****

**Proyecto Final Unity3D**

**2ºEvaluación**

**Alvaro Vian Hernández**

**Jesús Merino Quevedo**

**Salvador Gurucharri**

**Javier Merchán**

**Contenido**

[1. MENÚS 3](#_Toc32862917)

[1.1 MENÚ PRINCIPAL 3](#_Toc32862918)

[1.2 MENU JUEGO 4](#_Toc32862919)

[1.3 MENÚ INFORMACIÓN 5](#_Toc32862920)

[2. TERRENO 6](#_Toc32862921)

[2.1 TEXTURAS 7](#_Toc32862922)

[2.2 COLECCIONABLES 8](#_Toc32862923)

[3. PERSONAJE 9](#_Toc32862924)

[3.1 DISEÑO Y ANIMACIÓN 9](#_Toc32862925)

[3.1.1 DISEÑO 9](#_Toc32862926)

[3.1.2 ANIMACIÓN 10](#_Toc32862927)

[3.2 ANIMAR PERSONAJE 10](#_Toc32862928)

[4. ENEMIGO 13](#_Toc32862929)

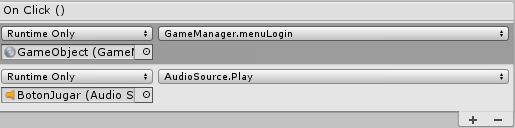
# MENÚS

Se ha creado un GameManager para controla los cambios de menús al igual que se le indica que no se destruya el menú actual al cargar otra escena, los menús han sido creados como “**Escenas (Scenes)”.**

Todos los menús cuentan con botones que se han incorporado como elementos “Button”. Se ha creado un **GameObject** al que se la ha asignado el script “**GameManager**”, después se le asigna ese **GameObject** al botón en su “**On Click()**” y a su vez asignarle su método correspondiente que se encargara de hacer la llamada a su menú correspondiente.

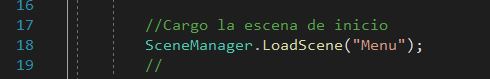
También se le añadió un **Audio Source** para cuando se pulsara los botones.





## MENÚ PRINCIPAL

El menú principal está definido en el GameManager como la escena de inicio.



Cuenta con tres botones para distintos eventos:

1. **Botón Jugar**: Carga el menú de login para iniciar sesión y empezar a jugar si el usuario existe en la base de datos.
2. **Botón Información**: Carga el menú de información donde se encuentra los controles e información de los desarrolladores.
3. **Botón Salir**.

También se le añadió un **AudioSource** para tela música de fondo que se inicia con el juego y un paisaje como fondo.



## MENU JUEGO

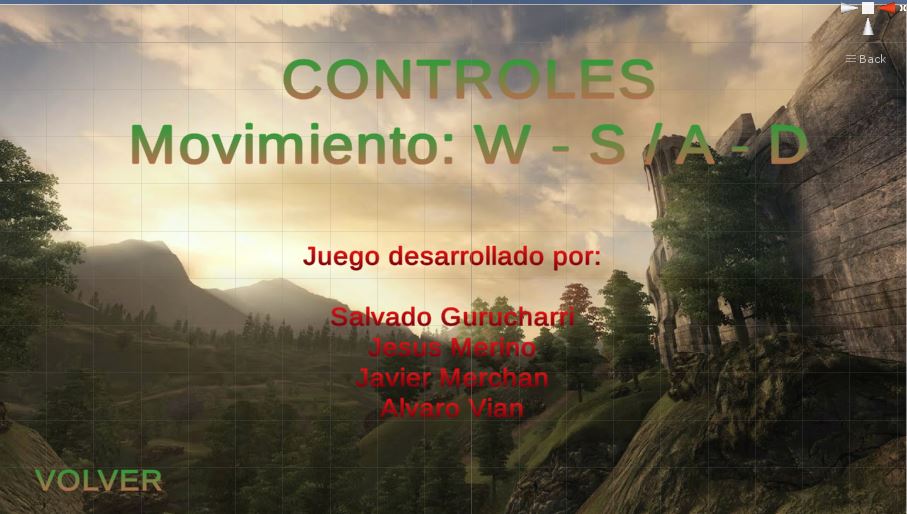
El botón “**Jugar**” del menú principal nos llevara hasta esteme menú donde tendremos que introducir un nombre de **usuario** y **password** que serán validados con la base de datos para poder acceder al juego, si no tiene cuenta no accederá y deberá registrarse pulsando en el botón “**Registrarse**” el cual te redirigirá a la página web donde poder realizar el registro.

