

Máster Título Propio

Programación de Videojuegos



Máster Título Propio Programación de Videojuegos

- » Modalidad: online
- » Duración: 12 meses
- » Titulación: TECH Universidad
- » Acreditación: 60 ECTS
- » Horario: a tu ritmo
- » Exámenes: online

Acceso web: www.techtute.com/informatica/master/master-programacion-videojuegos



Índice

01

Presentación del programa

pág. 4

02

¿Por qué estudiar en TECH?

pág. 8

03

Plan de estudios

pág. 12

04

Objetivos docentes

pág. 26

05

Salidas profesionales

pág. 32

06

Licencias de software incluidas

pág. 36

07

Metodología de estudio

pág. 40

08

Titulación

pág. 50

01

Presentación del programa

La Programación de Videojuegos ha emergido como una de las áreas más dinámicas y demandadas dentro de la industria tecnológica. Según un informe del Instituto Nacional de Estadística y Geografía, la industria global de videojuegos genera ingresos cercanos a los 180 mil millones de dólares, reflejando el impacto económico y el crecimiento exponencial de este sector. Este contexto, se refleja la necesidad de profesionales capacitados con conocimientos técnicos y creativos para desarrollar productos innovadores. De hecho, es en este escenario que surge esta oportunidad académica de TECH, que tiene como objetivo preparar a los profesionales para afrontar los retos de la programación moderna. A través de material didáctico 100% online, el alumnado podrá perfeccionar habilidades esenciales para esta disciplina.



“

Este programa universitario 100% online representa una oportunidad única para potenciar habilidades en la Programación de Videojuegos”

El desarrollo de videojuegos se ha convertido en una disciplina esencial dentro de la tecnología moderna, con un impacto que trasciende la industria del entretenimiento. Actualmente, el diseño y la Programación de Videojuegos tienen aplicaciones más allá del ocio, abarcando áreas como la simulación, la educación y la inteligencia artificial. En un contexto digital creciente, contar con habilidades en este campo es crucial para los profesionales que buscan contribuir a la innovación tecnológica y participar en el desarrollo de soluciones interactivas en diversos sectores.

Por lo tanto, TECH mediante un novedoso plan de estudios, profundizará en los pilares fundamentales de la Programación de Videojuegos, abordando temas clave como los fundamentos de programación, la estructura de datos y los algoritmos. De hecho, estos conocimientos son esenciales para el desarrollo de juegos que no solo sean funcionales, sino también eficientes y optimizados. Asimismo, la programación proporcionará las bases para crear código limpio y escalable, mientras que una comprensión sólida de la estructura de datos y los algoritmos permitirá gestionar información de manera efectiva, mejorando la interacción y el rendimiento de los juegos.

A su vez, este programa universitario proporcionará a los profesionales las herramientas necesarias para enfrentar desafíos técnicos complejos en el campo del desarrollo de videojuegos. A través de un enfoque integral, el alumnado adquirirá capacidades que les permitirán desarrollar videojuegos innovadores y funcionales, con aplicaciones en diversos sectores. Así, los egresados estarán equipados para destacarse en un mercado laboral en constante evolución.

Finalmente, la metodología TECH Universidad ofrecerá un enfoque flexible y adaptado a las demandas actuales, permitiendo a los profesionales acceder al contenido en línea las 24 horas del día, los 7 días de la semana, desde cualquier dispositivo con conexión a internet. Posteriormente, el método *Relearning* proporcionará una experiencia académica continua, permitiendo repasar y reforzar los conocimientos en cualquier momento, lo que garantizará una capacitación efectiva y que se adapta a las necesidades del alumnado.

Este **Máster Título Propio en Programación de Videojuegos** contiene el programa universitario más completo y actualizado del mercado. Sus características más destacadas son:

- ♦ El desarrollo de casos prácticos presentados por expertos en Programación de Videojuegos
- ♦ Los contenidos gráficos, esquemáticos y eminentemente prácticos con los que están concebidos recogen una información científica y práctica sobre aquellas disciplinas indispensables para el ejercicio profesional
- ♦ Los ejercicios prácticos donde realizar el proceso de autoevaluación para mejorar el aprendizaje
- ♦ Su especial hincapié en metodologías innovadoras en videojuegos
- ♦ Las lecciones teóricas, preguntas al experto, foros de discusión de temas controvertidos y trabajos de reflexión individual
- ♦ La disponibilidad de acceso a los contenidos desde cualquier dispositivo fijo o portátil con conexión a internet



Gracias a esta novedosa preparación académica, ampliarás tu capacidad para integrar nuevas tecnologías en el desarrollo de juegos innovadores”

“

Garantizarás la correcta implementación de los fundamentos de Programación, asegurando el desarrollo de códigos eficientes”

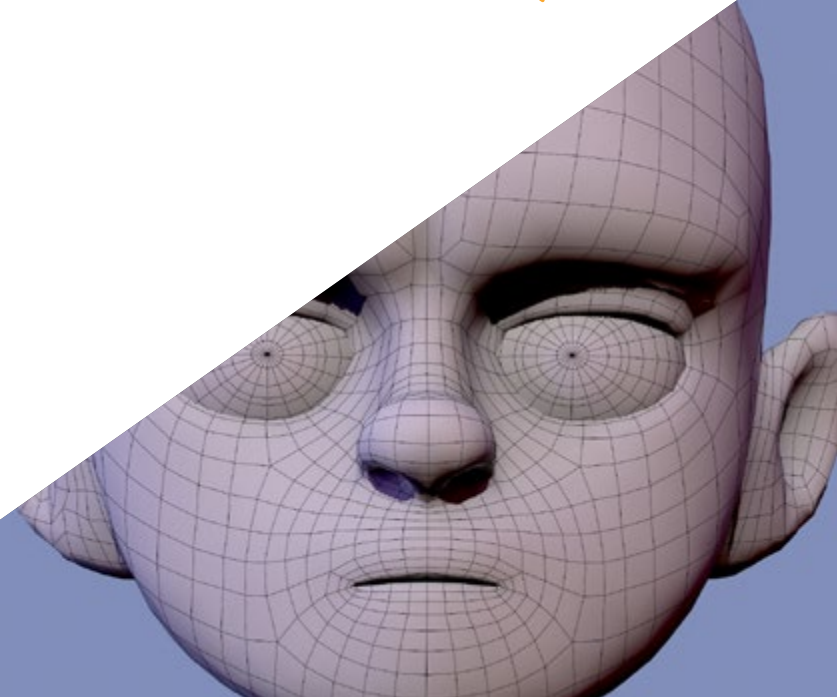
Incluye en su cuadro docente a profesionales pertenecientes al ámbito de la Programación de Videojuegos, que vierten en este programa la experiencia de su trabajo, además de reconocidos especialistas de sociedades de referencia y universidades de prestigio.

Su contenido multimedia, elaborado con la última tecnología educativa, permitirá al profesional un aprendizaje situado y contextual, es decir, un entorno simulado que proporcionará un estudio inmersivo programado para entrenarse ante situaciones reales.

El diseño de este programa se centra en el Aprendizaje Basado en Problemas, mediante el cual el alumno deberá tratar de resolver las distintas situaciones de práctica profesional que se le planteen a lo largo del curso académico. Para ello, el profesional contará con la ayuda de un novedoso sistema de vídeo interactivo realizado por reconocidos expertos.

Ahondarás en los módulos de esta titulación universitaria a través de la innovadora metodología Relearning, incorporando sus conceptos de manera rápida y flexible.

Aplicarás los algoritmos adecuados, optimizando el desarrollo de Videojuegos y adaptándolos a contextos de rendimiento.



02

¿Por qué estudiar en TECH?

TECH es la mayor Universidad digital del mundo. Con un impresionante catálogo de más de 14.000 programas universitarios, disponibles en 11 idiomas, se posiciona como líder en empleabilidad, con una tasa de inserción laboral del 99%. Además, cuenta con un enorme claustro de más de 6.000 profesores de máximo prestigio internacional.



“

Estudia en la mayor universidad digital del mundo y asegura tu éxito profesional. El futuro empieza en TECH”

La mejor universidad online del mundo según FORBES

La prestigiosa revista Forbes, especializada en negocios y finanzas, ha destacado a TECH como «la mejor universidad online del mundo». Así lo han hecho constar recientemente en un artículo de su edición digital en el que se hacen eco del caso de éxito de esta institución, «gracias a la oferta académica que ofrece, la selección de su personal docente, y un método de aprendizaje innovador orientado a formar a los profesionales del futuro».

El mejor claustro docente top internacional

El claustro docente de TECH está integrado por más de 6.000 profesores de máximo prestigio internacional. Catedráticos, investigadores y altos ejecutivos de multinacionales, entre los cuales se destacan Isaiah Covington, entrenador de rendimiento de los Boston Celtics; Magda Romanska, investigadora principal de MetaLAB de Harvard; Ignacio Wistuba, presidente del departamento de patología molecular traslacional del MD Anderson Cancer Center; o D.W Pine, director creativo de la revista TIME, entre otros.

La mayor universidad digital del mundo

TECH es la mayor universidad digital del mundo. Somos la mayor institución educativa, con el mejor y más amplio catálogo educativo digital, cien por cien online y abarcando la gran mayoría de áreas de conocimiento. Ofrecemos el mayor número de titulaciones propias, titulaciones oficiales de posgrado y de grado universitario del mundo. En total, más de 14.000 títulos universitarios, en once idiomas distintos, que nos convierten en la mayor institución educativa del mundo.



Forbes
Mejor universidad
online del mundo

Plan
de estudios
más completo

Profesorado
TOP
Internacional

La metodología
más eficaz

nº1
Mundial
Mayor universidad
online del mundo

Los planes de estudio más completos del panorama universitario

TECH ofrece los planes de estudio más completos del panorama universitario, con temarios que abarcan conceptos fundamentales y, al mismo tiempo, los principales avances científicos en sus áreas científicas específicas. Asimismo, estos programas son actualizados continuamente para garantizar al alumnado la vanguardia académica y las competencias profesionales más demandadas. De esta forma, los títulos de la universidad proporcionan a sus egresados una significativa ventaja para impulsar sus carreras hacia el éxito.

Un método de aprendizaje único

TECH es la primera universidad que emplea el *Relearning* en todas sus titulaciones. Se trata de la mejor metodología de aprendizaje online, acreditada con certificaciones internacionales de calidad docente, dispuestas por agencias educativas de prestigio. Además, este disruptivo modelo académico se complementa con el "Método del Caso", configurando así una estrategia de docencia online única. También en ella se implementan recursos didácticos innovadores entre los que destacan vídeos en detalle, infografías y resúmenes interactivos.

La universidad online oficial de la NBA

TECH es la universidad online oficial de la NBA. Gracias a un acuerdo con la mayor liga de baloncesto, ofrece a sus alumnos programas universitarios exclusivos, así como una gran variedad de recursos educativos centrados en el negocio de la liga y otras áreas de la industria del deporte. Cada programa tiene un currículo de diseño único y cuenta con oradores invitados de excepción: profesionales con una distinguida trayectoria deportiva que ofrecerán su experiencia en los temas más relevantes.

