

## Guía docente

### 804254 - XJO - Redes y Juegos en Línea

Última modificación: 12/09/2025

**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:** Catalán, Inglés

#### PROFESORADO

---

**Profesorado responsable:** Sergi Abadal

**Otros:** Sergi Abadal  
Manuel Rello  
Laura Gotarra

#### CAPACIDADES PREVIAS

---

Ser capaz de programar y desarrollar aplicaciones informáticas  
Conocer las principales herramientas y servicios de Internet a nivel de usuario

#### REQUISITOS

---

Programación básica.

## METODOLOGÍAS DOCENTES

La metodología docente se divide en cuatro partes:

- Sesiones presenciales de exposición de los contenidos.
- Sesiones presenciales de trabajo práctico (aprendizaje en el uso de las herramientas, presentaciones y debate de casos).
- Desarrollo práctico de juegos en red aplicando los conceptos aprendidos durante el curso.
- Trabajo autónomo de estudio y realización de ejercicios y actividades.

En las sesiones de exposición de los contenidos, el profesorado introducirá las bases teóricas de la materia, conceptos, métodos y resultados ilustrándolo con ejemplos convenientes para facilitar su comprensión. Dentro de este bloque también se incluye una actividad donde el alumnado escogerá un tema relacionado con la asignatura y lo expondrá en clase.

En las sesiones de trabajo práctico en el aula, el profesorado guiará al estudiante en la aplicación de los conceptos teóricos para la resolución de problemas, fundamentándose en todo momento el razonamiento crítico. Se propondrá un proyecto final, con múltiples entregables evaluables que el estudiante resolverá tanto en el aula como fuera del aula, tratando de favorecer el contacto y utilización de las herramientas básicas necesarias para la resolución de problemas.

El estudiante, de forma autónoma, tiene que trabajar el material proporcionado por el profesorado y el resultado de las sesiones de trabajo-problemas para asimilar y fijar los conceptos. El profesorado proporcionará un plan de estudio y de seguimiento de actividades (CAMPUS).

La metodología de la asignatura es una combinación entre la clase presencial y la realización de diferentes actividades complementarias (resolución de casos, problemas, etc.), ya sean individuales o en pequeños grupos, tanto dentro como fuera del aula. En las horas presenciales del estudiante recibe la información básica para trabajar los contenidos del plan docente que le garanticen la obtención de los objetivos establecidos.

En el Campus Virtual se publica toda la información relativa a la asignatura (normas de evaluación, fechas de entrega de las actividades, etc.). La asistencia a las sesiones es importante para el proceso de aprendizaje, el cual se basa tanto en el trabajo individual o en equipo, como en la participación en el aula.

## OBJETIVOS DE APRENDIZAJE DE LA ASIGNATURA

- Identificar servidores de juegos y protocolos para el desarrollo de videojuegos en línea.
- 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.
- 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.
- Utilizar motores de juego en red en el desarrollo de juegos online.

## 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

### 1. Introducción

**Descripción:**

- 1.1 Aspectos generales de la asignatura
- 1.2 Una breve historia de redes y juegos on-line
- 1.3 Arquitecturas y métricas
- 1.4 Teoría de grafos para modelar redes
- 1.5 Algoritmos aplicados a grafos

**Objetivos específicos:**

Adquirir conocimientos básicos sobre la historia de Internet y los juegos on-line, los tipos básicos de arquitectura de red y métricas utilizadas para medir el rendimiento. Aprender las bases de la teoría de grafos y algoritmos aplicados a grafos, los cuales forman una base abstracta común a redes y a muchos otros ámbitos.

**Actividades vinculadas:**

Programación con threads

**Dedicación:** 12h

Grupo grande/Teoría: 4h

Aprendizaje autónomo: 8h

### 2. La Pila de Protocolos OSI

**Descripción:**

- 2.1 Introducción. Las capas de aplicación y transporte
- 2.2 Las capas inferiores: red, enlace, y física
- 2.3 Otros protocolos relevantes (NAT, HTTP, QUIC, etc.)
- 2.4 Análisis de protocolos con Wireshark

**Objetivos específicos:**

Entender la estructura organizada en capas que se utiliza en redes, así como el proceso de encapsulamiento y su utilidad. Aprender la funcionalidad de cada capa de abstracción, así como conocer los protocolos más extendidos en cada capa con especial énfasis en los más relevantes para la programación de juegos on-line (IP, TCP/UDP, NAT). Interiorizar la conexión entre el extremo del juego que toca la red (serializador) y la red (socket). Aprender las bases de uso de packet sniffers, así como usarlos para observar la estructura de paquetes en redes reales.

