebas de código:
 - Cubrimiento, valores límite, clases de equivalencia.
 - Pruebas unitarias de clases y funciones.
 - Uso de herramientas integradas en los entornos de desarrollo para realizar pruebas unitarias.
 - Automatización de pruebas unitarias.
 - Pruebas de integración.
 - Diseño y documentación casos de prueba.
- Depuración de programas:
 - Herramientas de depuración integradas en los entornos de desarrollo,
 - Puntos de ruptura y seguimiento en tiempo de ejecución.
 - Examinadores de variables.

Optimización y documentación:

-
- Refactorización:
 - Concepto. Limitaciones.
 - Patrones de refactorización más usuales.
 - Refactorización y pruebas.
 - Herramientas de ayuda a la refactorización.
 - Control de versiones:
 - Desarrollos colectivos
 - Herramientas de control de versiones. Utilidad. Características. Estructura (cliente/servidor). Repositorio.
 - Clientes de control de versiones. Descarga de ficheros inicial. Modificación de ficheros. Actualización de ficheros en local. Actualización de ficheros en el repositorio. Diferencias entre versiones. Restauración de versiones anteriores. Resolución de conflictos. Historial de versiones.
 - Documentación:
 - Uso de comentarios.
 - Herramientas integradas en el entorno de desarrollo para generar documentación automáticas de clases.
 - Alternativas.

Introducción al lenguaje unificado de modelado (UML, Unified Modeling Language):

- Características.
- Versiones.
- Diagramas UML.
- Utilización en metodologías de desarrollo orientado a objetos.
- Herramientas CASE con soporte UML.

Elaboración de diagramas de clases:

- Notación de los diagramas de clases:
 - Clases. Atributos, métodos y visibilidad.
 - Objetos. Instanciación.
 - Relaciones. Asociación, herencia, composición, agregación, dependencia, navegabilidad.
 - Clases abstractas. Interfaces.
 - Paquetes.
 - Grado de detalle.
- Utilización de herramientas CASE para elaborar diagramas de clases.
- Módulos integrados en entornos de desarrollo para elaborar diagramas de clases.
- Creación de código a partir de diagramas de clases.
- Generación de diagramas de clases a partir de código (ingeniería inversa).

Elaboración de diagramas de comportamiento:

- Tipos. Campo de aplicación.
- Diagramas de casos de uso. Actores, casos de uso, escenario, asociaciones (relación de comunicación entre actores y casos de uso), relaciones entre casos de uso.
- Diagramas de secuencia. Línea de vida de un objeto/actor, activación, envío de mensajes.
- Diagramas de colaboración. Objetos/actores, mensajes.

-
- Otros diagramas:
 - Diagramas de actividades.
 - Diagramas de estado.
 - Utilización de herramientas CASE para elaborar diagramas de comportamiento
 - Módulos integrados en entornos de desarrollo para elaborar diagramas de comportamiento.

8. Criterios de evaluación

Según Real Decreto 450/2010, de 16 de abril Modificados en Real Decreto 405/2023, de 29 de mayo

8.1. RA1 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 identificado las fases de desarrollo de una aplicación informática.
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 han clasificado los lenguajes de programación, identificando sus características.
6. Se ha evaluado la funcionalidad ofrecida por las herramientas utilizadas en el desarrollo de software.
7. Se han identificado las características y escenarios de uso de las metodologías ágiles de desarrollo de software.

8.2. RA2 Evalúa entornos integrados de desarrollo analizando sus características para editar código fuente y generar ejecutables.

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.

8.3. RA3 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 han efectuado pruebas unitarias de clases y funciones.
7. Se han implementado pruebas automáticas.
8. Se han documentado las incidencias detectadas.
9. Se han utilizado dobles de prueba para aislar los componentes durante las pruebas.

8.4. RA4 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 han utilizado herramientas del entorno de desarrollo para documentar las clases.
8. Se han utilizado repositorios remotos para el desarrollo de código colaborativo.
9. Se han utilizado herramientas para la integración continua del código.

8.5. RA5 Genera diagramas de clases valorando su importancia en el desarrollo de aplicaciones y empleando herramientas específicas.

Criterios de evaluación:

1. Se han identificado los conceptos básicos de la programación orientada a objetos.
2. Se han utilizado herramientas para la elaboración de diagramas de clases.
3. Se ha interpretado el significado de diagramas de clases.
4. Se han trazado diagramas de clases a partir de las especificaciones de las mismas.
5. Se ha generado código a partir de un diagrama de clases.
6. Se ha generado un diagrama de clases mediante ingeniería inversa.

8.6. RA6 Genera diagramas de comportamiento valorando su importancia en el desarrollo de aplicaciones y empleando herramientas específicas.

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 ha interpretado el significado de diagramas de actividades.
6. Se han elaborado diagramas de actividades sencillos.
7. Se han interpretado diagramas de estados.
8. Se han planteado diagramas de estados sencillos.

9. Distribución de RA en unidades de trabajo

[Enlace a hoja excel](#)

10. Referencias

- Formatos:
 - [Transparencias](#)
 - [PDF](#)
 - [Página web](#)
 - [EPUB](#)
- Creado con:
 - [Emacs](#)
 - [org-re-reveal](#)
 - [Latex](#)
- Alojado en [Github](#)