

«Talento Tech»

Desarrollo de Videojuegos

Unity 3D

Clase 04



Clase N° 4 | Animaciones en 3D

Temario:

- Introducción a animaciones en 3D.
- Configuración de Animator.
- Uso de animaciones importadas.

Objetivos de la clase

En esta clase vas a dar el siguiente paso para que tu proyecto empiece a sentirse como un verdadero videojuego: **darle vida al personaje y al entorno a través de la animación**. Hasta ahora, trabajaste con movimiento, plataformas, físicas y mecánicas. Pero todo eso, sin animaciones, se siente rígido y poco natural. Las animaciones permiten transmitir **intención, peso, emoción y dinamismo**.

Este conocimiento no solo te permitirá mejorar la estética de tu juego, sino también **integrar acciones visuales con el comportamiento del personaje y las interacciones del entorno**, haciendo que todo lo que pase en pantalla tenga una respuesta visual coherente y atractiva.

Introducción a la animación en 3D.

Una **animación en 3D** es el proceso de darle vida a un objeto o personaje dentro de un entorno tridimensional, mediante la **modificación progresiva de sus propiedades a lo largo del tiempo**. Esto permite que parezca moverse, reaccionar o interactuar de forma orgánica dentro de un juego o simulación.

En Unity, la animación se aplica tanto a personajes como a objetos del entorno (puertas, luces, cámaras) y es clave para generar una experiencia más fluida y atractiva.

Características principales de una animación en 3D:

1. Keyframes:

- Los keyframes son puntos clave en la línea de tiempo que definen estados específicos de un objeto (posición, rotación, escala u otras propiedades).
- Entre estos puntos, el software interpola automáticamente para generar un movimiento fluido.

2. Movimiento en un espacio tridimensional:

- Los objetos pueden moverse, rotar o escalar a lo largo de los tres ejes: X, Y y Z.
- Esto permite crear animaciones complejas como caminar hacia la cámara, saltar o rotar sobre el eje Y.

3. Aplicaciones:

- **Personajes:** Movimientos corporales, expresiones faciales, gestos.
- **Objetos:** Rotación de ruedas, apertura de puertas, explosiones, etc.
- **Cámaras:** Animación de cámaras para crear cinemáticas.

¿De dónde provienen las animaciones?

En Unity, las animaciones pueden generarse de diferentes formas:

- **Animaciones manuales:** Diseñadas cuadro por cuadro en herramientas como Maya, Blender o directamente en Unity.
- **Motion capture (Mocap):** Datos capturados de movimientos reales, aplicados a un modelo 3D.
- **Bibliotecas externas:** Ejemplo: Mixamo ofrece animaciones predefinidas para personajes.

Unity utiliza el componente **Animator** para gestionar cómo se combinan y controlan las animaciones en un personaje u objeto. A través del **Animator Controller**, podemos establecer reglas y transiciones para que el sistema sepa **cuándo y cómo reproducir cada animación**, ya sea en respuesta a un input del jugador, una colisión o una condición lógica del juego.

Animation Clips.

En Unity, una animación se gestiona a través de lo que se conoce como un **Animation Clip**. Este clip es básicamente un contenedor que guarda información sobre cómo cambian ciertas propiedades de un objeto (como posición, rotación o escala) a lo largo del tiempo.

Con estos clips, podemos crear animaciones directamente desde Unity, sin necesidad de usar software externo.

Crear una animación propia en Unity

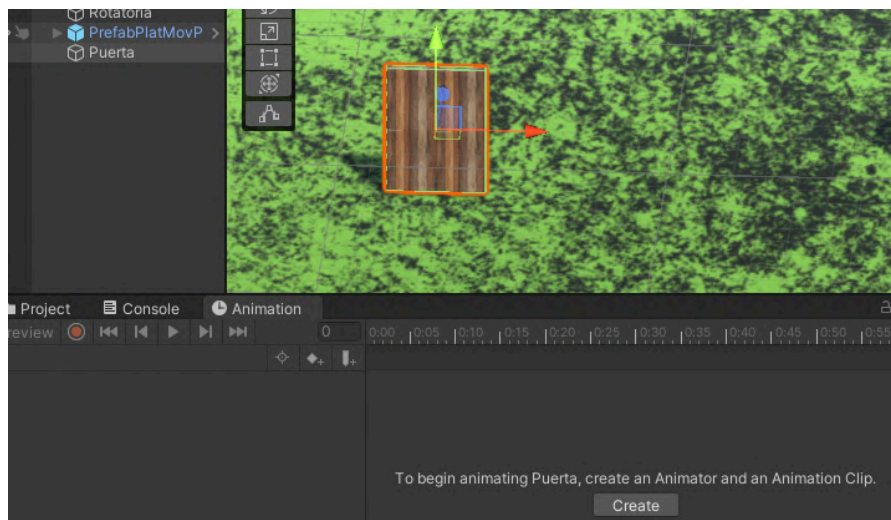
1. Seleccionar un objeto

Podés empezar con cualquier objeto sencillo de la escena (por ejemplo, un cubo o una esfera).