Líderes en empleabilidad

TECH ha conseguido convertirse en la universidad líder en empleabilidad. El 99% de sus alumnos obtienen trabajo en el campo académico que ha estudiado, antes de completar un año luego de finalizar cualquiera de los programas de la universidad. Una cifra similar consigue mejorar su carrera profesional de forma inmediata. Todo ello gracias a una metodología de estudio que basa su eficacia en la adquisición de competencias prácticas, totalmente necesarias para el desarrollo profesional.



Google Partner Premier

El gigante tecnológico norteamericano ha otorgado a TECH la insignia Google Partner Premier. Este galardón, solo al alcance del 3% de las empresas del mundo, pone en valor la experiencia eficaz, flexible y adaptada que esta universidad proporciona al alumno. El reconocimiento no solo acredita el máximo rigor, rendimiento e inversión en las infraestructuras digitales de TECH, sino que también sitúa a esta universidad como una de las compañías tecnológicas más punteras del mundo.



La universidad mejor valorada por sus alumnos

Los alumnos han posicionado a TECH como la universidad mejor valorada del mundo en los principales portales de opinión, destacando su calificación más alta de 4,9 sobre 5, obtenida a partir de más de 1.000 reseñas. Estos resultados consolidan a TECH como la institución universitaria de referencia a nivel internacional, reflejando la excelencia y el impacto positivo de su modelo educativo.



03

Plan de estudios

Este programa universitario destaca por ofrecer un enfoque técnico y actualizado que fortalece competencias clave en la Programación de Videojuegos. A lo largo del itinerario académico, se abordarán temas como los sistemas inteligentes, que permite diseñar comportamientos autónomos en entornos interactivos. Además, se explorarán las particularidades de la naturaleza del software aplicado a videojuegos, facilitando la construcción de arquitecturas funcionales. También se profundizará en el diseño para consolas y dispositivos especializados, lo que preparará al profesional para adaptar experiencias a distintas plataformas. Esta combinación de contenidos favorece una preparación integral orientada a la innovación y al alto desempeño técnico.





“

Profundizarás en los principios del software aplicado a Videojuegos, lo que permitirá desarrollar estructuras más eficientes y adaptadas a cada proyecto”

Módulo 1. Fundamentos de Programación

- 1.1. Introducción a la Programación
 - 1.1.1. Estructura básica de un ordenador
 - 1.1.2. Software
 - 1.1.3. Lenguajes de Programación
 - 1.1.4. Ciclo de vida de una aplicación informática
- 1.2. Diseño de algoritmos
 - 1.2.1. La resolución de problemas
 - 1.2.2. Técnicas descriptivas
 - 1.2.3. Elementos y estructura de un algoritmo
- 1.3. Elementos de un programa
 - 1.3.1. Origen y características del lenguaje C++
 - 1.3.2. El entorno de desarrollo
 - 1.3.3. Concepto de programa
 - 1.3.4. Tipos de datos fundamentales
 - 1.3.5. Operadores
 - 1.3.6. Expresiones
 - 1.3.7. Sentencias
 - 1.3.8. Entrada y salida de datos
- 1.4. Sentencias de control
 - 1.4.1. Sentencias
 - 1.4.2. Bifurcaciones
 - 1.4.3. Bucles
- 1.5. Abstracción y modularidad: funciones
 - 1.5.1. Diseño modular
 - 1.5.2. Concepto de función y utilidad
 - 1.5.3. Definición de una función
 - 1.5.4. Flujo de ejecución en la llamada de una función
 - 1.5.5. Prototipo de una función
 - 1.5.6. Devolución de resultados
 - 1.5.7. Llamada a una función: parámetros
 - 1.5.8. Paso de parámetros por referencia y por valor
 - 1.5.9. Ámbito identificador
- 1.6. Estructuras de datos estáticas
 - 1.6.1. *Arrays*
 - 1.6.2. Matrices. Poliedros
 - 1.6.3. Búsqueda y ordenación
 - 1.6.4. Cadenas. Funciones de E/S para cadenas
 - 1.6.5. Estructuras. Uniones
 - 1.6.6. Nuevos tipos de datos
- 1.7. Estructuras de datos dinámicas: punteros
 - 1.7.1. Concepto. Definición de puntero
 - 1.7.2. Operadores y operaciones con punteros
 - 1.7.3. *Arrays* de punteros
 - 1.7.4. Punteros y *arrays*
 - 1.7.5. Punteros a cadenas
 - 1.7.6. Punteros a estructuras
 - 1.7.7. Indirección múltiple
 - 1.7.8. Punteros a funciones
 - 1.7.9. Paso de funciones, estructuras y *arrays* como parámetros de funciones
- 1.8. Ficheros
 - 1.8.1. Conceptos básicos
 - 1.8.2. Operaciones con ficheros
 - 1.8.3. Tipos de ficheros
 - 1.8.4. Organización de los ficheros
 - 1.8.5. Introducción a los ficheros C++
 - 1.8.6. Manejo de ficheros
- 1.9. Recursividad
 - 1.9.1. Definición de recursividad
 - 1.9.2. Tipos de recursión
 - 1.9.3. Ventajas e inconvenientes
 - 1.9.4. Consideraciones
 - 1.9.5. Conversión recursivo - iterativa
 - 1.9.6. La pila de recursión

- 1.10. Prueba y documentación
 - 1.10.1. Pruebas de programas
 - 1.10.2. Prueba de la caja blanca
 - 1.10.3. Prueba de la caja negra
 - 1.10.4. Herramientas para realizar las pruebas
 - 1.10.5. Documentación de programas

Módulo 2. Estructura de Datos y Algoritmos

- 2.1. Introducción a las estrategias de diseño de algoritmos
 - 2.1.1. Recursividad
 - 2.1.2. Divide y conquista
 - 2.1.3. Otras estrategias
- 2.2. Eficiencia y análisis de los algoritmos
 - 2.2.1. Medidas de eficiencia
 - 2.2.2. Medir el tamaño de la entrada
 - 2.2.3. Medir el tiempo de ejecución
 - 2.2.4. Caso peor, mejor y medio
 - 2.2.5. Notación asintótica
 - 2.2.6. Criterios de Análisis matemático de algoritmos no recursivos
 - 2.2.7. Análisis matemático de algoritmos recursivos
 - 2.2.8. Análisis empírico de algoritmos
- 2.3. Algoritmos de ordenación
 - 2.3.1. Concepto de ordenación
 - 2.3.2. Ordenación de la burbuja
 - 2.3.3. Ordenación por selección
 - 2.3.4. Ordenación por inserción
 - 2.3.5. Ordenación por mezcla (merge_sort)
 - 2.3.6. Ordenación rápida (quick_sort)
- 2.4. Algoritmos con árboles
 - 2.4.1. Concepto de árbol
 - 2.4.2. Árboles binarios
 - 2.4.3. Recorridos de árbol
 - 2.4.4. Representar expresiones
 - 2.4.5. Árboles binarios ordenados
 - 2.4.6. Árboles binarios balanceados
- 2.5. Algoritmos con *heaps*
 - 2.5.1. Los *heaps*
 - 2.5.2. El algoritmo *heapsort*
 - 2.5.3. Las colas de prioridad
- 2.6. Algoritmos con grafos
 - 2.6.1. Representación
 - 2.6.2. Recorrido en anchura
 - 2.6.3. Recorrido en profundidad
 - 2.6.4. Ordenación topológica
- 2.7. Algoritmos *greedy*
 - 2.7.1. La estrategia *greedy*
 - 2.7.2. Elementos de la estrategia *greedy*
 - 2.7.3. Cambio de monedas
 - 2.7.4. Problema del viajante
 - 2.7.5. Problema de la mochila
- 2.8. Búsqueda de caminos mínimos
 - 2.8.1. El problema del camino mínimo
 - 2.8.2. Arcos negativos y ciclos
 - 2.8.3. Algoritmo de Dijkstra
- 2.9. Algoritmos *greedy* sobre grafos
 - 2.9.1. El árbol de recubrimiento mínimo
 - 2.9.2. El algoritmo de Prim
 - 2.9.3. El algoritmo de Kruskal
 - 2.9.4. Análisis de complejidad

2.10. *Backtracking*

2.10.1. El *backtracking*

2.10.2. Técnicas alternativas

Módulo 3. Programación Orientada a Objetos

3.1. Introducción a la programación orientada a objetos

3.1.1. Introducción a la programación orientada a objetos

3.1.2. Diseño de clases

3.1.3. Introducción a UML para el modelado de los problemas

3.2. Relaciones entre clases

3.2.1. Abstracción y herencia

3.2.2. Conceptos avanzados de herencia

3.2.3. Polimorfismo

3.2.4. Composición y agregación

3.3. Introducción a los patrones de diseño para problemas orientados a objetos

3.3.1. Qué son los patrones de diseño

3.3.2. Patrón Factory

3.3.4. Patrón Singleton

3.3.5. Patrón Observer

3.3.6. Patrón Composite

3.4. Excepciones

3.4.1. ¿Qué son las excepciones?

3.4.2. Captura y gestión de excepciones

3.4.3. Lanzamiento de excepciones

3.4.4. Creación de excepciones

3.5. Interfaces de usuarios

3.5.1. Introducción a Qt

3.5.2. Posicionamiento

3.5.3. ¿Qué son los eventos?

3.5.4. Eventos: definición y captura

3.5.5. Desarrollo de interfaces de usuario



- 3.6. Introducción a la programación concurrente
 - 3.6.1. Introducción a la programación concurrente
 - 3.6.2. El concepto de proceso e hilo
 - 3.6.3. Interacción entre procesos o hilos
 - 3.6.4. Los hilos en C++
 - 3.6.6. Ventajas e inconvenientes de la programación concurrente
- 3.7. Gestión de hilos y sincronización
 - 3.7.1. Ciclo de vida de un hilo
 - 3.7.2. La clase Thread
 - 3.7.3. Planificación de hilos
 - 3.7.4. Grupos hilos
 - 3.7.5. Hilos de tipo demonio
 - 3.7.6. Sincronización
 - 3.7.7. Mecanismos de bloqueo
 - 3.7.8. Mecanismos de comunicación
 - 3.7.9. Monitores
- 3.8. Problemas comunes dentro de la programación concurrente
 - 3.8.1. El problema de los productores consumidores
 - 3.8.2. El problema de los lectores y escritores
 - 3.8.3. El problema de la cena de los filósofos
- 3.9. Documentación y pruebas de software
 - 3.9.1. ¿Por qué es importante documentar el software?
 - 3.9.2. Documentación de diseño
 - 3.9.3. Uso de herramientas para la documentación
- 3.10. Pruebas de software
 - 3.10.1. Introducción a las pruebas del software
 - 3.10.2. Tipos de pruebas
 - 3.10.3. Prueba de unidad
 - 3.10.4. Prueba de integración
 - 3.10.5. Prueba de validación
 - 3.10.6. Prueba del sistema