El menú juego consta con dos “**TextMeshPro**” y dos “**InputField**” para introducir los datos.



## 1.3 MENÚ INFORMACIÓN

En el menú de información se encontrarán los controles para mover al jugador y los nombres de los desarrolladores del juego y de la plataforma web del juego, también cuenta con el botón volver y mismo fondo que en el menú principal.



# 2. TERRENO

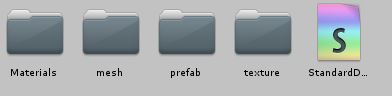
La base del terreno ha sido diseñado con el elemento **Terrain** y modificado con los pinceles para crear distinta alturas en el terreno y montañas para limitar el borde del mapa.



## 2.1 TEXTURAS

Todas las texturas utilizadas se han descargado e importado desde la propia **Asset Store** que nos ofrece Unity, todas de forma gratuita.

Se han utilizado texturas importadas con sus materiales, prefabs y algunas como con sus animaciones.



Los materiales que se han utilizado para diseñar el mapa son:

* Césped
* Arboles
* Rocas
* Setas
* Casa
* Estatua de un oso
* Arbustos
* Coleccionables (manzanas, trampas…etc.)

El elemento **Terrain** nos permite asignarle una capa con el Asset que nosotros queramos pintar con un pincel que nos proporciona, el pincel nos permite asignarle la cantidad y separación para pintar.





Por ejemplo para los arboles ya contenía su animación, y se le ha añadido un elemento viento que proporciona Unity “**Wind Zone**” para darle más realismo a nuestro escenario.



Se puede colocar su posición para que el viento vaya en una dirección u otra.

## 2.2 COLECCIONABLES

# 3. PERSONAJE

Los personajes utilizados han sido descargados desde este sitio web <https://www.mixamo.com/> que nos permite su descarga de forma gratuita siempre y cuando se utilice de forma no comercial. En esta página web nos permite descargarnos el diseño del personaje como animaciones de movimiento queramos.

## DISEÑO Y ANIMACIÓN

### 3.1.1 DISEÑO

Hemos elegido el diseño de **THE BOSS** como el personaje que controlara el jugador.



### 3.1.2 ANIMACIÓN

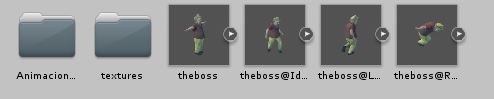
Para las animaciones nos hemos descargado:

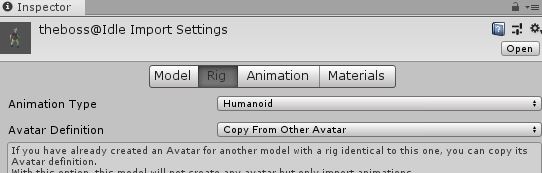
* Animación para correr hacia delante.
* Animación para cuando está parado.
* Animación para cuando gira a la izquierda.

Tantos la animación para correr nos sirve para cuando vamos hacia detrás, al igual que la animación de girar a la izquierda Unity nos da una opción para poder crear otra hacia la derecha invirtiéndola.

## ANIMAR PERSONAJE

Primero se tienen que importar el diseño y animación a Unity descomprimiendo los archivos descargados y arrastrándolos a la carpeta Unity donde queramos colocar los elementos.