2. Abrir la ventana Animation

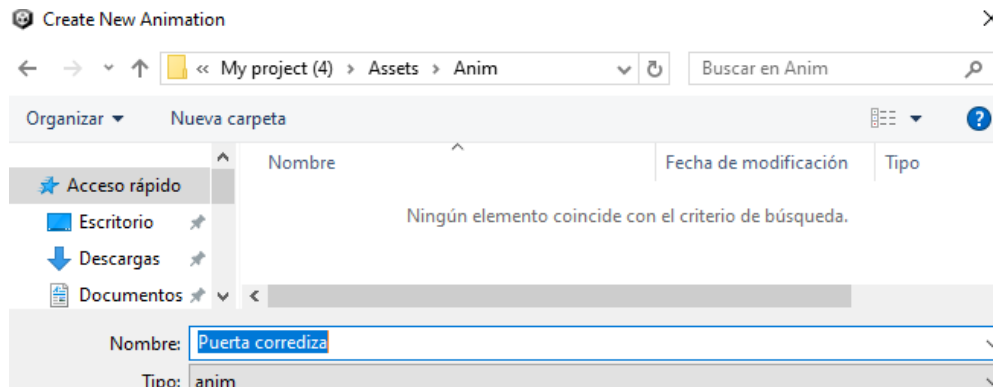
Menú superior: **Window** → **Animation** → **Animation**

Esta ventana es donde vas a construir la animación, similar a una línea de tiempo en un editor de video.



3. Crear un nuevo Animation Clip

Hacés clic en el botón **"Create"**. Esto genera un nuevo archivo de animación asociado al objeto seleccionado.

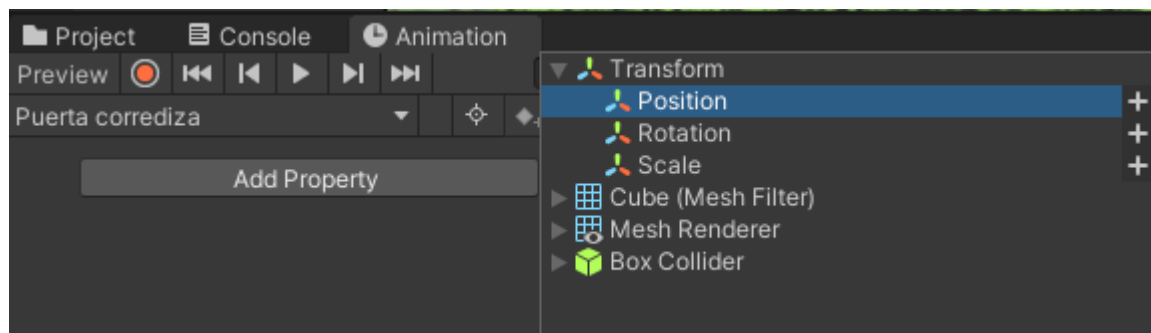


Agregar propiedades a animar

Una vez creado el clip, aparece el botón **"Add Property"**, que te permite elegir **qué parte del objeto querés animar**.

Por ejemplo:

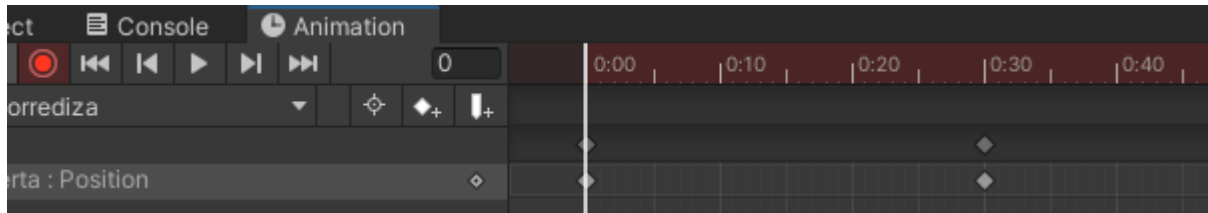
- **Transform** → **Position**: para animar el movimiento.
- **Transform** → **Rotation**: para que gire.
- **Mesh Renderer** → **Material**: para cambiar colores o texturas.



En nuestro caso elegiremos **"Position"** del **"Transform"**.