Módulo 4. Consolas y Dispositivos para Videojuegos

- 4.1. Historia de la programación en videojuegos
 - 4.1.1. Periodo Atari (1977 - 1985)
 - 4.1.2. Periodo NES y SNES (1985 - 1995)
 - 4.1.3. Periodo PlayStation / PlayStation 2 (1995 - 2005)
 - 4.1.4. Periodo Xbox 360, PS3 y Wii (2005 - 2013)
 - 4.1.5. Periodo Xbox One, PS4 y Wii U – Switch (2013 - actualidad)
 - 4.1.6. El futuro
- 4.2. Historia de la jugabilidad en Videojuegos
 - 4.2.1. Introducción
 - 4.2.2. Contexto social
 - 4.2.3. Diagrama estructural
 - 4.2.4. Futuro
- 4.3. Adaptación a los tiempos modernos
 - 4.3.1. Juegos basados en movimiento
 - 4.3.2. Realidad virtual
 - 4.3.3. Realidad aumentada
 - 4.3.4. Realidad mixta
- 4.4. Unity: Scripting I y ejemplos
 - 4.4.1. ¿Qué es un script?
 - 4.4.2. Nuestro primer script
 - 4.4.3. Añadiendo un script
 - 4.4.4. Abriendo un script
 - 4.4.5. MonoBehaviour
 - 4.4.6. *Debugging*
- 4.5. Unity: Scripting II y ejemplos
 - 4.5.1. Entrada de teclado y ratón
 - 4.5.2. *Raycast*
 - 4.5.3. Instanciación
 - 4.5.4. Variables
 - 4.5.5. Variables públicas y serializadas

- 4.6. Unity: Scripting III y ejemplos
 - 4.6.1. Obteniendo componentes
 - 4.6.2. Modificando componentes
 - 4.6.3. Testeo
 - 4.6.4. Múltiples objetos
 - 4.6.5. *Colliders y triggers*
 - 4.6.6. Cuaterniones
- 4.7. Periféricos
 - 4.7.1. Evolución y clasificación
 - 4.7.2. Periféricos e interfaces
 - 4.7.3. Periféricos actuales
 - 4.7.4. Futuro próximo
- 4.8. Videojuegos: perspectivas futuras
 - 4.8.1. Juego basado en la nube
 - 4.8.2. Ausencia de controladores
 - 4.8.3. Realidad inmersiva
 - 4.8.4. Otras alternativas
- 4.9. Arquitectura
 - 4.9.1. Necesidades especiales de los videojuegos
 - 4.9.2. Evolución de la arquitectura
 - 4.9.3. Arquitectura actual
 - 4.9.4. Diferencias entre arquitecturas
- 4.10. Kits de desarrollo y su evolución
 - 4.10.1. Introducción
 - 4.10.2. Tercera generación de kits de desarrollo
 - 4.10.3. Cuarta generación de kits de desarrollo
 - 4.10.4. Quinta generación de kits de desarrollo
 - 4.10.5. Sexta generación de kits de desarrollo

Módulo 5. Ingeniería de Software

- 5.1. Introducción a la ingeniería del software y al modelado
 - 5.1.1. La naturaleza del software
 - 5.1.2. La naturaleza única de las *webapps*
 - 5.1.3. Ingeniería del software
 - 5.1.4. El proceso del software
 - 5.1.5. La práctica de la ingeniería del software
 - 5.1.6. Mitos del software
 - 5.1.7. Cómo comienza todo
 - 5.1.8. Conceptos orientados a objetos
 - 5.1.9. Introducción a UML
- 5.2. El proceso del software
 - 5.2.1. Un modelo general de proceso
 - 5.2.2. Modelos de proceso prescriptivos
 - 5.2.3. Modelos de proceso especializado
 - 5.2.4. El proceso unificado
 - 5.2.5. Modelos del proceso personal y del equipo
 - 5.2.6. ¿Qué es la agilidad?
 - 5.2.7. ¿Qué es un proceso ágil?
 - 5.2.8. Scrum
 - 5.2.9. Conjunto de herramientas para el proceso ágil
- 5.3. Principios que guían la práctica de la ingeniería del software
 - 5.3.1. Principios que guían el proceso
 - 5.3.2. Principios que guían la práctica
 - 5.3.3. Principios de comunicación
 - 5.3.4. Principios de planificación
 - 5.3.5. Principios de modelado
 - 5.3.6. Principios de construcción
 - 5.3.7. Principios de despliegue

- 5.4. Comprensión de los requisitos
 - 5.4.1. Ingeniería de requisitos
 - 5.4.2. Establecer las bases
 - 5.4.3. Indagación de los requisitos
 - 5.4.4. Desarrollo de casos de uso
 - 5.4.5. Elaboración del modelo de los requisitos
 - 5.4.6. Negociación de los requisitos
 - 5.4.7. Validación de los requisitos
- 5.5. Modelado de los requisitos: escenarios, información y clases de análisis
 - 5.5.1. Análisis de los requisitos
 - 5.5.2. Modelado basado en escenarios
 - 5.5.3. Modelos UML que proporcionan el caso de uso
 - 5.5.4. Conceptos de modelado de datos
 - 5.5.5. Modelado basado en clases
 - 5.5.6. Diagramas de clases
- 5.6. Modelado de los requisitos: flujo, comportamiento y patrones
 - 5.6.1. Requisitos que modelan las estrategias
 - 5.6.2. Modelado orientado al flujo
 - 5.6.3. Diagramas de estado
 - 5.6.4. Creación de un modelo de comportamiento
 - 5.6.5. Diagramas de secuencia
 - 5.6.6. Diagramas de comunicación
 - 5.6.7. Patrones para el modelado de requisitos
- 5.7. Conceptos de diseño
 - 5.7.1. Diseño en el contexto de la ingeniería del software
 - 5.7.2. El proceso de diseño
 - 5.7.3. Conceptos de diseño
 - 5.7.4. Conceptos de diseño orientado a objetos
 - 5.7.5. El modelo del diseño
- 5.8. Diseño de la arquitectura
 - 5.8.1. Arquitectura del software
 - 5.8.2. Géneros arquitectónicos
 - 5.8.3. Estilos arquitectónicos
 - 5.8.4. Diseño arquitectónico
 - 5.8.5. Evolución de los diseños alternativos para la arquitectura
 - 5.8.6. Mapeo de la arquitectura con el uso del flujo de datos
- 5.9. Diseño en el nivel de componentes y basado en patrones
 - 5.9.1. ¿Qué es un componente?
 - 5.9.2. Diseño de componentes basados en clase
 - 5.9.3. Realización del diseño en el nivel de componentes
 - 5.9.4. Diseño de componentes tradicionales
 - 5.9.5. Desarrollo basado en componentes
 - 5.9.6. Patrones de diseño
 - 5.9.7. Diseño de software basado en patrones
 - 5.9.8. Patrones arquitectónicos
 - 5.9.9. Patrones de diseño en el nivel de componentes
 - 5.9.10. Patrones de diseño de la interfaz de usuario
- 5.10. Calidad del software y administración de proyectos
 - 5.10.1. Calidad
 - 5.10.2. Calidad del software
 - 5.10.3. El dilema de la calidad del software
 - 5.10.4. Lograr la calidad del software
 - 5.10.5. Aseguramiento de la calidad del software
 - 5.10.6. El espectro administrativo
 - 5.10.7. El personal
 - 5.10.8. El producto
 - 5.10.9. El proceso
 - 5.10.10. El proyecto
 - 5.10.11. Principios y prácticas

Módulo 6. Motores de Videojuegos

- 6.1. Los videojuegos y las TICs
 - 6.1.1. Introducción
 - 6.1.2. Oportunidades
 - 6.1.3. Desafíos
 - 6.1.4. Conclusiones
- 6.2. Historia de los motores de videojuegos
 - 6.2.1. Introducción
 - 6.2.2. Época Atari
 - 6.2.3. Época de los 80
 - 6.2.4. Primeros motores. Época de los 90
 - 6.2.5. Motores actuales
- 6.3. Motores de videojuegos
 - 6.3.1. Tipos de motores
 - 6.3.2. Partes de un motor de videojuegos
 - 6.3.3. Motores actuales
 - 6.3.4. Selección de un motor para nuestro proyecto
- 6.4. Motor Game Maker
 - 6.4.1. Introducción
 - 6.4.2. Diseño de escenarios
 - 6.4.3. Sprites y animaciones
 - 6.4.4. Colisiones
 - 6.4.5. Scripting en GML
- 6.5. Motor Unreal Engine 4: Introducción
 - 6.5.1. ¿Qué es Unreal Engine 4? ¿Cuál es su filosofía?
 - 6.5.3. Materiales
 - 6.5.4. UI
 - 6.5.5. Animaciones
 - 6.5.6. Sistema de partículas
 - 6.5.7. Inteligencia artificial
 - 6.5.8. FPS
- 6.6. Motor Unreal Engine 4: Visual Scripting
 - 6.6.1. Filosofía de los Blueprints y el Visual Scripting
 - 6.6.2. Debugging
 - 6.6.3. Tipos de variables
 - 6.6.4. Control de flujo básico
- 6.7. Motor Unity 5
 - 6.7.1. Programación en C# y Visual Studio
 - 6.7.2. Creación de Prefabs
 - 6.7.3. Uso de Gizmos para el control del videojuego
 - 6.7.4. Motor adaptativo: 2D y 3D
- 6.8. Motor Godot
 - 6.8.1. Filosofía de diseño de Godot
 - 6.8.2. Diseño orientado a objetos y composición
 - 6.8.3. Todo incluido en un paquete
 - 6.8.4. Software libre y dirigido por la comunidad
- 6.9. Motor RPG Maker
 - 6.9.1. Filosofía de RPG Maker
 - 6.9.2. Tomando como referencia
 - 6.9.3. Crear un juego con personalidad
 - 6.9.4. Juegos comerciales de éxito
- 6.10. Motor Source 2
 - 6.10.1. Filosofía de Source 2
 - 6.10.2. Source y Source 2: Evolución
 - 6.10.3. Uso de la comunidad: Contenido audiovisual y videojuegos
 - 6.10.4. Futuro del motor Source 2
 - 6.10.5. Mods y juegos de éxito

Módulo 7. Sistemas Inteligentes

- 7.1. Teoría de Agentes
 - 7.1.1. Historia del concepto
 - 7.1.2. Definición de agente
 - 7.1.3. Agentes en Inteligencia Artificial
 - 7.1.4. Agentes en ingeniería de software

- 7.2. Arquitecturas de Agentes
 - 7.2.1. El proceso de razonamiento de un agente
 - 7.2.2. Agentes reactivos
 - 7.2.3. Agentes deductivos
 - 7.2.4. Agentes híbridos
 - 7.2.5. Comparativa
- 7.3. Información y conocimiento
 - 7.3.1. Distinción entre datos, información y conocimiento
 - 7.3.2. Evaluación de la calidad de los datos
 - 7.3.3. Métodos de captura de datos
 - 7.3.4. Métodos de adquisición de información
 - 7.3.5. Métodos de adquisición de conocimiento
- 7.4. Representación del conocimiento
 - 7.4.1. La importancia de la representación del conocimiento
 - 7.4.2. Definición de representación del conocimiento a través de sus roles
 - 7.4.3. Características de una representación del conocimiento
- 7.5. Ontologías
 - 7.5.1. Introducción a los metadatos
 - 7.5.2. Concepto filosófico de ontología
 - 7.5.3. Concepto informático de ontología
 - 7.5.4. Ontologías de dominio y ontologías de nivel superior
 - 7.5.5. Cómo construir una ontología
- 7.6. Lenguajes para ontologías y software para la creación de ontologías
 - 7.6.1. Tripletas RDF, Turtle y N3
 - 7.6.2. RDF Schema
 - 7.6.3. OWL
 - 7.6.4. SPARQL
 - 7.6.5. Introducción a las diferentes herramientas para la creación de ontologías
 - 7.6.6. Instalación y uso de Protégé
- 7.7. La web semántica
 - 7.7.1. El estado actual y futuro de la web semántica
 - 7.7.2. Aplicaciones de la web semántica

