

---

# MUCOSITY

Diego Matilla De La Cierva

Desarrollo Y Diseño De Videjuegos

---

---

## 1.Introduccion, objetivos y motivación del Proyecto

Esta práctica consistirá en programar y realizar un pequeño juego arcade, este proyecto se centrará en la programación del movimiento de los personajes, estos simplemente se moverán de izquierda a derecha y deberán de evitar chocar con objetos de distinto color al suyo. La parte principal del programa del personaje se centrará en su movimiento, que se basará en estados de animaciones. Dentro de este código también se encontrará la función Game Over que se ocupa de activar el canvas Game Over y destruir a los personajes.

Con este proyecto me propuse desde el inicio aprender a programar con la herramienta Unity ya que mi conocimiento con esta era muy básico. Este proyecto lo realicé a solas con el fin de aprender sin intención de publicarlo, pero más tarde encontré a un artista interesado en crearlo y se decidió publicarlo.

---

---

## 2 Descripción técnica de los componentes del proyecto.

### -Movimiento mediante Estados de Animación:

#### Función:

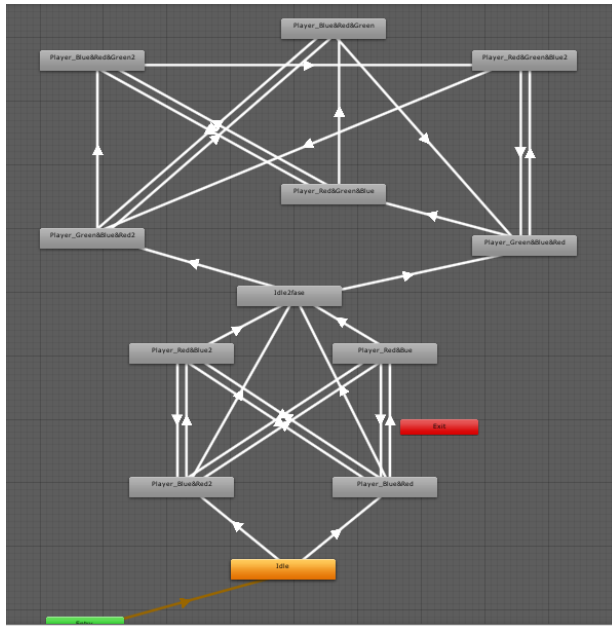
Este código se ocupará del movimiento de los personajes. Dependiendo de la animación actual en la que se encuentre el GameObject Player, se ejecutara una animación u otra dependiendo del movimiento que haya ejecutado el usuario.



## Descripción Técnica:

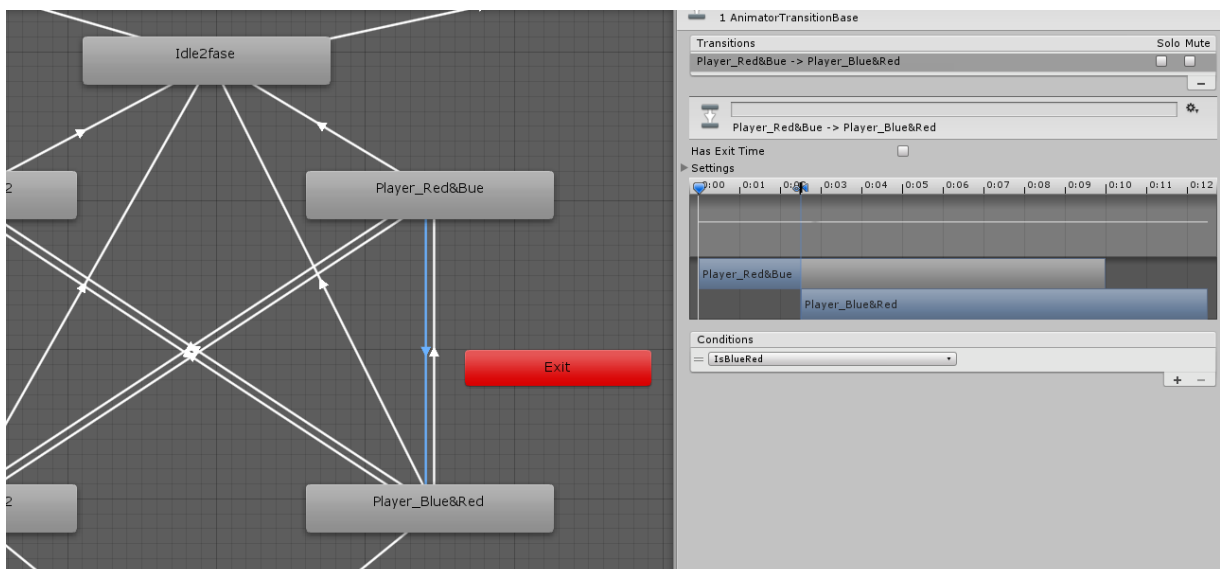
### -Movimiento 2 Player:

El juego empieza con dos personajes, los cuales están dentro del Gameobject Player. El código del jugador se centra en preguntar todo el rato en que animación se encuentra el Player y que tecla a pulsado el jugador para poder usar la animación correspondiente para que los personajes se muevan. Para evitar ciertos problemas en el juego bloquea los Input del jugador para que no se pueda apretar los controles en mitad de una animación y esto se hará mediante una variable bool que se pondrá true con un evento que se pone en el comienzo de todas las animaciones y al acabar el movimiento se cambia a false con otro evento al final de las animaciones.



Layers	Parameters
QName	
= RedLeft	<input type="checkbox"/>
= IsBlueRed	<input type="radio"/>
= IsBlueRed2	<input type="radio"/>
= IsRedBlue	<input type="radio"/>
= IsRedBlue2	<input type="radio"/>
= Is2Fase	<input type="radio"/>
= IsGreenBlueRed	<input type="radio"/>
= IsGreenBlueRed2	<input type="radio"/>
= IsBlueRedGreen	<input type="radio"/>
= IsBlueRedGreen2	<input type="radio"/>
= IsRedGreenBlue	<input type="radio"/>
= IsRedGreenBlue2	<input type="radio"/>
= IsPoint	<input type="radio"/>

Al comienzo solo partes con dos jugadores por lo que solo existen 4 tipos de movimientos que tendrás que conectarlos con sus anteriores y posteriores posibles animaciones. El movimiento al tratarse de un movimiento en circulo cada animación solo dispone de dos conexiones posibles (girar derecha o girar izquierda), como he explicado antes el código simplemente será una gran lista de If que se ocuparan de preguntar en que animación se encuentra el Player mediante el código "anim.GetCurrentAnimatorStateInfo(0).IsName("NombreAnimación")". A este código se le suma la condición de si has pulsado izquierda o derecha. Dependiendo de esto llamaremos al componente Animator del Player y activaremos el SetTrigger correspondiente para que se active la animación correspondiente.



---

### -Movimiento 3 Player:

Cuando el jugador llega a cierta puntuación el código que se ocupa de esta misma hace que el Animator del Player active el SetTrigger correspondiente para que pase a la segunda fase con una nueva formación en los jugadores. En cuanto al funcionamiento de esta 3 fase es completamente igual a la anteriormente explicada fase.