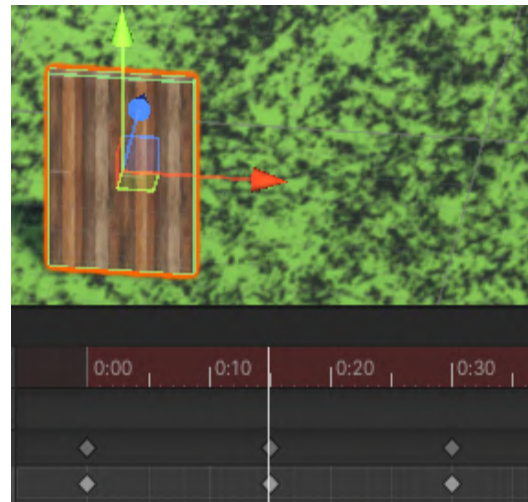
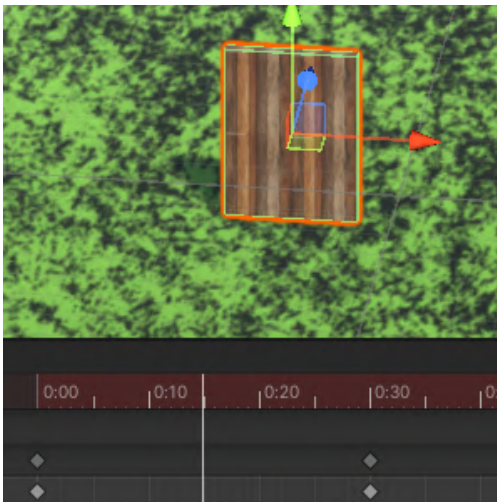
Grabar la animación con keyframes

Al presionar el botón rojo (junto a *Preview*), Unity entra en modo grabación. La línea de tiempo se tiñe de rojo y todo cambio que hagas **se registrará como un keyframe**.



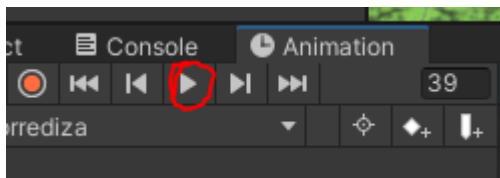
¿Cómo se hace?

1. Mueves la **barra blanca** (indicador de tiempo) a un segundo determinado.
2. Modificas la propiedad del objeto (por ejemplo, lo desplazas en el eje X).
3. Unity guarda automáticamente un keyframe en ese punto.



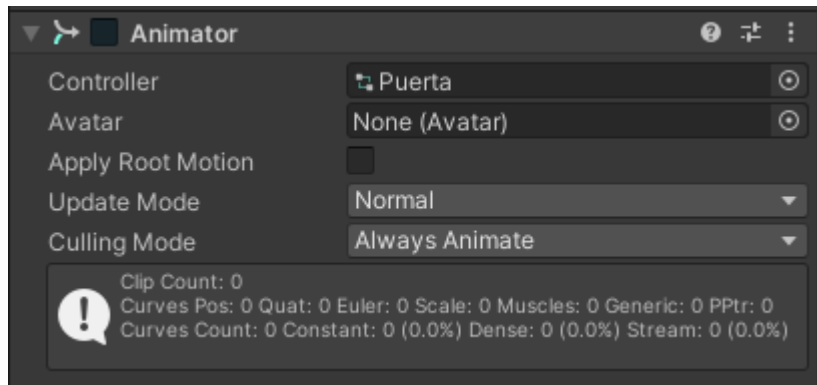
Reproducir la animación

Cuando finalices la secuencia de movimientos, hacé clic en **Play** en la ventana Animation para visualizar el resultado. Si todo está bien, verás cómo el objeto se mueve de forma automática según los valores que marcaste.



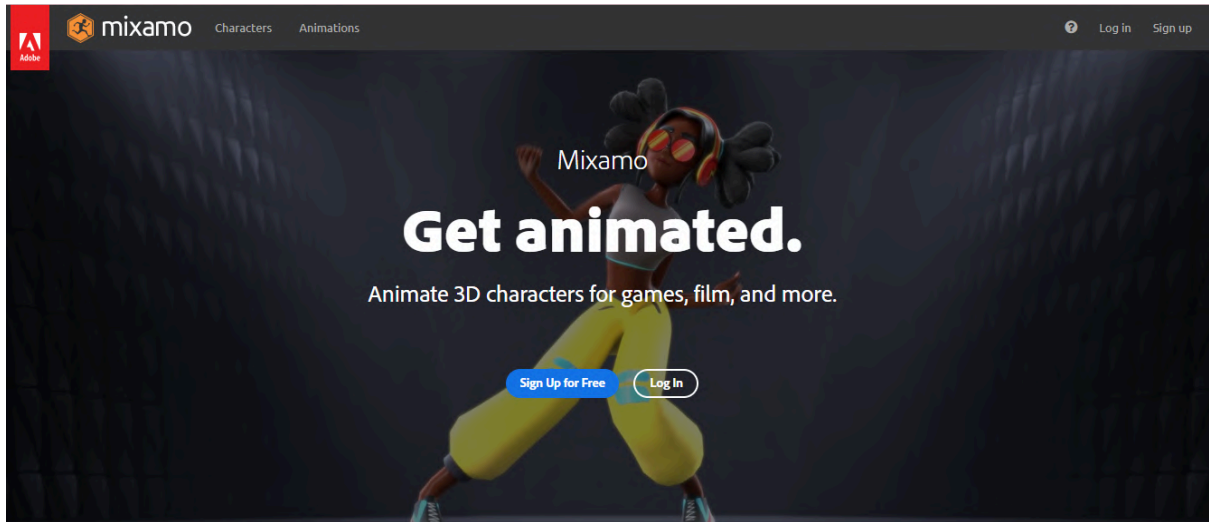
¿Qué pasa con el Animator?