- 7.8. Otros modelos de representación del conocimiento
 - 7.8.1. Vocabularios
 - 7.8.2. Visión global
 - 7.8.3. Taxonomías
 - 7.8.4. Tesauros
 - 7.8.5. Folksonomías
 - 7.8.6. Comparativa
 - 7.8.7. Mapas mentales
- 7.9. Evaluación e integración de representaciones del conocimiento
 - 7.9.1. Lógica de orden cero
 - 7.9.2. Lógica de primer orden
 - 7.9.3. Lógica descriptiva
 - 7.9.4. Relación entre diferentes tipos de lógica
 - 7.9.5. Prolog: programación basada en lógica de primer orden
- 7.10. Razonadores semánticos, sistemas basados en conocimiento y sistemas expertos
 - 7.10.1. Concepto de razonador
 - 7.10.2. Aplicaciones de un razonador
 - 7.10.3. Sistemas basados en el conocimiento
 - 7.10.4. MYCIN, historia de los Sistemas Expertos
 - 7.10.5. Elementos y Arquitectura de Sistemas Expertos
 - 7.10.6. Creación de Sistemas Expertos

Módulo 8. Programación en Tiempo Real

- 8.1. Conceptos básicos de la programación concurrente
 - 8.1.1. Conceptos fundamentales
 - 8.1.2. Concurrencia
 - 8.1.3. Beneficios de la concurrencia
 - 8.1.4. Concurrencia y hardware
- 8.2. Estructuras básicas de soporte a la concurrencia en Java
 - 8.2.1. Concurrencia en Java
 - 8.2.2. Creación de threads
 - 8.2.3. Métodos
 - 8.2.4. Sincronización

- 8.3. Threads, ciclo de vida, prioridades, interrupciones, estados, ejecutores
 - 8.3.1. Threads
 - 8.3.2. Ciclo de vida
 - 8.3.3. Prioridades
 - 8.3.4. Interrupciones
 - 8.3.5. Estados
 - 8.3.6. Ejecutores
- 8.4. Exclusión mutua
 - 8.4.1. ¿Qué es la exclusión mutua?
 - 8.4.2. Algoritmo de Dekker
 - 8.4.3. Algoritmo de Peterson
 - 8.4.4. Exclusión mutua en Java
- 8.5. Dependencias de estados
 - 8.5.1. Inyección de dependencias
 - 8.5.2. Implementación del patrón en Java
 - 8.5.3. Formas de inyectar las dependencias
 - 8.5.4. Ejemplo
- 8.6. Patrones de diseño
 - 8.6.1. Introducción
 - 8.6.2. Patrones de creación
 - 8.6.3. Patrones de estructura
 - 8.6.4. Patrones de comportamiento
- 8.7. Uso de bibliotecas Java
 - 8.7.1. ¿Qué son las bibliotecas en Java?
 - 8.7.2. mockito - all, mockito - core
 - 8.7.3. guava
 - 8.7.4. commons - io
 - 8.7.5. commons - lang, commons - lang3
- 8.8. Programación de *shaders*
 - 8.8.1. Pipeline 3D y rasterizado
 - 8.8.2. Vertex Shading
 - 8.8.3. Pixel Shading: Iluminación I
 - 8.8.4. Pixel Shading: Iluminación II
 - 8.8.5. Post - efectos

- 8.9. Programación de tiempo real
 - 8.9.1. Introducción
 - 8.9.2. Procesamiento de interrupciones
 - 8.9.3. Sincronización y comunicación entre procesos
 - 8.9.4. Los sistemas de planificación en tiempo real
- 8.10. Planificación de tiempo real
 - 8.10.1. Conceptos
 - 8.10.2. Modelo de referencia de los sistemas de tiempo real
 - 8.10.3. Políticas de planificación
 - 8.10.4. Planificadores cíclicos
 - 8.10.5. Planificadores con propiedades estáticas
 - 8.10.6. Planificadores con propiedades dinámicas

Módulo 9. Diseño y Desarrollo de Juegos Web

- 9.1. Orígenes y estándares de la web
 - 9.1.1. Orígenes de Internet
 - 9.1.2. Creación de World Wide Web
 - 9.1.3. Aparición de los estándares web
 - 9.1.4. El auge de los estándares web
- 9.2. HTTP y estructura cliente - servidor
 - 9.2.1. Rol cliente - servidor
 - 9.2.2. Comunicación cliente - servidor
 - 9.2.3. Historia reciente
 - 9.2.4. Computación centralizada
- 9.3. Programación web: introducción
 - 9.3.1. Conceptos básicos
 - 9.3.2. Preparando un servidor web
 - 9.3.3. Conceptos básicos de HTML5
 - 9.3.4. Formas HTML
- 9.4. Introducción a HTML y ejemplos
 - 9.4.1. Historia de HTML5
 - 9.4.2. Elementos de HTML5
 - 9.4.3. APIS
 - 9.4.4. CCS3

- 9.5. Modelo de objeto de documento
 - 9.5.1. ¿Qué es el Modelo de Objetos del Documento?
 - 9.5.2. Uso de DOCTYPE
 - 9.5.3. La importancia de validar el HTML
 - 9.5.4. Accediendo a los elementos
 - 9.5.5. Creando elementos y textos
 - 9.5.6. Usando innerHTML
 - 9.5.7. Eliminando un elemento o nodo de texto
 - 9.5.8. Lectura y escritura de los atributos de un elemento
 - 9.5.9. Manipulando los estilos de los elementos
 - 9.5.10. Adjuntar múltiples ficheros a la vez
- 9.6. Introducción a CSS y ejemplos
 - 9.6.1. Sintaxis CSS3
 - 9.6.2. Hojas de estilo
 - 9.6.3. Etiquetas
 - 9.6.4. Selectores
 - 9.6.5. Diseño web con CSS
- 9.7. Introducción a JavaScript y ejemplos
 - 9.7.1. ¿Qué es JavaScript?
 - 9.7.2. Breve historia del lenguaje
 - 9.7.3. Versiones de JavaScript
 - 9.7.4. Mostrar un cuadro de diálogo
 - 9.7.5. Sintaxis de JavaScript
 - 9.7.6. Comprensión de scripts
 - 9.7.7. Espacios
 - 9.7.8. Comentarios
 - 9.7.9. Funciones
 - 9.7.10. JavaScript en la página y externo
- 9.8. Funciones en JavaScript
 - 9.8.1. Declaraciones de función
 - 9.8.2. Expresiones de función
 - 9.8.3. Llamar a funciones
 - 9.8.4. Recursividad
 - 9.8.5. Funciones anidadas y cierres

- 9.8.6. Preservación de variables
- 9.8.7. Funciones multi - anidadas
- 9.8.8. Conflictos de nombres
- 9.8.9. Clausuras o cierres
- 9.8.10. Parámetros de una función
- 9.9. PlayCanvas para desarrollar juegos web
 - 9.9.1. ¿Qué es PlayCanvas?
 - 9.9.2. Configuración del proyecto
 - 9.9.3. Creando un objeto
 - 9.9.4. Agregando físicas
 - 9.9.5. Añadiendo un modelo
 - 9.9.6. Cambiando los ajustes de gravedad y escena
 - 9.9.7. Ejecutando Scripts
 - 9.9.8. Controles de cámara
- 9.10. Phaser para desarrollar juegos web
 - 9.10.1. ¿Qué es Phaser?
 - 9.10.2. Cargando recursos
 - 9.10.3. Construyendo el mundo
 - 9.10.4. Las plataformas
 - 9.10.5. El jugador
 - 9.10.6. Añadir físicas
 - 9.10.7. Usar el teclado
 - 9.10.8. Recoger pickups
 - 9.10.9. Puntos y puntuación
 - 9.10.10. Bombas de rebote

Módulo 10. Redes y Sistemas Multijugador

- 10.1. Historia y evolución de Videojuegos multijugador
 - 10.1.1. Década 1970: Primeros juegos multijugador
 - 10.1.2. Años 90: Duke Nukem, Doom, Quake
 - 10.1.3. Auge de videojuegos multijugador
 - 10.1.4. Multijugador local y online
 - 10.1.5. Juegos de fiesta

10.2. Modelos de negocio multijugador

- 10.2.1. Origen y funcionamiento de los modelos de negocio emergentes
- 10.2.2. Servicios de venta en línea
- 10.2.3. Libre para jugar
- 10.2.4. Micropagos
- 10.2.5. Publicidad
- 10.2.6. Suscripción con pagos mensuales
- 10.2.7. Pagar por juego
- 10.2.8. Prueba antes de comprar

10.3. Juegos locales y juegos en red

- 10.3.1. Juegos locales: Inicios
- 10.3.2. Juegos de fiesta: nintendo y la unión de la familia
- 10.3.3. Juegos en red: Inicios
- 10.3.4. Evolución de los juegos en red

10.4. Modelo OSI: Capas I

- 10.4.1. Modelo OSI: Introducción
- 10.4.2. Capa física
- 10.4.3. Capa de enlace de datos
- 10.4.4. Capa de red

10.5. Modelo OSI: Capas II

- 10.5.1. Capa de transporte
- 10.5.2. Capa de sesión
- 10.5.3. Capa de presentación
- 10.5.4. Capa de aplicación

10.6. Redes de computadores e internet

- 10.6.1. ¿Qué es una red de computadoras?
- 10.6.2. Software
- 10.6.3. Hardware
- 10.6.4. Servidores
- 10.6.5. Almacenamiento en red
- 10.6.6. Protocolos de red





10.7. Redes móviles e inalámbricas

- 10.7.1. Red móvil
- 10.7.2. Red inalámbrica
- 10.7.3. Funcionamiento de las redes móviles
- 10.7.4. Tecnología digital

10.8. Seguridad

- 10.8.1. Seguridad personal
- 10.8.2. *Hacks* y *cheats* en videojuegos
- 10.8.3. Seguridad anti - trampas
- 10.8.4. Análisis de sistemas de seguridad anti - trampas

10.9. Sistemas multijugador: servidores

- 10.9.1. Alojamiento de servidores
- 10.9.2. Videojuegos MMO
- 10.9.3. Servidores de videojuegos dedicados
- 10.9.4. LAN Parties

10.10. Diseño de videojuegos multijugador y programación

- 10.10.1. Fundamentos de diseño de videojuegos multijugador en Unreal
- 10.10.2. Fundamentos de diseño de videojuegos multijugador en Unity
- 10.10.3. Como hacer que un juego multijugador sea divertido
- 10.10.4. Más allá de un mando: Innovación en controles multijugador

“Ampliarás tu perspectiva sobre el diseño de sistemas multijugador, integrando con precisión aspectos como la sincronización”

04

Objetivos docentes

Este programa universitario tiene como meta principal proporcionar las competencias necesarias para desenvolverse con solvencia en el desarrollo de videojuegos, desde una perspectiva técnica y especializada. A lo largo del recorrido académico, se abordarán conceptos como la estructura básica de un ordenador y los elementos que componen un programa, lo cual permite comprender el funcionamiento interno de los sistemas y su aplicación en entornos interactivos. Asimismo, se reforzarán habilidades para optimizar procesos, gestionar recursos informáticos y diseñar soluciones adaptadas a distintos dispositivos. De esta manera, se impulsará una preparación orientada a la innovación y a la creación de experiencias digitales.



