

PROGRAMACIÓN MULTIMEDIA Y DISPOSITIVOS MÓVILES

TÉCNICO EN DESARROLLO DE APLICACIONES MULTIPLATAFORMA

Introducción a los videojuegos

Motor de videojuego

Los videojuegos. Concepto

Los videojuegos son mundialmente conocidos, pero ¿cómo podríamos definir qué es un videojuego? Podemos decir que un videojuego es un **mecanismo electrónico capaz de interactuar con uno o varios jugadores** para poder conseguir una serie de **objetivos, o misiones**, proporcionando una respuesta a cada acción del jugador.

Los videojuegos se pueden **clasificar mediante plataformas**, que son en sí el sistema donde se ejecutan los mismos.

Pueden ser:

- PC.
- PlayStation 1/2/3/4/5.
- Xbox.
- Dispositivos Móviles.
- Consolas de Nintendo (Switch, GameBoy, etc.).
- Máquinas recreativas.

Los videojuegos pueden ser:

- Desarrollados para **varias plataformas**, como, por ejemplo, Dark Souls Remastered, que está disponible para Nintendo Switch, PlayStation 4, Xbox One y PC.
- **Exclusivos**, es decir, que solo están **disponibles en una plataforma**, por ejemplo, Bloodborne, que únicamente está disponible para PlayStation 4.

La **arquitectura global** de un videojuego es una construcción que la componen distintos **bloques funcionales**, de los que destacamos:

- **Historia**: Es la manera en que se desenvolverán los personajes del juego y el hilo conductor del mundo que se representará.
- **Arte conceptual**: Mostrará el **aspecto general** del juego.
- **Sonido**: Voces, sonidos ambientales, efectos sonoros y música.
- **Mecánica de juego**: Donde se especificará el **funcionamiento general** del juego.
- **Diseño de programación**: Mediante este elemento se describirá la manera en que el videojuego será **implementado** en una máquina (PC, consola, móviles, etc.) mediante un determinado **lenguaje** de programación y siguiendo una **metodología** concreta.

Historia de los videojuegos

Los primeros videojuegos se remontan sobre el año **1950**, al poco tiempo de la creación de las primeras computadoras electrónicas. Todos los videojuegos que se realizaron en esta época eran muy rudimentarios, incluso algunos no podrían ser catalogados como videojuegos en sí.

Sobre el año **1970** apareció una compañía que lo cambiaría todo, esta fue **Atari**, que con su videojuego **Pong**. Crearon lo que hoy se conoce como **máquinas recreativas**, iniciando una fortísima revolución en el mundo de los videojuegos.

A partir de este momento la industria del videojuego fue en aumento, llegando en los **años 80** las **videoconsolas con 16 bits**.

Otros hitos de este sector fueron la creación de la **primera PlayStation** por parte de Sony en el año **1994**, la salida al mercado de la **GameBoy de Nintendo** en **1986**, la aparición de la **XBOX** de Microsoft en el año **2002**, etc.

Clasificación de los videojuegos

Dentro del mundo de los videojuegos, existe una gran cantidad de clasificaciones posibles. Además de la clasificación por plataformas comentada anteriormente, podemos destacar:

- Videojuegos de **acción**: Este tipo de videojuegos son aquellos en los que habrá que realizar unos movimientos y combates rápidos.
- Videojuegos de **aventura**: Son aquellos en los que hay una **historia compleja**, que deberá ir desarrollando poco a poco el jugador. Algo muy típico en este tipo de videojuegos es la inclusión de algún tipo de puzzle para hacer que el jugador tenga que pensar cómo resolverlo.
- Videojuegos de **lucha**: Consisten en pelear o luchar contra otros personajes del videojuego, (los llamados **NPC, Non-Player Character**. Personajes No Jugables en español) o contra otros jugadores en un modo multijugador.
- Videojuegos de **plataformas**: Consisten en que el jugador deberá ir **moviéndose y saltando sobre una serie de terrenos** o plataformas para avanzar en el videojuego. Quizás, el más famoso de esta categoría sea Super Mario Bros.
- Videojuegos de **estrategia**: Estos requerirán al jugador una **planificación** bastante elevada de sus movimientos, ya que de ello dependerá su victoria.
- Videojuegos **puzzles**: Son los que ofrecen una serie de puzzles al jugador donde este deberá averiguar cómo poder resolverlos para ir avanzando en el mismo.
- Simuladores: Son los que ofrecen una simulación de tareas del mundo real. Pueden ser simuladores de vuelo, de negocios, etc.