Al crear una animación, Unity añade automáticamente un componente llamado **Animator** al objeto, y genera un **Animator Controller**. Este controlador es el que se usará para organizar y conectar múltiples animaciones entre sí (como caminar, saltar, atacar), algo que se explorará más adelante en esta clase.



Importar personajes y animaciones.

Además de crear animaciones manualmente en Unity, también podemos importar personajes animados desde bibliotecas externas. En esta clase vamos a trabajar con [Mixamo](https://mixamo.com), una plataforma gratuita de Adobe que ofrece cientos de personajes y animaciones listos para usar en motores como Unity.

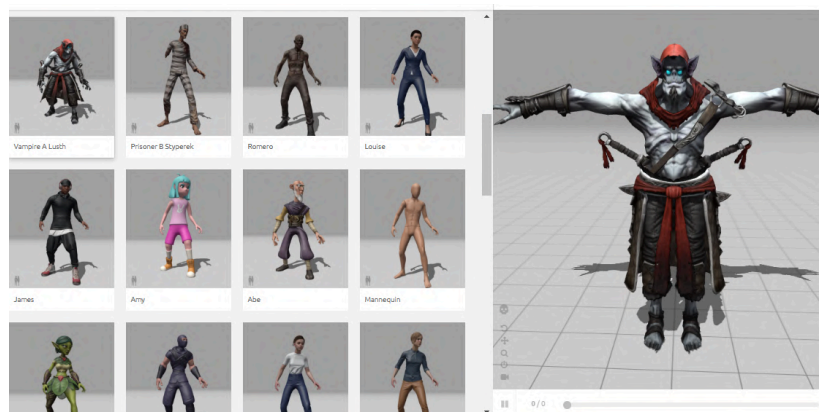


¿Qué vas a lograr?

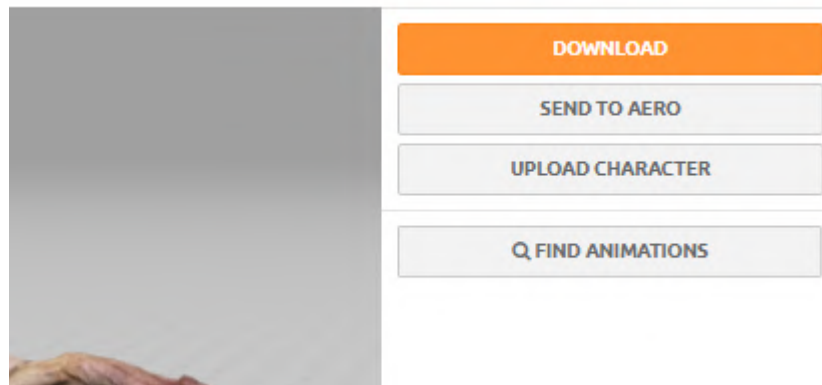
- Descargar un personaje 3D con esqueleto (rig) incluido.
- Descargar animaciones compatibles con ese personaje.
- Importarlos en Unity y aplicar las animaciones correctamente.
- Configurar texturas y materiales para que el modelo se vea bien en escena.

Paso a paso: Descargar personaje en Mixamo

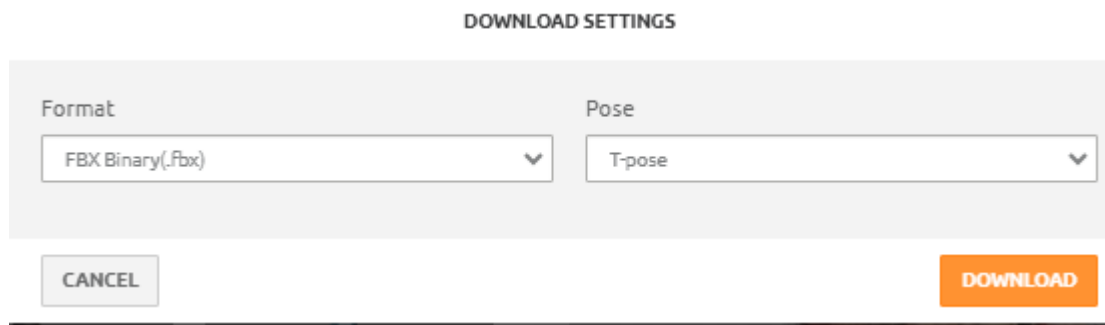
1. Ingresá a mixamo.com y logueate con una cuenta de Adobe (puede ser gratuita).
2. En el menú superior izquierdo, hacé clic en **Characters**.
3. Elegí un personaje del catálogo.