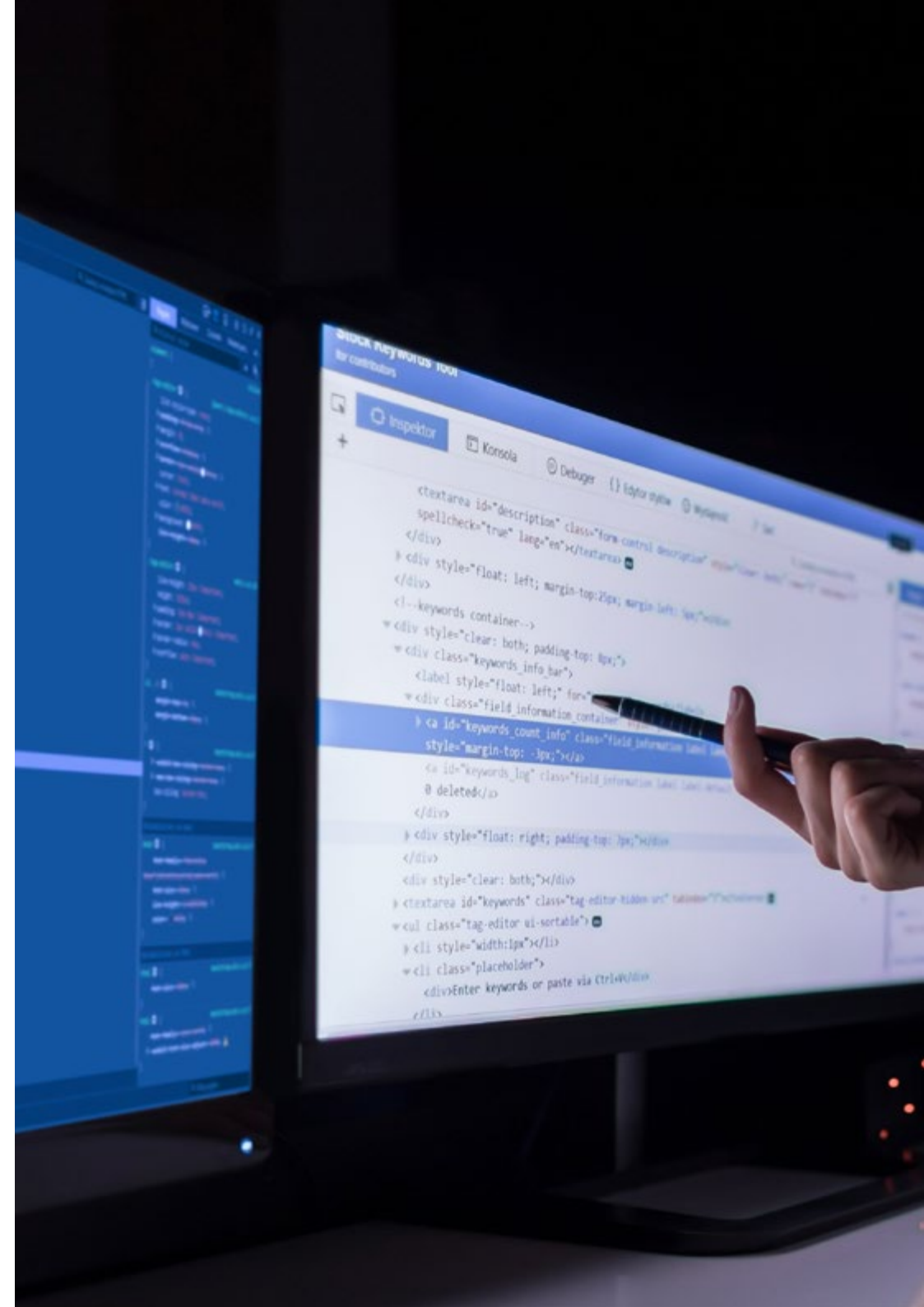
A close-up, high-detail image of a character's face, likely from a video game. The character has a thick, dark beard and mustache. The skin is pale with visible texture and several small, bloody wounds or scratches, particularly around the eye and cheek area. The character's eyes are visible, looking slightly to the side. The image is partially obscured by a diagonal split in the background, which transitions from a dark teal to a lighter teal and then to white.

Diseñarás e implementarás arquitecturas de software escalables y eficientes para videojuegos multiplataforma”



Objetivos generales

- ♦ Desarrollar competencias técnicas para implementar soluciones funcionales en proyectos de Videojuegos
- ♦ Comprender los principios de la programación orientada a objetos aplicados al diseño de entornos interactivos
- ♦ Aplicar estructuras de datos y algoritmos para optimizar el rendimiento de sistemas digitales
- ♦ Integrar conocimientos sobre consolas y dispositivos para adaptar experiencias a distintas plataformas
- ♦ Interpretar los fundamentos de la ingeniería de software enfocados en Videojuegos
- ♦ Dominar el uso de motores de videojuegos para crear entornos gráficos eficientes
- ♦ Explorar sistemas inteligentes para mejorar la autonomía de personajes virtuales
- ♦ Implementar técnicas de programación en tiempo real y entornos multijugador





Objetivos específicos

Módulo 1. Fundamentos de Programación

- ♦ Identificar los componentes esenciales de un ordenador y su relación con los lenguajes de Programación
- ♦ Diseñar algoritmos eficientes utilizando técnicas descriptivas y estructuras lógicas claras
- ♦ Aplicar principios de modularidad y abstracción mediante el uso de funciones en C++
- ♦ Implementar estructuras de datos estáticas y dinámicas para resolver problemas computacionales

Módulo 2. Estructura de Datos y Algoritmos

- ♦ Emplear estrategias de diseño algorítmico como divide y vencerás, recursividad y técnicas *greedy*
- ♦ Evaluar la eficiencia de los algoritmos mediante análisis matemático y empírico.
- ♦ Implementar algoritmos de ordenación, árboles, *heaps* y grafos en contextos computacionales
- ♦ Resolver problemas complejos utilizando técnicas como búsqueda de caminos mínimos y *backtracking*

Módulo 3. Programación Orientada a Objetos

- ♦ Implementar estructuras orientadas a objetos mediante clases, herencia y polimorfismo
- ♦ Aplicar patrones de diseño para resolver problemas recurrentes en el desarrollo de software
- ♦ Desarrollar interfaces gráficas funcionales utilizando herramientas como Qt
- ♦ Gestionar procesos concurrentes a través de hilos, sincronización y mecanismos de comunicación

Módulo 4. Consolas y Dispositivos para Videojuegos

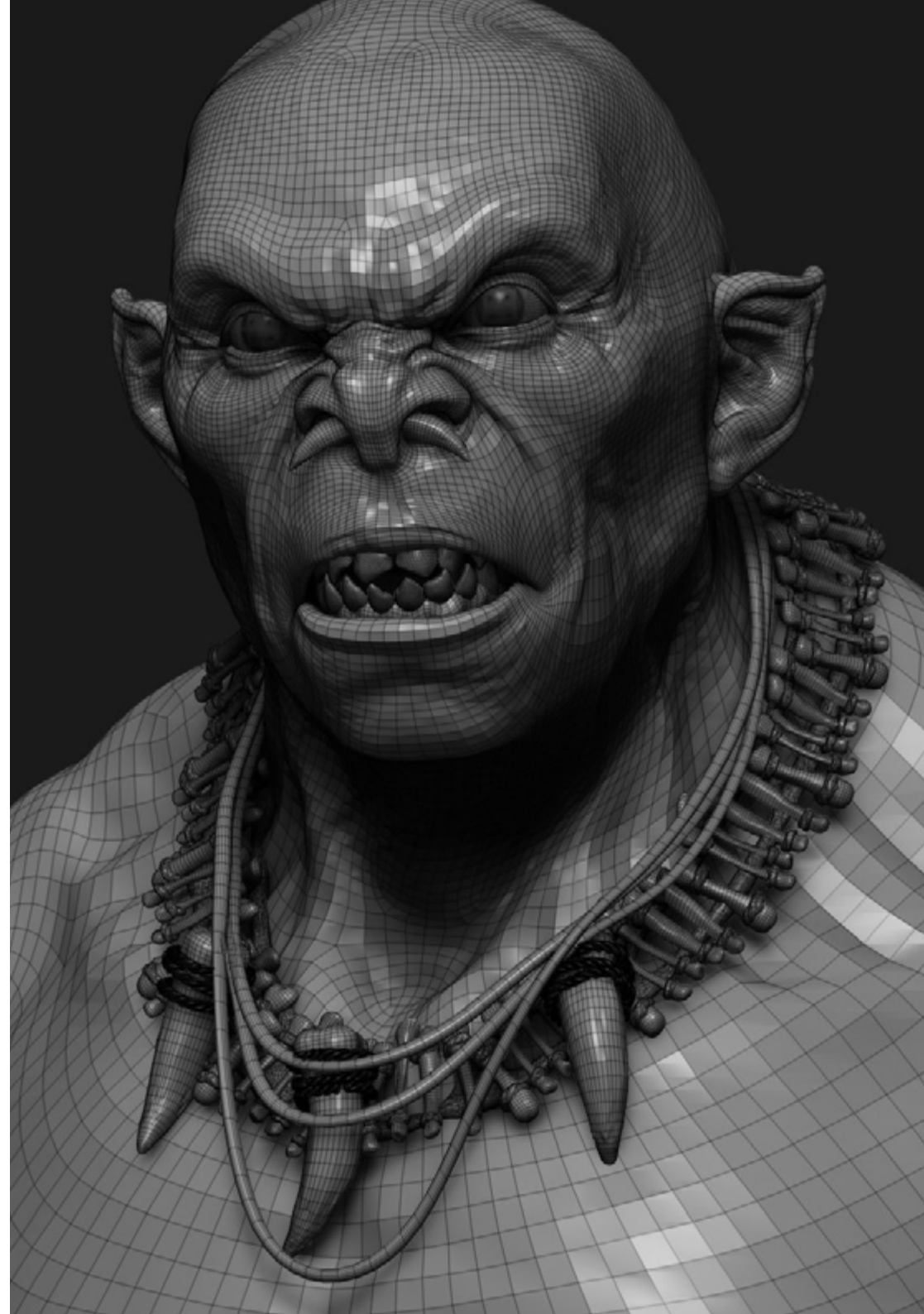
- ♦ Analizar la evolución histórica de las consolas y su impacto en la programación de videojuegos
- ♦ Aplicar técnicas de *scripting* en Unity para el desarrollo de videojuegos interactivos
- ♦ Desarrollar aplicaciones para periféricos y dispositivos de entrada en Videojuegos
- ♦ Explorar las tendencias futuras en la arquitectura de videojuegos y los kits de desarrollo