**Actividades vinculadas:**

- 2. Sockets (TCP/UDP)
  - 3. Serialización de datos
- (incluye también temas prácticos de NAT)

**Dedicación:** 24h

Grupo grande/Teoría: 8h

Aprendizaje autónomo: 16h

### 3. Impacto de la red en la programación de juegos on-line

**Descripción:**

- 3.1 Introducción y aspectos generales. Replicación de objetos
- 3.2 Latencia, jitter, fiabilidad
- 3.3 Técnicas para esconder la latencia
- 3.4 Escalabilidad y entornos masivos

**Objetivos específicos:**

Entender como la red impacta en la programación de juegos en línea, mediante el estudio de problemas derivados de la pérdida de información o retrasos.

Aprender técnicas para esconder los problemas de la red, minimizando su impacto para que no lastre el rendimiento del juego.

Aprender las ideas más utilizadas en la mayoría de juegos, ya sea con pocos o muchos jugadores simultáneos.

**Actividades vinculadas:**

- 4. Replicación del estado del mundo
- 5. Técnicas de mitigación de latencia y jitter
- 6. Juego online completo

**Dedicación:** 30h

Grupo grande/Teoría: 10h

Aprendizaje autónomo: 20h

### 4. Presentaciones de teoría

**Descripción:**

Los estudiantes deberán profundizar en algún tema de la asignatura, escogido por ellos y acordado con el profesor, para la realización de una breve presentación en clase.

**Objetivos específicos:**

- Profundizar en un tema específico de redes o programación de juegos online.
- Recopilar información y estudiar el tema de manera autónoma.
- Presentar el tema de manera clara.

**Actividades vinculadas:**

N/A

**Dedicación:** 18h

Grupo grande/Teoría: 6h

Aprendizaje autónomo: 12h

## ACTIVIDADES

### 1. Threads

**Descripción:**

Aprender a utilizar mecanismos de ejecución de código en paralelo (threads) i las herramientas necesarias para mantener la sincronización entre estos, y la seguridad a la hora de acceder de forma concurrent a regiones de memoria compartidas.

**Objetivos específicos:**

- Threads (mecanismo de ejecución de código en paralelo)
- Mutex objects (mecanismo para proteger accesos concurrentes a datos)
- Condition variables (mecanismo de sincronización entre threads)

**Material:**

Instrucciones para la práctica y esqueleto de código.

**Entregable:**

Entregable #1 (sin peso en la nota final)

**Dedicación:** 6h

Grupo mediano/Prácticas: 2h

Aprendizaje autónomo: 4h

### 2. Sockets (TCP/UDP)

**Descripción:**

Aprender los mecanismos básicos de utilización de sockets TCP i UDP con tal de transmitir datos entre dos o más máquinas a través de una red. Esta actividad define las bases para comunicar información entre cliente y servidor en el contexto de un proyecto final.

**Objetivos específicos:**

- Sockets TCP (orientados a conexión)
- Sockets UDP (no orientados a conexión)
- Uso no bloqueante de funciones de acceso a sockets (non-blocking mode, select).

**Material:**

Instrucciones para la práctica i esqueleto de código.

**Entregable:**

Entregable #2 (5% nota final)

**Dedicación:** 12h

Grupo mediano/Prácticas: 4h

Aprendizaje autónomo: 8h

### 3. Serialización de datos

**Descripción:**

Aquí veremos la importancia de la serialización de datos para transmitir objetos potencialmente complejos en memoria, ya que estos objetos pueden no estar representados de forma contigua y pueden contener punteros a otros objetos, que imposibiliten una copia directa. Esta actividad extenderá el proyecto final añadiendo las herramientas para serializar, al menos, uno de los objetos del videojuego.

**Objetivos específicos:**

- Acceso secuencial y ordenado a los datos.
- Objetos del tipo MemoryStream.

**Material:**

Instrucciones para la práctica i esqueleto de código.