4. Presioná el botón **Download** (ubicado a la derecha).



5. En la ventana emergente, dejá las opciones por defecto:
- Formato: **FBX for Unity**
 - Pose: **T-pose**



6. Hacé clic en **Download**.

Esto descargará un archivo **.fbx**, que podés **arrastrar directamente a la carpeta Assets de tu proyecto Unity**.

¿Por qué no se ve bien el personaje al importarlo?

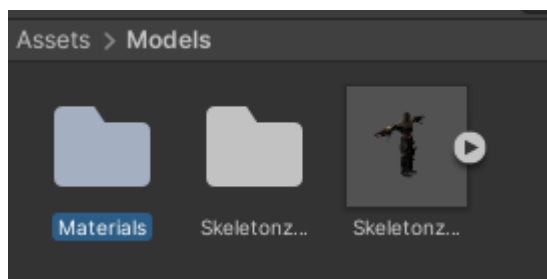
Cuando colocás el modelo en la escena, es posible que **no tenga texturas visibles**. Esto se debe a que Unity, por defecto, no aplica automáticamente los materiales externos



Restaurar materiales

1. Seleccioná el modelo en el panel **Project**.
2. En el **Inspector**, hacé clic en la pestaña **Materials**.
3. Buscá la opción **Location** y cambiala a:
Use External Materials (Legacy)
4. Hacé clic en **Apply**.
5. Unity te mostrará un mensaje. Presioná **Fix Now**.

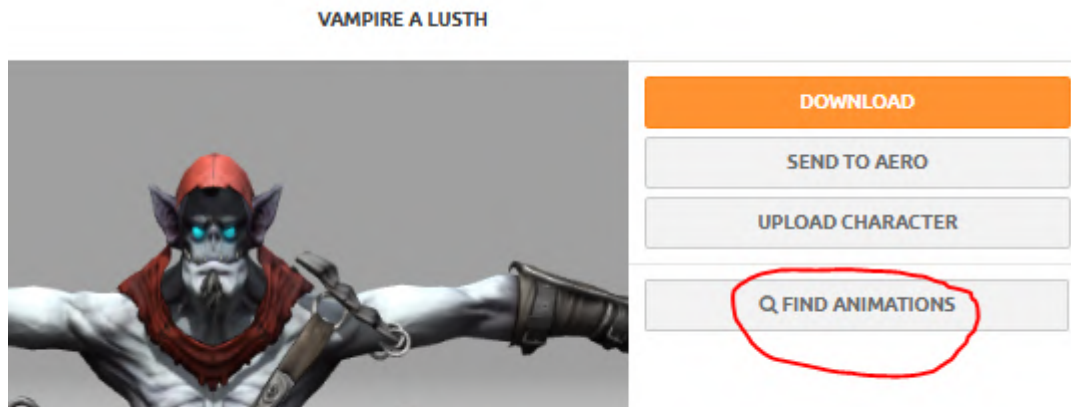
Esto generará automáticamente las carpetas de **Materiales** y **Texturas** asociadas al modelo, y tu personaje se verá correctamente en la escena.



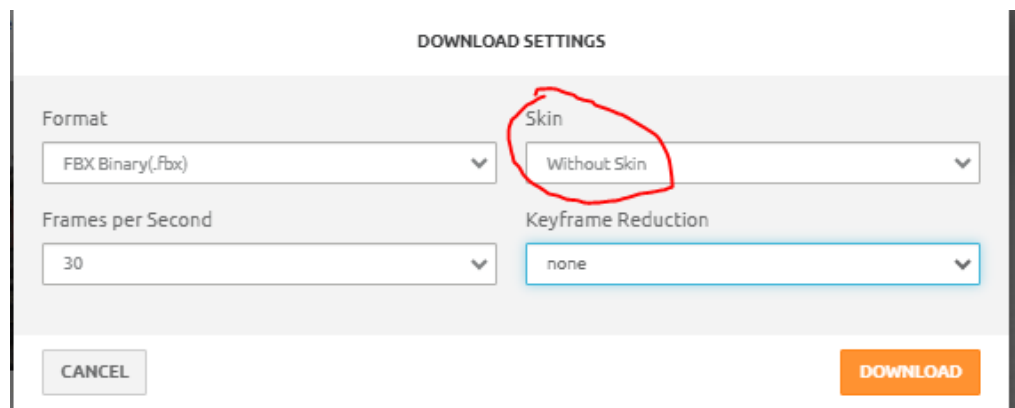
Descargar animaciones desde Mixamo.