Módulo 5. Ingeniería de Software

- ♦ Aplicar los principios de la ingeniería del software en el diseño y desarrollo de aplicaciones
- ♦ Analizar y utilizar modelos de procesos ágiles y prescriptivos en la gestión de proyectos de software
- ♦ Implementar técnicas de modelado de requisitos y diseño utilizando UML y diagramas de clases
- ♦ Desarrollar soluciones de diseño arquitectónico basadas en estilos y géneros arquitectónicos
- ♦ Aplicar patrones de diseño en componentes de software y en la interfaz de usuario
- ♦ Evaluar la calidad del software a través de prácticas de aseguramiento y gestión de proyectos

Módulo 6. Motores de Videojuegos

- ♦ Analizar los diferentes tipos de motores de Videojuegos y su evolución a lo largo del tiempo
- ♦ Desarrollar y diseñar escenarios interactivos utilizando motores como Unreal Engine 4, Unity 5, y Godot
- ♦ Aplicar técnicas de programación y *scripting* en motores de Videojuegos, incluyendo GML y C#
- ♦ Explorar y seleccionar motores de Videojuegos adecuados para proyectos específicos, considerando sus características y filosofías de diseño



Módulo 7. Sistemas Inteligentes

- ♦ Explorar y analizar las diferentes arquitecturas de agentes, incluyendo agentes reactivos, deductivos y híbridos
- ♦ Evaluar métodos de adquisición y representación de conocimiento, así como la importancia de la calidad de los datos
- ♦ Desarrollar ontologías utilizando herramientas y lenguajes como RDF, OWL y SPARQL para representar conocimiento
- ♦ Aplicar razonadores semánticos y sistemas expertos para resolver problemas complejos mediante la integración de representaciones del conocimiento

Módulo 8. Programación en Tiempo Real

- ♦ Explorar los conceptos fundamentales de la programación concurrente, analizando los beneficios de la concurrencia y su relación con el hardware
- ♦ Desarrollar aplicaciones en Java utilizando estructuras básicas de soporte a la concurrencia, como *threads* y sincronización
- ♦ Implementar algoritmos de exclusión mutua, como los de Dekker y Peterson, para garantizar la integridad en entornos concurrentes
- ♦ Aplicar patrones de diseño, como los de creación, estructura y comportamiento, para optimizar la programación de sistemas en tiempo real

Módulo 9. Diseño y Desarrollo de Juegos Web

- ♦ Implementar motores de Videojuegos en proyectos interactivos, usando Unity, Unreal Engine y Godot
- ♦ Identificar características clave de motores de Videojuegos para elegir el adecuado para cada proyecto
- ♦ Desarrollar Videojuegos con animaciones, modelado 3D y físicas, optimizando el rendimiento
- ♦ Crear experiencias interactivas completas, desde el diseño hasta la programación de mecánicas de juego

Módulo 10. Redes y Sistemas Multijugador

- ♦ Desarrollar y gestionar servidores multijugador para juegos online, mejorando la conectividad y la experiencia de usuario
- ♦ Aplicar el modelo OSI para optimizar las comunicaciones en redes de Videojuegos, garantizando un flujo de datos eficiente
- ♦ Implementar sistemas de seguridad robustos contra trampas y *hacks* en Videojuegos multijugador, protegiendo la integridad del juego
- ♦ Diseñar experiencias multijugador innovadoras en Unity y Unreal Engine, optimizando jugabilidad y la interacción entre usuarios



Fortalecerás tus exclusivos conocimientos en la utilización de herramientas y lenguajes como RDF para la gestión de datos en sistemas semánticos”

05

Salidas profesionales

Este Máster Título Propio capacitará a los profesionales para asumir roles clave en la industria de la Programación de Videojuegos, como desarrolladores, diseñadores de Videojuegos y líderes de equipos en empresas tecnológicas, estudios de desarrollo y compañías de entretenimiento digital. A través de una capacitación avanzada, el alumnado estará preparados para integrar tecnologías emergentes y enfrentar los retos más complejos en la creación de Videojuegos, destacándose en un entorno competitivo y con alto potencial de innovación. Con estos conocimientos, podrán acceder a puestos estratégicos en compañías de renombre o incluso emprender proyectos propios.





“

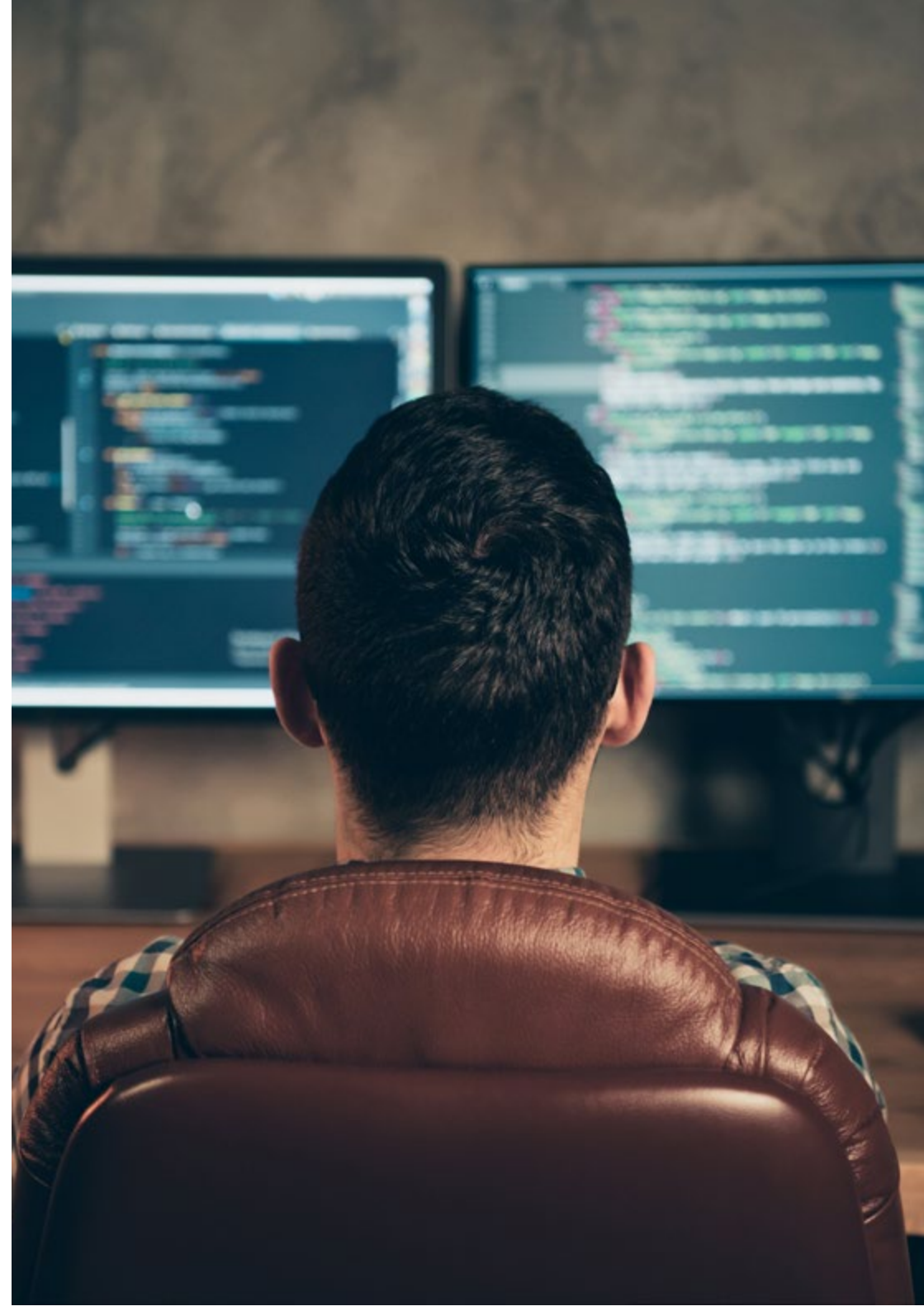
Este Máster Título Propio te proporcionará un enfoque flexible que te impulsará a destacar en compañías de entretenimiento digital”

Perfil del egresado

El egresado será un profesional altamente capacitado para liderar proyectos innovadores en la industria del entretenimiento digital. De hecho, contará con habilidades avanzadas en programación, diseño de interfaces y creación de experiencias inmersivas. Además, dominará las tendencias tecnológicas emergentes, como la realidad aumentada aplicada a los Videojuegos. Como resultado, podrá desempeñarse como líder de equipos creativos, gestionar estudios de desarrollo o colaborar en la optimización de plataformas digitales, destacándose por su capacidad para transformar ideas en soluciones interactivas y altamente atractivas para los usuarios.

Asumirás roles especializados en la creación de experiencias inmersivas, dominando el diseño y desarrollo de entornos virtuales interactivos.

- ♦ **Trabajo en equipo multidisciplinario:** colaborar eficazmente con profesionales de diversas áreas, como diseñadores, programadores y artistas, para crear experiencias inmersivas que integren múltiples perspectivas y habilidades
- ♦ **Gestión del tiempo y proyectos:** organizar y gestionar proyectos de manera eficiente, cumpliendo plazos y alcanzando objetivos, tanto en el desarrollo individual como en el trabajo colaborativo
- ♦ **Pensamiento crítico y resolución de problemas:** identificar problemas complejos dentro del desarrollo de experiencias inmersivas y aplicar soluciones innovadoras, tomando decisiones informadas y estratégicas
- ♦ **Adaptabilidad tecnológica:** aprender rápidamente nuevas herramientas y tecnologías emergentes, adaptándose a los cambios en el campo de la creación de experiencias inmersivas para mantener un alto nivel de competencia profesional



Después de realizar el programa universitario, podrás desempeñar tus conocimientos y habilidades en los siguientes cargos:

1. Desarrollador de videojuegos: encargado de diseñar, programar y optimizar videojuegos. Trabaja con diferentes lenguajes de programación y plataformas para crear experiencias interactivas y atractivas para los usuarios.

2. Diseñador de experiencias inmersivas: dedicado a la creación de entornos virtuales interactivos, utilizando tecnologías como realidad virtual, aumentada y simulaciones 3D para ofrecer experiencias envolventes.

3. Productor de Videojuegos: responsable de la supervisión general de los proyectos de videojuegos, desde la planificación hasta la ejecución. Coordina equipos, gestiona plazos y asegura que los recursos se utilicen eficientemente.

4. Ingeniero de sonido para Videojuegos: encargado de la creación, implementación y gestión de efectos de sonido y música dentro de los videojuegos, asegurando una experiencia auditiva de alta calidad.

5. Analista de datos de Videojuegos: dedicado a la recopilación y análisis de datos sobre el comportamiento de los jugadores dentro de los videojuegos, con el fin de optimizar la jugabilidad y la experiencia del usuario.

6. Gerente de proyecto de Videojuegos: responsable de la planificación, ejecución y seguimiento de los proyectos de Videojuegos. Administra recursos, cronogramas y equipos para garantizar que los productos sean entregados dentro del plazo y presupuesto establecidos.

7. Artista 3D para Videojuegos: responsable de la creación de modelos tridimensionales, texturas y entornos dentro de los Videojuegos, trabajando con software especializado para dar vida a los mundos virtuales y personajes del juego.

8. Director creativo de Videojuegos: líder del equipo de diseño y concepto del videojuego, encargado de definir la visión artística y narrativa del juego. Asegura que todos los aspectos del proyecto estén alineados con la propuesta original.