En el mundo de los videojuegos, es habitual que haya modificaciones de los mismos mediante **actualizaciones o parches** que proporcionará el desarrollador, ya sea para introducir alguna **mejora** o por adaptarse **mejor al tipo de mercado** que lo demanda. Incluso, es muy común que se realicen

remakes completos de un videojuego antiguo e incluir en el mismo nuevas zonas de mapeado, personajes, misiones, etc. Este es el caso de Demon Souls en su remake para PlayStation5.

Una ventaja de desarrollar videojuegos para videoconsolas frente a PC es que su hardware no se puede ampliar.

¿Por qué el hecho de no poder ampliar el hardware constituye una ventaja?

A pesar de que pueda parecer una ventaja el hecho de poder ampliar el hardware realmente no lo es, sobre todo para los desarrolladores, ya que si el hardware tiene un tamaño fijo, los desarrolladores ya conocen de antemano exactamente las especificaciones del hardware para el cual están creando el juego. Entonces no hay variabilidad en términos de tarjetas gráficas, procesadores o memoria RAM como puede haber en PC. Esto es una ventaja ya que permite una mayor estabilidad y consistencia en el rendimiento del juego.

El por qué de los motores de videojuegos

Los videojuegos se componen de multitud de bloques, y no todos son sencillos de implementar, por ejemplo, ¿cómo podemos hacer que salte un personaje?, ¿y si queremos que uno de los enemigos del videojuego persiga a nuestro personaje?

Todas estas partes se pueden hacer desde cero, pero no es nada recomendable, ya que si investigamos un poco, podremos observar que existe un componente llamado **motor de un videojuego**, que ya nos **implementa las bases** de la gran mayoría de **elementos** que vamos a necesitar realizar en nuestros videojuegos. La implementación de cero de un motor completo puede ser una tarea colosal, así que siempre optaremos por **usar un motor que ya esté implementado** y disponible.

Para conocer algunos de los entornos de desarrollo básicos que te permitirán iniciarte en la creación de videojuegos sin que intervenga demasiado la programación, puedes consultar el siguiente [enlace](#).

Motores de videojuegos

Los videojuegos de hoy en día necesitan una gran cantidad de recursos para poder ejecutarse, basta con ver los requerimientos mínimos que necesita cualquier videojuego de PC o las características de las videoconsolas actuales.

En los videojuegos, podemos distinguir **tres grandes partes**:

1. El código: Este será todo el código que se necesitará para crear el juego en sí. En esta parte será donde se **implementará toda la lógica** del juego.
2. El motor del juego.
3. Los recursos requeridos: Estos son los recursos necesarios por el juego, como pueden ser banda sonora, sonidos de efectos, imágenes, etc.

Quizás, la **parte más importante** de todo videojuego sea su **motor**, ya que este **va a contener la funcionalidad** sobre la que se va a apoyar directamente el videojuego.

Los motores de los videojuegos surgieron como tales en los **años 90** y se empezaron a utilizar en los **videojuegos 3D**, ya que estos necesitaban una potencia de cálculo muy elevada. **También** podemos usar motores en los **videojuegos 2D**, aunque estos serán mucho más simples que los utilizados en los 3D.

Los motores nos van a proporcionar una **abstracción sobre el hardware** del dispositivo, lo cual va a permitir a los desarrolladores del videojuego poder **programar sin tener la necesidad de conocer la arquitectura del hardware** sobre el que se ejecutará el videojuego. Debido a esto, la gran mayoría de los motores existentes se han **desarrollado a partir de una API ya implementada**, como puede ser OpenGL, DirectX, SDL, etc.

Otra gran ventaja de esta abstracción del hardware es que puede **permitir** el desarrollo de **aplicaciones multiplataforma**.

Es importante mencionar que con los **primeros videojuegos** se utilizaba una **aceleración de gráficos 3D basada en hardware**, lo cual implicaba que el **hardware** de la máquina donde se ejecutaría el videojuego **debía ser bastante potente** para soportar dicha aceleración.

Clasificaciones de los motores de videojuegos

<https://bit.ly/36RLnYe>

Según el nivel de abstracción

(Tipo de motor : Descripción)

Motores para mostrar en pantalla:

Este tipo de motores son **muy simples**, y están diseñados únicamente para mostrar elementos por pantalla. Como ejemplos tenemos: Biblioteca **SDL**.

Motores con soporte de lógica:

Este tipo motores agregan a lo anterior un **soporte total o parcial de la lógica** del videojuego. Como ejemplos tenemos: **Ogre3D**.