Una vez tengas el personaje, volvé a Mixamo para elegir una animación:

1. Presioná el botón **Find Animations** (debajo de "Download").



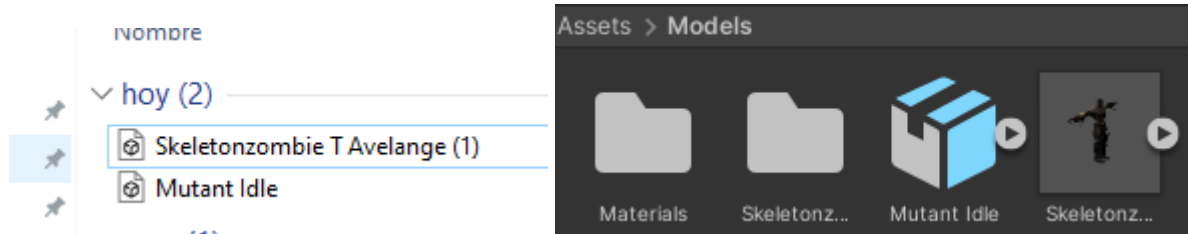
2. Buscá animaciones por categorías:
 - *Idle, Walk, Run, Jump, Attack, Death, Reaction...*
3. Seleccioná una animación que quieras usar.
4. En la opción **Skin**, marcá **"Without Skin"** (así no volvéis a descargar el modelo).



5. Hacé clic en **Download** → **FBX for Unity**.

Importar animaciones a Unity

- Arrastrá el nuevo **.fbx** a la carpeta del proyecto.
- Unity detectará que contiene **solo animación** y no modelo.
- Ahora vas a poder aplicar esa animación al personaje original, configurando correctamente el **rig** y el **Animator**, que veremos en el siguiente bloque.



Configurando para animar.

Una vez que importamos el modelo 3D del personaje y las animaciones desde Mixamo, necesitamos asegurarnos de que ambos compartan una estructura compatible, llamada rig. Además, tenemos que configurar un Animator que pueda controlar esas animaciones en Unity, mediante un **Animator Controller**.

¿Qué es un rig?

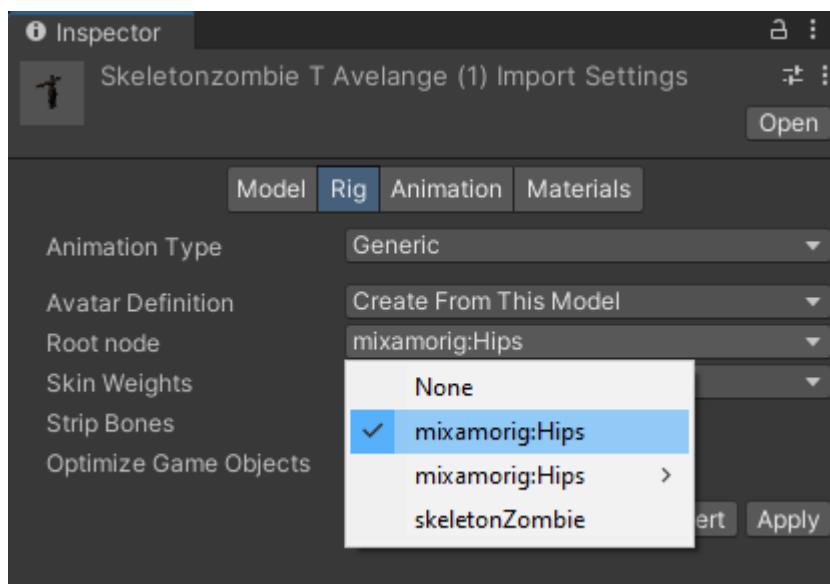
El **rig** es el esqueleto digital del personaje, formado por huesos (bones) que permiten que las animaciones se apliquen al modelo. Si dos elementos (personaje y animación) usan el mismo rig o estructura, pueden compartir animaciones.

En el caso de los modelos descargados desde Mixamo, **el rig ya está incluido**, lo cual simplifica mucho el proceso.

Paso 1: Crear el avatar del personaje

1. Seleccioná el modelo **.fbx** del personaje en el panel **Project**.
2. En el **Inspector**, abrí la pestaña **Rig**.
3. En el campo **Avatar Definition**, seleccioná:
Create From This Model
4. En el campo **Root Node**, elegí:
"Hips" (o el nodo raíz del esqueleto del personaje).
5. Hacé clic en **Apply**.