9. Desarrollador de realidad aumentada/virtual: encargado de diseñar, desarrollar e implementar aplicaciones en realidad aumentada o virtual. Crea experiencias inmersivas para videojuegos, simulaciones o entrenamientos virtuales.



Te especializarás como un desarrollador de Videojuegos potencial, adquiriendo una titulación que combina creatividad e innovación”

06

Licencias de software incluidas

TECH es referencia en el mundo universitario por combinar la última tecnología con las metodologías docentes para potencial el proceso de enseñanza-aprendizaje. Para ello, ha establecido una red de alianzas que le permite tener acceso a las herramientas de *software* más avanzadas del mundo profesional.



“

Al matricularte recibirás, de forma completamente gratuita, las credenciales de uso académico de las siguientes aplicaciones de software profesional”

TECH ha establecido una red de alianzas profesionales en la que se encuentran los principales proveedores de software aplicado a las diferentes áreas profesionales. Estas alianzas permiten a TECH tener acceso al uso de centenares de aplicaciones informáticas y licencias de software para acercarlas a sus estudiantes.

Las licencias de software para uno académico permitirán a los estudiantes utilizar las aplicaciones informáticas más avanzadas en su área profesional, de modo que podrán conocerlas y aprender su dominio sin tener que incurrir en costes. TECH se hará cargo del procedimiento de contratación para que los alumnos puedan utilizarlas de modo ilimitado durante el tiempo que estén estudiando el programa de Máster Título Propio en Programación de Videojuegos, y además lo podrán hacer de forma completamente gratuita.

TECH te dará acceso gratuito al uso de las siguientes aplicaciones de software:



Arnold

Unity es una plataforma líder en desarrollo de software para la creación de experiencias interactivas y multimedia en 2D y 3D. Durante el transcurso de este programa universitario, los alumnos tendrán **acceso gratuito** a esta plataforma, cuyo valor aproximado es de **2040 dólares**, permitiendo así trabajar con una solución profesional **sin coste adicional**.

Esta herramienta ofrece un motor en tiempo real que soporta gráficos con renderizado de alta calidad, además de potentes herramientas de scripting que facilitan la personalización avanzada. **Unity** incluye un sistema de física integrado para simulaciones realistas, soporte multiplataforma para dispositivos móviles, consolas y tecnologías VR/AR y una amplia Asset Store con recursos que aceleran el desarrollo.

3ds Max

Durante la realización del programa, TECH pone a disposición de los egresados la licencia oficial de **3ds Max**, valorada en **2.300 euros**, **sin coste** adicional. Esta herramienta destaca en áreas como diseño arquitectónico, animación digital y simulación visual, y permite trabajar con tecnología líder del sector profesional.

Esta plataforma ofrece un entorno robusto para modelar, animar y renderizar proyectos complejos con precisión y eficiencia. Gracias a su arquitectura flexible, los usuarios pueden desarrollar desde visualizaciones estáticas hasta escenas completas de animación, utilizando funciones avanzadas en un espacio optimizado para resultados de alto nivel.



Google Career Launchpad

Google Career Launchpad es una solución para desarrollar habilidades digitales en tecnología y análisis de datos. Con un valor estimado de **5.000 dólares**, se incluye de forma **gratuita** en el programa universitario de TECH, brindando acceso a laboratorios interactivos y certificaciones reconocidas en el sector.

Esta plataforma combina capacitación técnica con casos prácticos, usando tecnologías como BigQuery y Google AI. Ofrece entornos simulados para experimentar con datos reales, junto a una red de expertos para orientación personalizada.

Maya

Durante este programa universitario, los egresados tendrán acceso gratis a **Maya**, una potente herramienta profesional valorada en **2.250 euros**. Se emplea ampliamente en la industria audiovisual para desarrollar animaciones 3D, modelado, simulación y renderizado, siendo esencial en cine, televisión y videojuegos de alto nivel.

Esta plataforma permite construir entornos y personajes con alto nivel de detalle, gestionar efectos visuales complejos y ejecutar procesos avanzados de animación. Su presencia en este programa refuerza las capacidades técnicas en contextos reales, favoreciendo la inserción laboral con recursos utilizados por estudios internacionales líderes en contenido digital.



Gracias a TECH podrás utilizar gratuitamente las mejores aplicaciones de software de tu área profesional"

07

Metodología de estudio

TECH es la primera universidad en el mundo que combina la metodología de los **case studies** con el **Relearning**, un sistema de aprendizaje 100% online basado en la reiteración dirigida.

Esta disruptiva estrategia pedagógica ha sido concebida para ofrecer a los profesionales la oportunidad de actualizar conocimientos y desarrollar competencias de un modo intensivo y riguroso. Un modelo de aprendizaje que coloca al estudiante en el centro del proceso académico y le otorga todo el protagonismo, adaptándose a sus necesidades y dejando de lado las metodologías más convencionales.



“

TECH te prepara para afrontar nuevos retos en entornos inciertos y lograr el éxito en tu carrera”

El alumno: la prioridad de todos los programas de TECH

En la metodología de estudios de TECH el alumno es el protagonista absoluto. Las herramientas pedagógicas de cada programa han sido seleccionadas teniendo en cuenta las demandas de tiempo, disponibilidad y rigor académico que, a día de hoy, no solo exigen los estudiantes sino los puestos más competitivos del mercado.

Con el modelo educativo asincrónico de TECH, es el alumno quien elige el tiempo que destina al estudio, cómo decide establecer sus rutinas y todo ello desde la comodidad del dispositivo electrónico de su preferencia. El alumno no tendrá que asistir a clases en vivo, a las que muchas veces no podrá acudir. Las actividades de aprendizaje las realizará cuando le venga bien. Siempre podrá decidir cuándo y desde dónde estudiar.

“

*En TECH NO tendrás clases en directo
(a las que luego nunca puedes asistir)”*



Los planes de estudios más exhaustivos a nivel internacional

TECH se caracteriza por ofrecer los itinerarios académicos más completos del entorno universitario. Esta exhaustividad se logra a través de la creación de temarios que no solo abarcan los conocimientos esenciales, sino también las innovaciones más recientes en cada área.

Al estar en constante actualización, estos programas permiten que los estudiantes se mantengan al día con los cambios del mercado y adquieran las habilidades más valoradas por los empleadores. De esta manera, quienes finalizan sus estudios en TECH reciben una preparación integral que les proporciona una ventaja competitiva notable para avanzar en sus carreras.

Y además, podrán hacerlo desde cualquier dispositivo, pc, tableta o smartphone.

“

El modelo de TECH es asincrónico, de modo que te permite estudiar con tu pc, tableta o tu smartphone donde quieras, cuando quieras y durante el tiempo que quieras”

Case studies o Método del caso

El método del caso ha sido el sistema de aprendizaje más utilizado por las mejores escuelas de negocios del mundo. Desarrollado en 1912 para que los estudiantes de Derecho no solo aprendiesen las leyes a base de contenidos teóricos, su función era también presentarles situaciones complejas reales. Así, podían tomar decisiones y emitir juicios de valor fundamentados sobre cómo resolverlas. En 1924 se estableció como método estándar de enseñanza en Harvard.

Con este modelo de enseñanza es el propio alumno quien va construyendo su competencia profesional a través de estrategias como el *Learning by doing* o el *Design Thinking*, utilizadas por otras instituciones de renombre como Yale o Stanford.

Este método, orientado a la acción, será aplicado a lo largo de todo el itinerario académico que el alumno emprenda junto a TECH. De ese modo se enfrentará a múltiples situaciones reales y deberá integrar conocimientos, investigar, argumentar y defender sus ideas y decisiones. Todo ello con la premisa de responder al cuestionamiento de cómo actuaría al posicionarse frente a eventos específicos de complejidad en su labor cotidiana.



Método Relearning

En TECH los *case studies* son potenciados con el mejor método de enseñanza 100% online: el *Relearning*.

Este método rompe con las técnicas tradicionales de enseñanza para poner al alumno en el centro de la ecuación, proveyéndole del mejor contenido en diferentes formatos. De esta forma, consigue repasar y reiterar los conceptos clave de cada materia y aprender a aplicarlos en un entorno real.

En esta misma línea, y de acuerdo a múltiples investigaciones científicas, la reiteración es la mejor manera de aprender. Por eso, TECH ofrece entre 8 y 16 repeticiones de cada concepto clave dentro de una misma lección, presentada de una manera diferente, con el objetivo de asegurar que el conocimiento sea completamente afianzado durante el proceso de estudio.

El Relearning te permitirá aprender con menos esfuerzo y más rendimiento, implicándote más en tu especialización, desarrollando el espíritu crítico, la defensa de argumentos y el contraste de opiniones: una ecuación directa al éxito.



Un Campus Virtual 100% online con los mejores recursos didácticos

Para aplicar su metodología de forma eficaz, TECH se centra en proveer a los egresados de materiales didácticos en diferentes formatos: textos, vídeos interactivos, ilustraciones y mapas de conocimiento, entre otros. Todos ellos, diseñados por profesores cualificados que centran el trabajo en combinar casos reales con la resolución de situaciones complejas mediante simulación, el estudio de contextos aplicados a cada carrera profesional y el aprendizaje basado en la reiteración, a través de audios, presentaciones, animaciones, imágenes, etc.

Y es que las últimas evidencias científicas en el ámbito de las Neurociencias apuntan a la importancia de tener en cuenta el lugar y el contexto donde se accede a los contenidos antes de iniciar un nuevo aprendizaje. Poder ajustar esas variables de una manera personalizada favorece que las personas puedan recordar y almacenar en el hipocampo los conocimientos para retenerlos a largo plazo. Se trata de un modelo denominado *Neurocognitive context-dependent e-learning* que es aplicado de manera consciente en esta titulación universitaria.

Por otro lado, también en aras de favorecer al máximo el contacto mentor-alumno, se proporciona un amplio abanico de posibilidades de comunicación, tanto en tiempo real como en diferido (mensajería interna, foros de discusión, servicio de atención telefónica, email de contacto con secretaría técnica, chat y videoconferencia).

Asimismo, este completísimo Campus Virtual permitirá que el alumnado de TECH organice sus horarios de estudio de acuerdo con su disponibilidad personal o sus obligaciones laborales. De esa manera tendrá un control global de los contenidos académicos y sus herramientas didácticas, puestas en función de su acelerada actualización profesional.



La modalidad de estudios online de este programa te permitirá organizar tu tiempo y tu ritmo de aprendizaje, adaptándolo a tus horarios”

La eficacia del método se justifica con cuatro logros fundamentales:

1. Los alumnos que siguen este método no solo consiguen la asimilación de conceptos, sino un desarrollo de su capacidad mental, mediante ejercicios de evaluación de situaciones reales y aplicación de conocimientos.
2. El aprendizaje se concreta de una manera sólida en capacidades prácticas que permiten al alumno una mejor integración en el mundo real.
3. Se consigue una asimilación más sencilla y eficiente de las ideas y conceptos, gracias al planteamiento de situaciones que han surgido de la realidad.
4. La sensación de eficiencia del esfuerzo invertido se convierte en un estímulo muy importante para el alumnado, que se traduce en un interés mayor en los aprendizajes y un incremento del tiempo dedicado a trabajar en el curso.

La metodología universitaria mejor valorada por sus alumnos

Los resultados de este innovador modelo académico son constatables en los niveles de satisfacción global de los egresados de TECH.

