



## Guía docente

# 804237 - DESVJ - Desarrollo de Videojuegos

Última modificación: 20/07/2025

**Unidad responsable:** Centro de la Imagen y la Tecnología Multimedia

**Unidad que imparte:** 804 - CITM - Centro de la Imagen y la Tecnología Multimedia.

**Titulación:** GRADO EN DISEÑO Y DESARROLLO DE VIDEOJUEGOS (Plan 2014). (Asignatura obligatoria).

**Curso:** 2025

**Créditos ECTS:** 6.0

**Idiomas:** Castellano, Inglés

## PROFESORADO

**Profesorado responsable:** Pedro Omedas

**Otros:**

## METODOLOGÍAS DOCENTES

A lo largo del curso, los estudiantes colaborarán en parejas para trabajar en el desarrollo de un juego de plataformas 2D utilizando C++ y la biblioteca SDL. En cada clase, el profesor demostrará varios enfoques y compartirá las mejores prácticas para implementar funcionalidades en sus proyectos. Para apoyar su aprendizaje, el profesor proporcionará ejemplos de código fuente y plantillas para que los estudiantes estudien, completen e integren en su propio código fuente para futuras referencias y aplicaciones prácticas.

Al final de cada sesión, el profesor ofrecerá ideas para mejorar los sistemas de juego, presentando desafíos que animen a los estudiantes a explorar y mejorar sus proyectos de forma independiente. Esta guía tiene como objetivo ayudar y guiar a los estudiantes durante su tiempo de autoaprendizaje, permitiéndoles desarrollar aún más sus habilidades y fomentar un sentido de autonomía.

Al seguir este enfoque de enseñanza, los estudiantes no solo adquieren experiencia práctica en el desarrollo de juegos, sino que también reciben orientación directa y exposición a técnicas y prácticas estándar de la industria. La combinación de trabajo colaborativo, ejemplos prácticos y oportunidades de aprendizaje independiente garantiza una experiencia de aprendizaje completa y atractiva a lo largo del curso.

## OBJETIVOS DE APRENDIZAJE DE LA ASIGNATURA

- Identificar las librerías gráficas para la creación de imágenes sintéticas, gráficos en 2D y 3D e interfaces de usuario.
- Utilizar lenguajes de programación, patrones algorítmicos, estructuras de datos, herramientas visuales de programación, motores de juego, librerías y servidores para el desarrollo de videojuegos de distintos géneros y para diferentes plataformas y dispositivos.
- Aplicar las técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real, en el desarrollo de videojuegos para juego local o en línea.
- Aplicar técnicas de programación gráfica, física, inteligencia artificial, realidad aumentada y virtual, interfaces de usuario e interacción persona-ordenador a proyectos de videojuegos de forma eficiente.
- Desarrollar videojuegos, a través del uso de lenguajes de programación, patrones algorítmicos, estructuras de datos, herramientas visuales de programación, motores de juego, librerías y servidores.
- Identificar la arquitectura de un motor de juego 2D, incluidos sus componentes principales y sus funciones en el desarrollo del juego.
- Desarrollar las capacidades del motor del juego incorporando nuevas funciones como entidades, GUI, mapas y sistemas de física, para que las experiencias de juego sean atractivas.



## HORAS TOTALES DE DEDICACIÓN DEL ESTUDIANTADO

Tipo	Horas	Porcentaje
Horas grupo grande	18,0	12.00
Horas grupo mediano	30,0	20.00
Horas actividades dirigidas	12,0	8.00
Horas aprendizaje autónomo	90,0	60.00

**Dedicación total:** 150 h

## CONTENIDOS

### Control de los FPS y los tiempos de la lógica

**Descripción:**

Técnicas para el control del Frame rate.  
Sistemas de manipulación de los tiempos de logica (pausa, bullet time, etc.)

**Dedicación:** 13h

Grupo grande/Teoría: 5h  
Aprendizaje autónomo: 8h

### Control de entidades de juego

**Descripción:**

Teoria de los sistemas de entidades en los videojuegos.  
Programación de un sistema de entidades autónomo.

**Dedicación:** 19h

Grupo grande/Teoría: 9h  
Aprendizaje autónomo: 10h

### Carga y pintado de mapas de Tiled

**Descripción:**

Uso de programa Tiles para creación de mapas.  
Análisis del formato TMX.  
Código de carga de ficheros TMX.  
Metodología para el pintado de mapas ortogonales.  
Metodología para el pintado de mapas en isométrica.

**Dedicación:** 22h

Grupo grande/Teoría: 8h  
Aprendizaje autónomo: 14h



## Serialización

**Descripción:**

Teoría detrás del arte de cargar recursos en videojuegos.