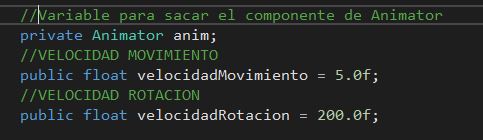




Se debe de pinchar en las cuatro animaciones que nos aparecen en la primera imagen y marcarlas como “**Humanoides**”, ver segunda imagen. Luego se arrastra a nuestro mapa para colocar al personaje.

Necesitaremos un Script para mover nuestro personaje, en el que añadiremos:

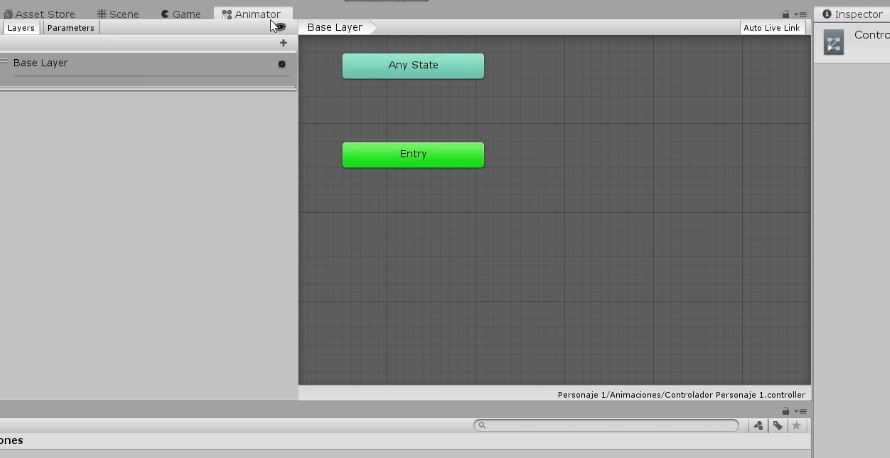
* Velocidad de movimiento
* Velocidad de rotación
* Variable para sacar el componente de Animator (ej**. prívate Animator anim**;)
* Variables X, Y para las posiciones



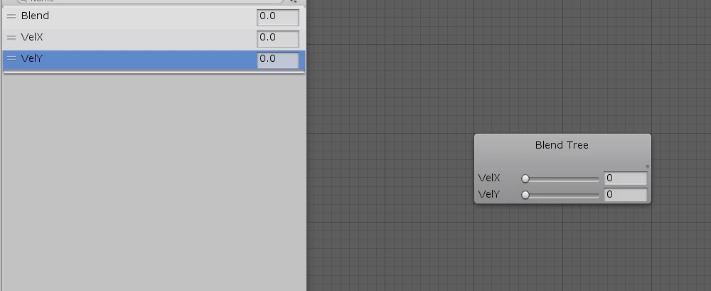
Luego de realizar el código correspondiente para mover al jugador con los controles que se deseen se realizara a implementar las animaciones para darle realismo al movimiento del personaje. Se recomienda crear una nueva carpeta llamada por ejemplo “**Animaciones**”, dentro de dicha carpeta se creará un “**Animator Controller”** y se le añadirá las animaciones que nos hemos descargado.



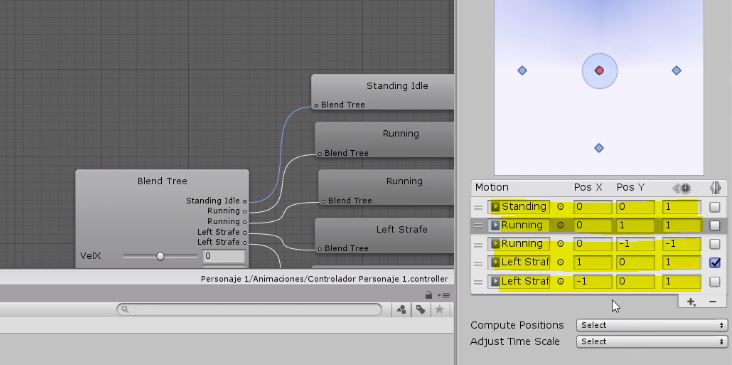
Le damos doble click al **Animator Controller** y se nos abrirá la siguiente ventana:



Ahí se le dará click derecho (*Create State\From New Blend Tree ),* al Blend Tree hemos creado le daremos doble click, ahí le añadiremos dos variables Float llamadas VelX y VelY.