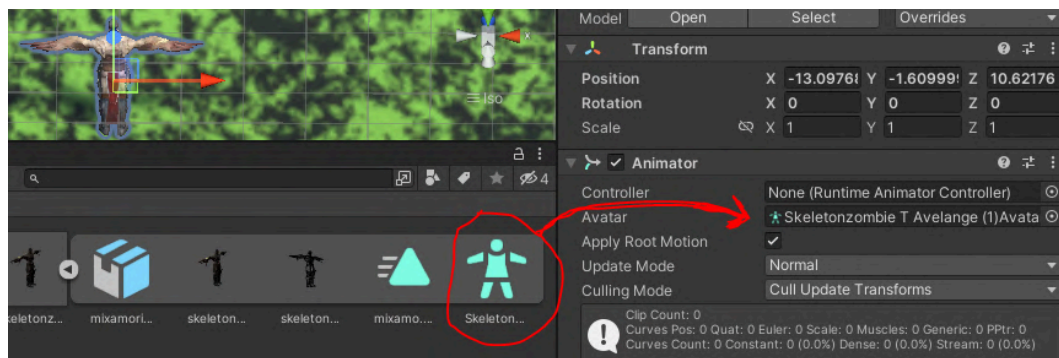
Esto genera un **avatar**, que es una representación del esqueleto, necesaria para que las animaciones puedan aplicarse correctamente.



Paso 2: Repetir el proceso con la animación

1. Seleccioná el archivo **.fbx** de la animación que importaste (sin skin).
2. En la pestaña **Rig**, repetí el mismo proceso:
 - **Avatar Definition: Create From This Model**
 - **Root Node: "Hips"**
3. Hacé clic en **Apply**.

Así nos aseguramos de que tanto el modelo como la animación están utilizando rigs compatibles.



Paso 3: Agregar el componente Animator al personaje

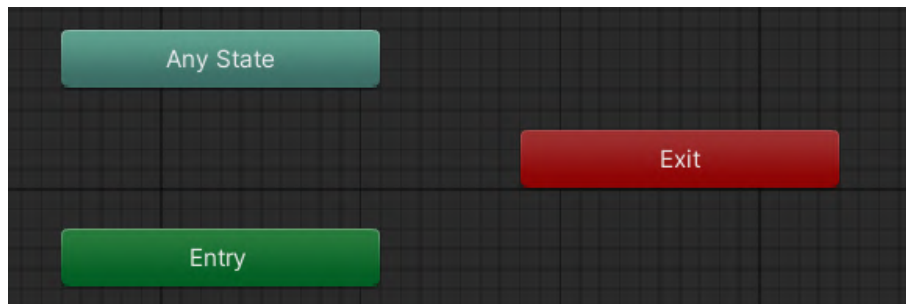
1. Seleccioná el personaje en la escena.
2. En el **Inspector**, hacé clic en **Add Component**.
3. Buscá y agregá **Animator**.
4. En el campo **Avatar**, seleccioná el avatar que creaste en el paso anterior.

💡 *Este paso le dice a Unity cómo debe interpretarse el esqueleto del personaje al reproducir animaciones.*

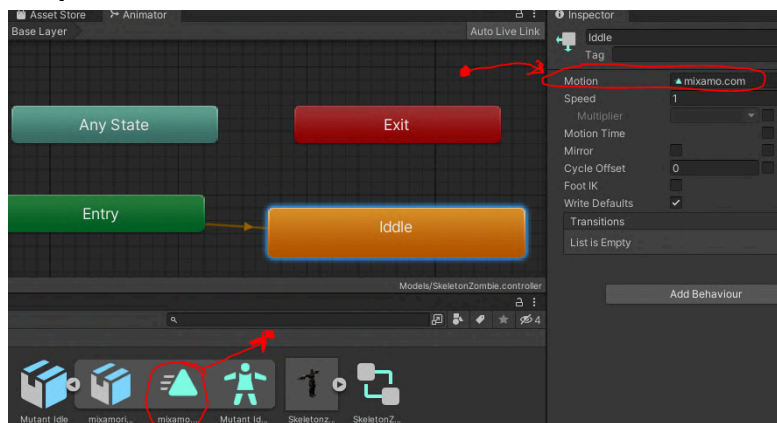
Paso 4: Crear y asignar un Animator Controller

1. En el panel **Project**, hacé clic derecho → **Create > Animator Controller**.
2. Nombralo con el nombre del personaje (ej: **Player_Controller**).
3. Seleccioná el personaje en la escena.
4. En el campo **Controller** del componente Animator, asigná el archivo que acabás de crear.