Formatos (XML, JSON, YAML).

Serialización de objetos usando bibliotecas.

Cargar y guardar datos.

**Dedicación:** 16h

Grupo grande/Teoría: 6h

Aprendizaje autónomo: 10h

## Metainformación y mapas de máscaras

**Descripción:**

Uso del Tiled para meta información.

Carga de mapas de navegación como meta información.

Uso alternativo de máscaras para mapas de navegación.

**Dedicación:** 13h

Grupo grande/Teoría: 5h

Aprendizaje autónomo: 8h

## Algoritmos de búsqueda de caminos

**Descripción:**

BFS (Breadth First Search).

Dijkstra.

A\*

**Dedicación:** 30h

Grupo grande/Teoría: 12h

Aprendizaje autónomo: 18h

## Física

**Descripción:**

Integrando el motor de física Box2D.

Aprende a usar las funciones de la biblioteca de física.

**Dedicación:** 8h

Grupo grande/Teoría: 4h

Aprendizaje autónomo: 4h



## Sistemas de Interfaces Gráficas de Usuario

### Descripción:

Estructura de los GUI para videojuegos.

Ventanas y scroll.

Botones con imágenes.

Entradas de texto.

Barras de progreso.

### Dedicación: 25h

Grupo grande/Teoría: 10h

Aprendizaje autónomo: 15h

## Optimización

### Descripción:

Profiling del código, detección de fugas de memoria.

### Dedicación: 4h

Grupo grande/Teoría: 2h

Aprendizaje autónomo: 2h

## ACTIVIDADES

### Asignación 1

#### Descripción:

Crea un juego de plataformas simple con los siguientes elementos:

- Carga del archivo de configuración
- Serialización de datos
- Carga y dibujo del mapa TMX en mosaico (ortográfico)
- Detección de colisiones en mapas (plataformas)
- Navegación del mapa: movimiento del jugador y saltos
- El juego debe tener un límite estable de 60 cuadros por segundo sin vsync.

#### Dedicación: 10h

Aprendizaje autónomo: 10h

### Asignación 2

#### Descripción:

Al expandir el juego de plataformas de la asignación anterior, debemos agregar:

- Tipo de enemigo que camina que puede encontrar al jugador. No es necesario que el enemigo pueda saltar (aunque se recomienda) pero debe detectar que puede llegar al jugador caminando normalmente y cayendo a otras plataformas.
- Tipo de enemigo volador que puede encontrar al jugador evitando áreas no transitables.
- Cargar / Guardar debe considerar cada estado enemigo. Los enemigos normalmente tienen un rango de percepción y no reaccionar ante el jugador hasta que esté cerca.

#### Dedicación: 10h

Aprendizaje autónomo: 10h



### Asignación 3

**Descripción:**

Expandir el juego de plataformas de la asignación anterior, agregando:

- Sistema de entidad
- GUI: menú principal de la pantalla de título
- GUI: pantalla de juego HUD
- Game profiling
- GUI: menú de pausa de la pantalla de juego

**Dedicación:** 10h

Aprendizaje autónomo: 10h

## SISTEMA DE CALIFICACIÓN

Tres prácticas con un peso del 20%, 20% y 20% cada una de la nota final.

Un examen final con un peso total del 30% de la nota final. Constará de una prueba práctica y teórica de dos horas de duración.

Una reevaluación con un peso total del 30% de la nota final (examen final). Constará de una prueba práctica y teórica de dos horas de duración. En caso de aprobar la asignatura, la nota final máxima será de 5.

Una calificación final del 10% será sobre participación en clase y actitud.

Las acciones irregulares que puedan llevar a una variación significativa de la calificación de uno o más estudiantes constituyen una realización fraudulenta de un acto de evaluación. Esta acción comporta la calificación descriptiva de suspenso y numérica de 0 del acto de evaluación ordinario global de la asignatura, sin derecho a reevaluación.

Si los docentes tienen indicios de la utilización de herramientas de IA no permitidas en las pruebas de evaluación, podrán convocar a los estudiantes implicados a una prueba oral o a una reunión para verificar la autoría.

## BIBLIOGRAFÍA

**Complementaria:**

- Thorn, A. Game engine design and implementation. Sudbury, Mass: Jones & Bartlett Learning, 2011. ISBN 9780763784515.
- McShaffry, M.; Graham, D. Game coding complete. 4th ed. Boston, Mass: Course Technology, 2012. ISBN 9781133776574.
- Gregory, J. Game engine architecture. 2nd ed. Boca Raton: CRC Press, 2014. ISBN 9781466560017.