Motores que incluyen plataformas para creación de contenido:

Este tipo de motores agregan a lo anterior una serie de programas que permitirán **crear contenido** para los videojuegos. Como ejemplos tenemos: **Blender**.

Según la licencia de uso

(Tipo de motor : Descripción)

De código abierto:

Para poder usar este tipo de motores no hace falta pagar, sólo hay que prestar atención al tipo de **licencia** de código abierto que esta usando y **respetarla**. Como ejemplos tenemos: **Ogre3D**.

Propietarios:

Estos motores son creados por grandes empresas y su uso estará limitado mediante una **licencia de uso**, la cual costara dinero. Como ejemplos tenemos: **UnrealEngine**.

Explicado:

Clasificaciones de Motores de Videojuegos:

Vamos a ver las diferentes clasificaciones de los motores de videojuegos. Podemos hacer dos principales distinciones.

1. Nivel de Abstracción:

- Motores para mostrar elementos por pantalla: Este tipo de motores son muy simples y se encargan únicamente de mostrar cosas por pantalla, como texto, imágenes o polígonos. Un ejemplo común es la biblioteca SDL, utilizada en el desarrollo del videojuego Age of Empires 2.
- Motores con soporte de lógica: Agregan a la funcionalidad de mostrar elementos por pantalla un soporte total o parcial de la lógica del videojuego. Un ejemplo de este tipo de motor es Ogre 3D.
- Motores que incluyen plataformas para creación de contenido: Estos motores, además de contar con soporte de lógica, ofrecen la posibilidad de crear contenido, como modelado de personajes, enemigos o escenarios. Blender es un ejemplo que permite desarrollar videojuegos y diseñar modelos 3D de personajes.

2. Licencia de Uso:

- Motores de código abierto: Permiten descargar su código y su uso es gratuito.
- Motores propietarios: Desarrollados por empresas propietarias que no proporcionan su código fuente y suelen requerir el pago de licencias.

En resumen, podemos encontrar cualquiera de estos tres tipos de motores con cualquiera de estas dos licencias de uso.

Ventajas de los motores de los videojuegos

Ya sabemos que los motores en los videojuegos van a soportar gran parte del peso de la lógica de los mismos, y es por eso que su uso trae consigo múltiples ventajas.

En la siguiente tabla, podemos ver **algunas de las más notables ventajas**, según diferentes situaciones que se pueden presentar:

Comparación con y sin motor

Migración del videojuego a otra plataforma distinta:

- **Sin usar motor:** No queda más remedio que **revisar el código completo** del videojuego y ver qué partes deberemos cambiar para poder migrarlo a la plataforma deseada. Puede ocurrir que haya recursos que ya no sean válidos en la nueva plataforma. Si llegamos a detectar **errores** en el código del videojuego, habrá que **cambiarlos en todas las plataformas** que tengamos.
- **Usando motor:** Si el motor que estamos usando es soportado por la nueva plataforma, lo único que tendremos que hacer será **recompilar de nuevo el proyecto** del videojuego.

Llegada de nuevos programadores al proyecto:

- **Sin usar motor:** Habrá que **explicarles** a los nuevos programadores toda la **estructura del código** del proyecto.
- **Usando motor:** Lo único que habrá que hacer será **formar** a estos nuevos programadores en el **uso del motor** utilizado.

Incorporación de nuevos efectos a nuestro videojuego:

- **Sin usar motor:** Habrá que invertir tiempo en **investigar cómo funciona** la característica nueva que queremos **incorporar, implementarla y realizar todas las pruebas** pertinentes para asegurar su funcionamiento. Además, habrá que hacerlo en todas las plataformas de las que dispongamos.
- **Usando motor:** Seguramente, el **motor** que estemos usando **implemente la característica** que queremos agregar.

Realizar una segunda parte de un videojuego:

- **Sin usar motor:** Tendremos que **volver a revisar todo el código** del mismo para ver qué elementos podremos reutilizar en la siguiente parte y cuáles no.
- **Usando motor:** Aquí no habrá ningún tipo de problema, ya que el **código está separado del motor**.

Como observamos, las ventajas de usar un motor en el desarrollo de videojuegos pueden llegar a ser muy importantes.

Estas son solo algunas de ellas, por lo que seguro que en el día a día de un proyecto se presentarán más situaciones que podremos solventar fácilmente si se están usando los motores.

Sin embargo, habrá que indicarles ciertos comportamientos para que puedan procesar los modelos.

Motores 2D

Los videojuegos en 2D son aquellos que solamente utilizan **2 dimensiones**, alto y ancho, dejando sin utilizar la profundidad. Este tipo de juegos utilizan **formas planas** que se desplazan por la pantalla, hacia los lados, y hacia arriba y abajo.