**Entregable:**

Entregable #3 (5% nota final)

Entregable #4 - Demo I (10% nota final)

**Dedicación:** 18h

Grupo mediano/Prácticas: 6h

Aprendizaje autónomo: 12h

### 4. Replicación del estado del mundo

**Descripción:**

Aprender y aplicar buenas prácticas para replicar correctamente el estado del mundo del videojuego. Esta práctica expandirá el proyecto incluyendo los métodos para serializar los datos de cualquier objeto del mundo y asegurar su correcta recepción en el cliente.

**Objetivos específicos:**

- Definición de un mánager de replicación.
- Tipos de replicación de objetos (activa/pasiva)
- Programación orientada a objetos

**Material:**

Instrucciones para la práctica, definiciones.

**Entregable:**

Entregable #5 (10% nota final)

**Dedicación:** 12h

Grupo mediano/Prácticas: 4h

Aprendizaje autónomo: 8h

## 5. Técnicas de mitigación de latencia, jitter

### Descripción:

Aquí veremos la importancia que cobran los problemas como el jitter o la latencia. El estudiante desarrollará técnicas para mitigar estos y otros problemas derivados de la comunicación en juegos online. Esta práctica será parte importante del proyecto final.

### Objetivos específicos:

- Predicción a nivel de cliente
- Reconciliación con el servidor
- Entity Interpolation
- Lag Compensation

### Material:

Instrucciones para la práctica y código auxiliar.

### Entregable:

Entregable #6 (10% nota final)

### Dedicación: 12h

Grupo mediano/Prácticas: 4h

Aprendizaje autónomo: 8h

## 6. Juego online completo

### Descripción:

Pondremos en práctica los conceptos aprendidos hasta ahora (threads, sockets, y serialización de datos) para desarrollar un videojuego multijugador en red sobre la base del juego desarrollado a lo largo del curso. Este entregable se evaluará en función de la integración de las anteriores actividades en el contexto de un juego completo.

### Objetivos específicos:

- Arquitectura cliente-servidor.
- Servidor con autoridad.
- Replicación del estado del juego.
- Mensajes entre hosts.
- Mecanismos de sincronización de datos (commands, syncvars y hooks, client RCPs, etc).

### Material:

Instrucciones para la práctica.

### Entregable:

Entregable #7 - Demo II (10% nota final)

### Dedicación: 18h

Grupo mediano/Prácticas: 6h

Aprendizaje autónomo: 12h

## SISTEMA DE CALIFICACIÓN

---

Prácticas: 50%

(de los cuales: Sockets 5%; serialización 15%, incluyendo una demo intermedia; entregable world-state 10%; entregable latencia y jitter 10%; demo de juego completo 10%)

Presentación de teoría (participación): 10%

Examen parcial teoría: 15%

Examen final: 25%

Todas las actividades son evaluadas por los profesores, excepto las demos de medio curso y final de curso, donde la evaluación será mixta entre profesores y estudiantes.

En caso de suspender la asignatura mediante la evaluación continua se tendrá la opción de realizar un examen de reevaluación de la parte teórica, correspondiente al 40% de la nota de la asignatura (examen parcial y final). En caso de aprobar la asignatura durante la reevaluación, la nota máxima final será un 5.

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.

## NORMAS PARA LA REALIZACIÓN DE LAS PRUEBAS.

---

La evaluación es individual, aunque la presentación de teoría y las prácticas se hacen en grupo.

Para las prácticas se puede consultar todo el material de estudio disponible y su entrega se ha de realizar en la fecha indicada sin posibilidad de extensión.

Los exámenes se realizarán sin acceso a ningún material de aprendizaje.

## BIBLIOGRAFÍA

---

### Básica:

- Tanenbaum, A.S.; Wetherall, D.J. Computer networks. 5th ed. Boston: Pearson, 2011. ISBN 9780132126953.
- Glazer, Josh; Madhav, Sanjay . Multiplayer game programming: Architecting networked games. Addison-Wesley Professional, 2015. ISBN 9780134034331.

## RECURSOS

---

### Otros recursos:

Lecturas adicionales que se publicarán en el campus virtual durante el curso, incluyendo:

- The history of Internet
- I shot you first: The Networking of HALO
- 1500 Archers on a 28.8: Network Programming in Age of Empires and Beyond
- The TRIBES Engine Networking Model