La valoración de los estudiantes sobre la calidad docente, calidad de los materiales, estructura del curso y sus objetivos es excelente. No en valde, la institución se convirtió en la universidad mejor valorada por sus alumnos según el índice global score, obteniendo un 4,9 de 5.

Accede a los contenidos de estudio desde cualquier dispositivo con conexión a Internet (ordenador, tablet, smartphone) gracias a que TECH está al día de la vanguardia tecnológica y pedagógica.

Podrás aprender con las ventajas del acceso a entornos simulados de aprendizaje y el planteamiento de aprendizaje por observación, esto es, Learning from an expert.

Así, en este programa estarán disponibles los mejores materiales educativos, preparados a conciencia:



Material de estudio

Todos los contenidos didácticos son creados por los especialistas que van a impartir el curso, específicamente para él, de manera que el desarrollo didáctico sea realmente específico y concreto.

Estos contenidos son aplicados después al formato audiovisual que creará nuestra manera de trabajo online, con las técnicas más novedosas que nos permiten ofrecerte una gran calidad, en cada una de las piezas que pondremos a tu servicio.



Prácticas de habilidades y competencias

Realizarás actividades de desarrollo de competencias y habilidades específicas en cada área temática. Prácticas y dinámicas para adquirir y desarrollar las destrezas y habilidades que un especialista precisa desarrollar en el marco de la globalización que vivimos.



Resúmenes interactivos

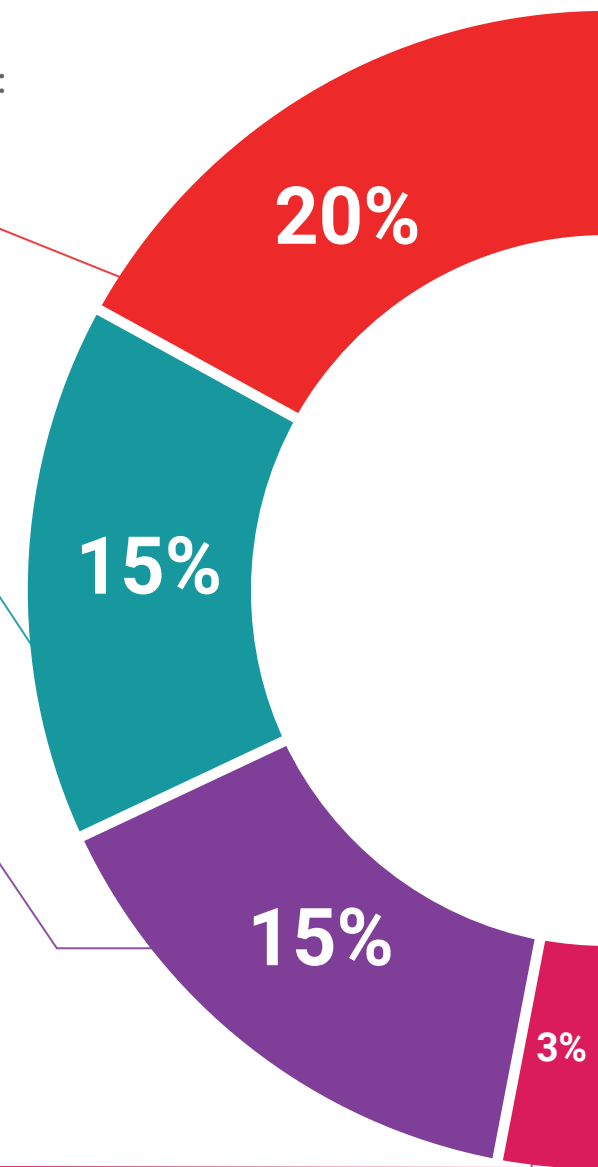
Presentamos los contenidos de manera atractiva y dinámica en píldoras multimedia que incluyen audio, vídeos, imágenes, esquemas y mapas conceptuales con el fin de afianzar el conocimiento.

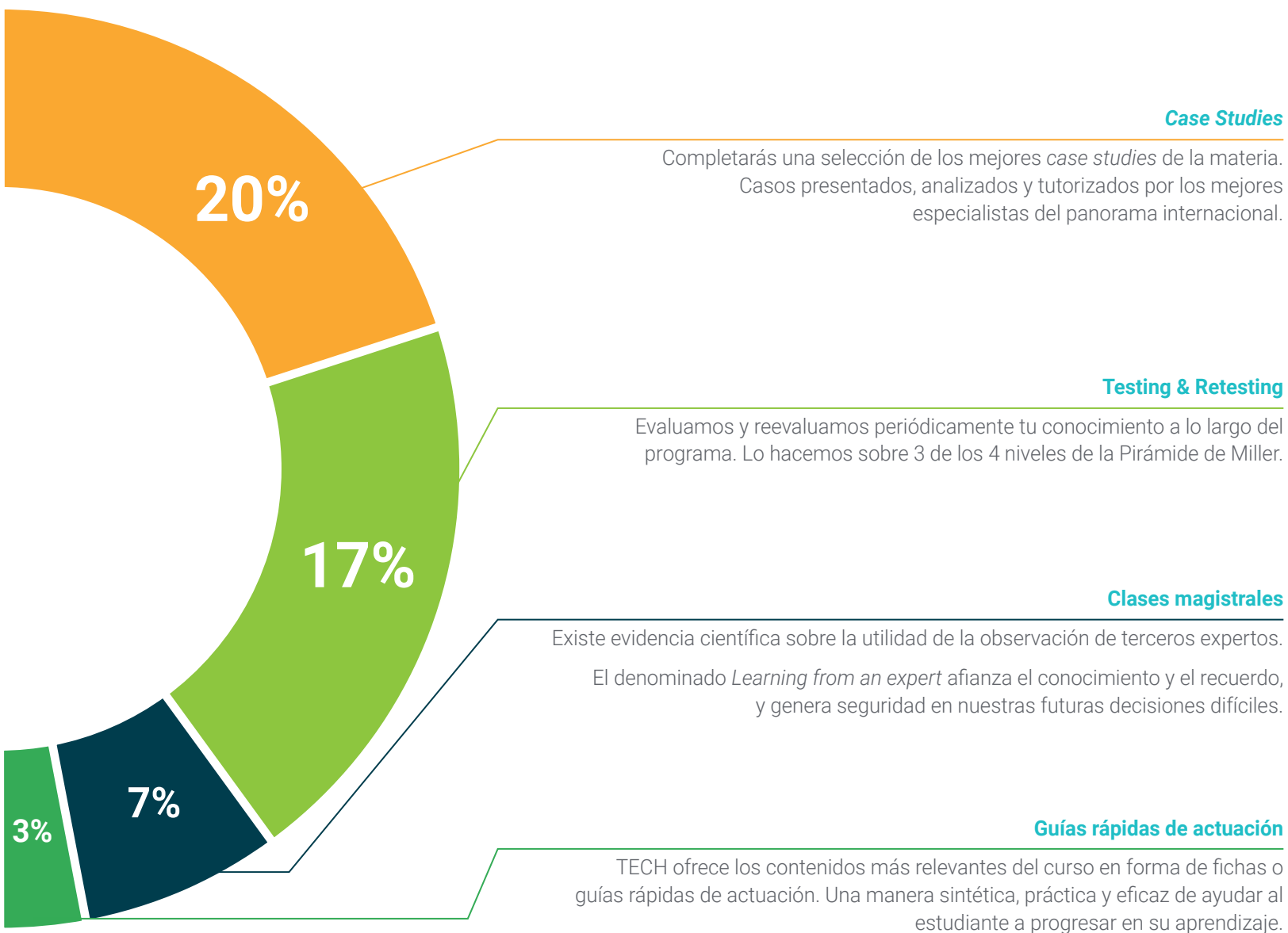
Este sistema exclusivo educativo para la presentación de contenidos multimedia fue premiado por Microsoft como "Caso de éxito en Europa".



Lecturas complementarias

Artículos recientes, documentos de consenso, guías internacionales... En nuestra biblioteca virtual tendrás acceso a todo lo que necesitas para completar tu capacitación.





09

Titulación

El Máster Título Propio en Programación de Videojuegos garantiza, además de la capacitación más rigurosa y actualizada, el acceso a un título de Máster Propio expedido por TECH Universidad.



“

Supera con éxito este programa y recibe tu titulación universitaria sin desplazamientos ni farragosos trámites”

Este **Máster Título Propio en Programación de Videojuegos** contiene el programa universitario más completo y actualizado del mercado.

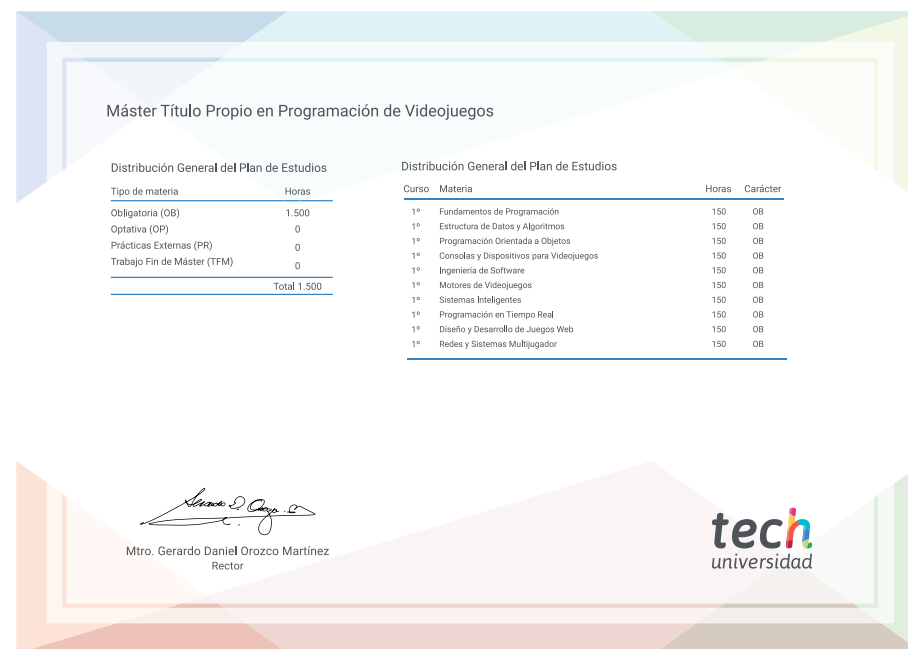
Tras la superación de la evaluación, el alumno recibirá por correo postal* con acuse de recibo su correspondiente título de **Máster Propio** emitido por **TECH Universidad**.

Este título expedido por **TECH Universidad** expresará la calificación que haya obtenido en el Máster Título Propio, y reunirá los requisitos comúnmente exigidos por las bolsas de trabajo, oposiciones y comités evaluadores de carreras profesionales.

Título: **Máster Título Propio en Programación de Videojuegos**

Modalidad: **No escolarizada (100% en línea)**

Duración: **12 meses**





Máster Título Propio Programación de Videojuegos

- » Modalidad: online
- » Duración: 12 meses
- » Titulación: TECH Universidad
- » Acreditación: 60 ECTS
- » Horario: a tu ritmo
- » Exámenes: online

Máster Título Propio

Programación de Videojuegos