**

Ahí ya le podremos añadir sus animaciones arrastrándolas:



# ENEMIGO

## DISEÑO Y ANIMACIÓN

### 4.1.1 DISEÑO

Hemos elegido el diseño de **ZOMBIEMASTER** como el personaje que será controlado por la IA, en forma de enemigo del jugador.

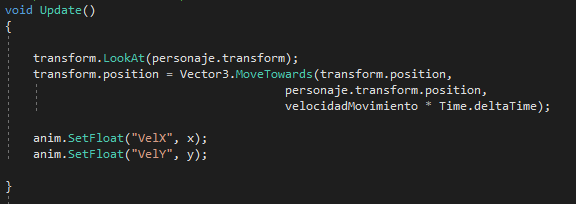


### 4.1.2 ANIMACIÓN

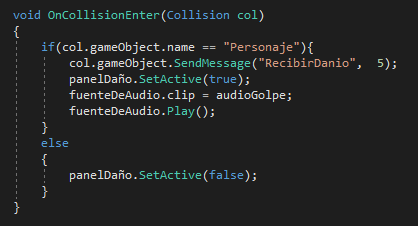
La animación del enemigo consiste en balancearse hacia los lados girando la cabeza. Se implementa de igual forma que las animaciones del personaje, mediante un objeto **Animator**.

## 4.2 PERSECUCIÓN AL JUGADOR

La persecución al jugador se hará desde un script aplicado al Asset de Enemigo. Desde el método **Update**, se busca la posición del jugador, haciendo mirar hacia ella, y después desplazándose hacia ella, con velocidades “VelX” y “VelY”;



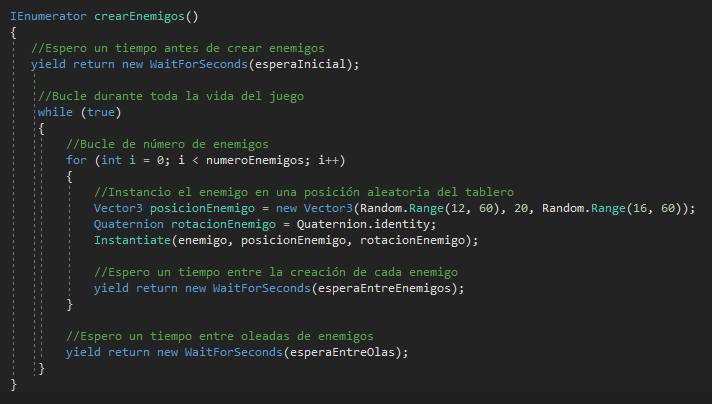
Una vez el enemigo alcanza al jugador, le causa daño. Esto se hace desde el método **OnCollisionEnter**, que se ejecuta en el momento en el que el componente **Collider** **del personaje** entra en contacto con el del **enemigo**.



Como vemos, le hace daño, activa el **canvas** del daño, y reproduce un sonido de un zarpazo.

## INSTANCIACIÓN / SPAWN

La instanciación de nuevos enemigos se hará en tiempo de ejecución. Se ejecutará este método de forma ininterrumpida, contenido en un script, el cual será llamado desde el **GameManager** en el momento del inicio del juego.



De esta forma, pasándole en unos parámetros la espera entre creación de cada enemigo, oleadas, y el número de ellos a instanciar, harán spawn en una posición aleatoria del mapa, yendo de forma directa a por el jugador.