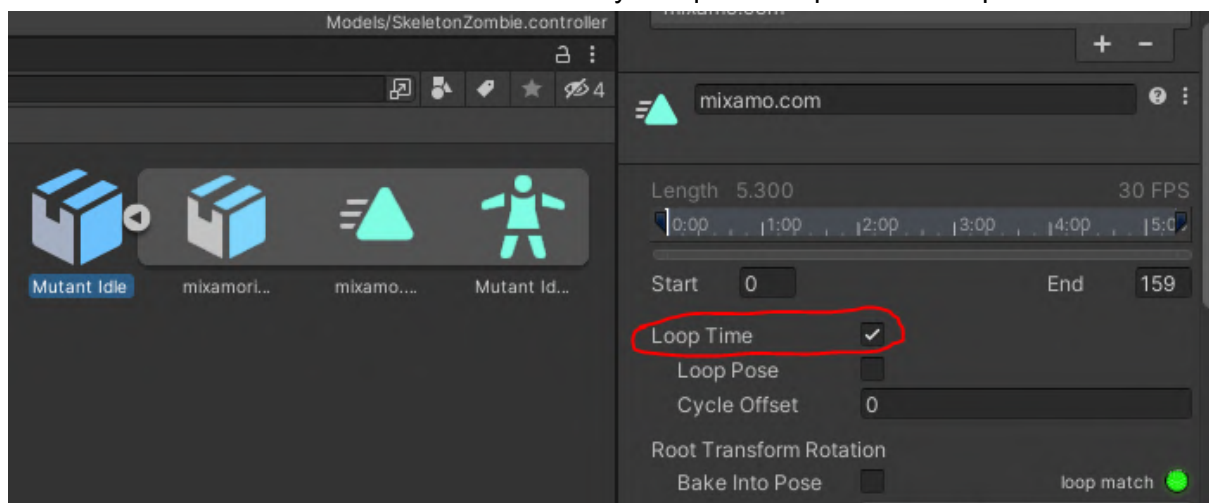
Antes:



Después:



Si le damos Play, veremos que nuestro personaje realiza la animación, pero NO la repite. Por lo tanto deberemos ir a nuestra animación y chequear la opción de “loop”



Le damos a “Apply” y listo!

¿Qué es el Animator Controller?

Es un sistema visual que permite organizar y conectar animaciones entre sí, definiendo en qué momento se reproducen, qué condiciones las activan y cómo deben transicionar unas con otras.

Materiales y recursos adicionales.

Animator Controller:

<https://docs.unity3d.com/Manual/class-AnimatorController.html>

Animation:

<https://docs.unity3d.com/2022.3/Documentation/Manual/AnimationSection.html>

Mixamo:

<https://www.mixamo.com/#/>

Próximos pasos.

En la próxima clase profundizaremos en el código aprendiendo sobre “Eventos” y “Delegados” para ver nuevas maneras de organizar nuestros códigos y experimentar con las interacciones entre los mismos.

¡Una nueva situación en TalentoLab!



El cliente ha quedado fascinado con las plataformas y el diseño vertical del nivel, pero ahora siente que algo crucial está faltando: el movimiento del personaje y los objetos carece de fluidez y realismo. Quieren que Nexus sea un juego donde cada salto, cada ataque y cada interacción se sienta vivo, inmersivo y lleno de personalidad.

El equipo de diseño de TalentoLab tiene ahora una misión clara: dar vida al personaje y a los elementos clave del juego a través de animaciones. Este paso no solo mejorará la experiencia visual, sino que también aportará un nivel de detalle que cautivará a los jugadores.

Ejercicios prácticos:

La solicitud del cliente:

El cliente quiere ver animaciones realistas y bien integradas en el juego. Necesitamos trabajar tanto en el personaje principal como en los elementos interactivos para que cada movimiento esté lleno de vida.



Luigi se encuentra sobrecargado de trabajo y desea delegar las animaciones básicas de los personajes para que puedan empezar a orientarse en el ambiente.

1) Configurando el Animator:

- Imagina que el personaje principal de Nexus está a punto de recibir sus primeros movimientos.
- Tu tarea será configurar el Animator para que pueda controlar estas animaciones de manera fluida y lógica.
- Crearás un diagrama básico en el Animator donde las animaciones transicionen entre sí dependiendo de las entradas del jugador.

2) Importando animaciones externas:

- El cliente ha proporcionado un paquete de animaciones que desean utilizar en el juego. Estas animaciones fueron creadas por un [estudio asociado](#) y representan movimientos clave, como ataques, interacciones con el entorno y animaciones de inactividad.
- Tu misión será importar estas animaciones y adaptarlas al rig del personaje principal en Unity.

3) Integrando animaciones al gameplay:

- Una vez configurado el Animator, conecta las animaciones a los controles existentes del personaje, asegurándote de que cada acción (como correr, caminar, saltar o atacar) se sienta natural y fluida.



Buenos Aires
aprende
Agencia de Políticas para el Futuro

BA Buenos
Aires
Ciudad