Los motores 2D nos van a permitir **dibujar de una forma más sencilla las figuras geométricas**, como pueden ser círculos, cuadrados, rectángulos, triángulos, etc., sobre las que estarán basados todos los componentes de cada acción.

Este tipo de videojuegos tienen **varias capas** en las que en cada una se irá representando **una parte del juego**, como pueden ser los personajes, los enemigos, las monedas, etc.

Los llamados **sprites** (significa duendecillo en inglés) serán las **representaciones de los elementos** del juego (personajes, enemigos...) en **cada una de las capas**. Estos sprites **van a reproducir todos los movimientos** que puedan realizar los personajes o, en definitiva, cualquier elementos del videojuego.

El encargado de dar el **soporte** necesario **a los sprites** será el **motor 2D**, el cual tendrá herramientas que nos van a permitir **escalarlos, rotarlos, “recortarlos”** para obtener un fragmento donde estará el elemento que queremos mostrar en pantalla, etc.

Los sprites se irán superponiendo respetando las capas del videojuego, por ejemplo, primero se dibujará el fondo, luego se dibujará al personaje con el que jugaremos, posteriormente los enemigos y las partes del mapeado donde podremos interactuar, y por último, se dibujará la información del mismo, como pueden ser la cantidad de vidas que tenemos, el tiempo que llevamos jugando, nuestra puntuación, etc.

Ejemplo de videojuego en 2D: Plants vs Zombies.

Fuente: <https://www.pngocean.com/gratis-png-clipart-xihxz/descargar>

Concepto de Colisión en Videojuegos en 2D:

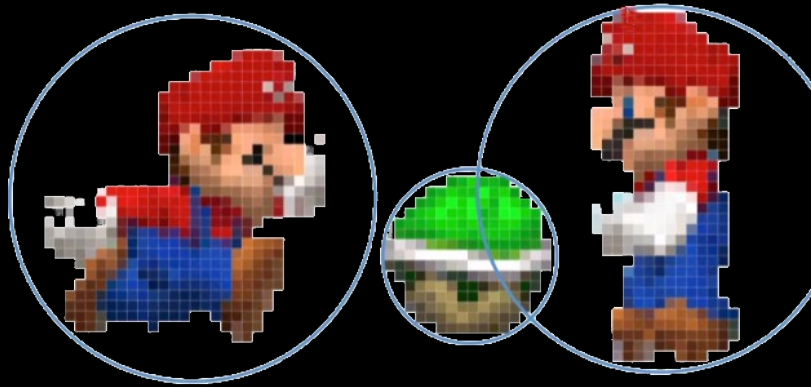
Vamos a ver qué es el concepto de colisión. Como ya sabemos, los videojuegos en dos dimensiones utilizan sprites para representar los personajes. Cuando utilizamos diferentes personajes, tendremos varios sprites moviéndose por el escenario entre comillas.

Puede ocurrir que en cierto momento haya dos sprites que se toquen o se crucen dentro del escenario, y a eso es a lo que llamamos una **colisión**. Las colisiones pueden darse entre personajes de jugadores, personajes de jugadores y personajes enemigos, o incluso con elementos del entorno de nuestro videojuego.

Lo más común para determinar si un enemigo ha matado a un personaje en un videojuego en 2D es calcular si ha habido una colisión entre ellos. En tal caso, podremos hacer que el jugador pierda la partida o ir quitándole vida al jugador.

Detección de Colisiones:

Para la detección de colisiones, hay varios métodos. Como podemos ver en esta imagen, tenemos tres sprites: uno, dos y tres. Se ha colocado una circunferencia sobre cada sprite, que es el método más preciso para la detección de una colisión. La circunferencia representa el espacio que ocupa nuestro personaje dentro del espacio de nuestro videojuego.



También es posible realizar la detección mediante cuadrados, aunque este método no es tan preciso como el de la circunferencia. Trozos como el mostrado aquí, por ejemplo, podrían provocar una colisión cuando en realidad no la hay.

Para determinar si ha habido una colisión, debemos observar dónde están los espacios ocupados por nuestros sprites. Si alguno de ellos se cruza, como en este caso, tendremos una colisión entre esos dos sprites.

La detección de colisiones es responsabilidad del motor 2D, motor 5D, motor 6D, motor 7D y motor 8D.

¿Cuál es el primer paso para comenzar con el desarrollo de mi videojuego?

Es común sentirse perdido al iniciar en el mundo del desarrollo de videojuegos, incluso cuando se trata de proyectos simples. La pregunta inicial podría ser: ¿deberíamos comenzar a programar y ajustar según los resultados, o sería más adecuado elaborar un pequeño guión primero? A menudo, entendemos que el proceso de crear un videojuego, como los que vemos en el mercado actual, implica diversas etapas:

- La **primera**, aunque obvia, es concebir una **buena idea** para el juego, algo atractivo y cautivador para los jugadores.
- Después de tener la idea, es crucial no lanzarse a programar sin un plan claro. Es necesario realizar un estudio exhaustivo sobre el tema central del videojuego y plasmarlo en un guión que defina los pasos a seguir en el desarrollo. Identificar los componentes clave, establecer un orden para su desarrollo, reconocer las herramientas de apoyo disponibles, decidir la plataforma objetivo (ya que no es lo mismo desarrollar para PC, PlayStation o dispositivos móviles), y esbozar los personajes y otros elementos son algunos de los muchos aspectos que debemos considerar.

Finalmente, una vez que todo esté bien definido, se inicia la fase de programación.

Motores 3D

Al igual que existen los videojuegos en 2D, que utilizan solo dos dimensiones, existen los videojuegos en 3D, los cuales utilizan las tres dimensiones: alto, ancho y profundidad. En estos, podemos movernos en libertad al poder utilizar los tres espacios.

Este tipo de videojuegos utiliza una operación llamada **renderizado**, que consiste en el procesamiento de imágenes en 3D para poder **transformarlas a imágenes 2D**, y así poder **adaptarlas al ojo humano**.

Por norma general, los **objetos 3D** que se usan en los videojuegos están **formados mediante vértices**, formando una serie de **polígonos** que darán lugar al objeto 3D. A mayor cantidad de vértices, tendremos una **mayor cantidad de polígonos**, haciendo que la imagen o modelo 3D sea mucho más realista, pero con el inconveniente de que será **muchísimo más costosa** de procesar.

Una vez que tenemos creados los polígonos, podremos **aplicarles texturas**, las cuales harán que el modelo 3D tenga una imagen mucho más realista.

Los **motores 3D** serán los **encargados de todo esto**, simplificando de una manera bastante considerable todo el proceso de desarrollo del videojuego.

Básicamente, el trabajo del motor 3D será el de:

- crear los modelos,
- moverlos,
- rotarlos según precisemos,
- escalarlos para hacerlos más grandes o más pequeños,
- y un largo etcétera.

Una vez que tenemos todo esto, le podremos **agregar focos de luces** y una **cámara**, lo que dará lugar a una **escena**, que es el **propio videojuego en 3D**.

Este tipo de videojuego **no va a utilizar sprites**, como ocurría en los videojuegos sino que necesitaremos de un programa especial de modelado en 3D para poder crear los personajes que utilizaremos en estos videojuegos. Un ejemplo de programa de modelado en 3D es Blender.

Motores de físicas, IA y sonido

Los movimientos de los personajes en los videojuegos deben ser lo más reales posibles, puede haber caídas, choques, saltos, peleas, etc. Todo esto tiene detrás una gran cantidad de **cálculos matemáticos y físicos**, siendo el **motor de físicas**, que está integrado en los motores (ya sean 2D o 3D), el encargado de realizar todos esos cálculos.

Existen otros motores, como el de inteligencia artificial, que es el encargado de llevar la inteligencia artificial de todos los **personajes** del juego, y el de **sonido**, que será el encargado de **gestionar todos los sonidos del juego**, reproducirlos, pararlos, cambiarlos, etc.

La industria de los videojuegos atrae a numerosas personas no solo por la experiencia de jugar y disfrutar los juegos, sino también por la oportunidad de participar en su desarrollo. Aunque es evidente que contar con una sólida base en programación es esencial para ingresar a este mundo, esto por sí solo no es suficiente, ya que la creación de un videojuego puede evolucionar en un proyecto sumamente ambicioso.

Los equipos de desarrollo de videojuegos se conforman con profesionales especializados en diversas **áreas**, entre ellas:

- Ingeniería de **software**,
- Diseño **gráfico**,
- **Matemáticas** y **física** (dado que los videojuegos involucran una cantidad significativa de cálculos matemáticos y físicos en su desarrollo).
- Además, es crucial contar con un equipo de **marketing** que se encargue del lanzamiento, la comercialización y las ventas del juego.
- Por supuesto, no se puede pasar por alto la importancia de tener equipos de **programadores** especializados en varios lenguajes de programación.