

## DESCRIPCION DE LA TAREA:

- Objetivo: Crear un entorno en el cual el usuario pueda "señalar" a otros objetos de Unity. Cada uno de esos objetos tendrá un a melodía asociada. El objetivo consiste en que a través de una tecla ("p") el usuario pueda indicar a cada uno de esos objetos (solo si les esta mirando ("apuntando con el ratón")) que reproduzcan su melodía asociada.

## IMPLEMENTACIÓN:

Si las imágenes no se ven, los scripts están en las rutas indicadas.

Bloque	Paso	Detalles a mencionar	Referencias
Creación de objetos básicos para la escena			
	Crear el suelo	- Usar cubo en vez de plane (para escapar de configuraciones físicas)	<a href="https://www.youtube.com/watch?v=G98dFZ2MCxc&amp;list=PLZ1b66Z1KFKik2g8D4wrmYj4yein4rCk8&amp;index=1">https://www.youtube.com/watch?v=G98dFZ2MCxc&amp;list=PLZ1b66Z1KFKik2g8D4wrmYj4yein4rCk8&amp;index=1</a> Minuto 6:30 - 9:21
	Agregar un material al suelo	- Valores de ejemplo en escala: x = 25, Y = 0.25, Z = 25	<a href="https://www.youtube.com/watch?v=eOlyQc1cBL4&amp;list=PLZ1b66Z1KFKik2g8D4wrmYj4yein4rCk8&amp;index=2">https://www.youtube.com/watch?v=eOlyQc1cBL4&amp;list=PLZ1b66Z1KFKik2g8D4wrmYj4yein4rCk8&amp;index=2</a> Minuto 2:20 - 8:20
Creacion de personaje principal (FPS Controller)			
	A continuación EXTENSIÓN CREACION DE PERSONAJE PRINCIPAL (FPS Controller)		
Configurar herramienta para la dirección del personaje principal			
	Crear "vara" para dirigir	- En el video lo llama weapon pero es el mismo principio. - El escenario tan elaborado del video da igual, lo que tenemos que aprender es la metodología. - Especial atención a la parte del Input (permite personalizar una tecla de teclado a un determinado comportamiento).	<a href="https://www.youtube.com/watch?v=LNmpFPk9eV4&amp;list=PLZ1b66Z1KFKik2g8D4wrmYj4yein4rCk8&amp;index=10">https://www.youtube.com/watch?v=LNmpFPk9eV4&amp;list=PLZ1b66Z1KFKik2g8D4wrmYj4yein4rCk8&amp;index=10</a> Minuto: 0:00 - 3:35
	Crear animación para indicar cuando el personaje principal usa la vara		<a href="https://www.youtube.com/watch?v=LNmpFPk9eV4&amp;list=PLZ1b66Z1KFKik2g8D4wrmYj4yein4rCk8&amp;index=10">https://www.youtube.com/watch?v=LNmpFPk9eV4&amp;list=PLZ1b66Z1KFKik2g8D4wrmYj4yein4rCk8&amp;index=10</a> Minuto 3:40 - 13:30
Crear musicos			
	Crear objetos de musicos	- Simplemente requiere crear 5 empty objects que actuarán como músicos. - Acuerdate de que CTRL + d, te permite duplicar un objeto en la escena	Referencias en videos de "Creacion de Objetos básicos para la escena".
Configuración de Raycast			
	Creacion de un empty object al que herede de Cylinder (al mismo nivel que la "vara" para dirigir) para asociarle el script de Raycast que llamaremos "directing"	- En el video lo llama "Attacking"	<a href="https://www.youtube.com/watch?v=Q_o74dbn3ZU&amp;list=PLZ1b66Z1KFKik2g8D4wrmYj4yein4rCk8&amp;index=11">https://www.youtube.com/watch?v=Q_o74dbn3ZU&amp;list=PLZ1b66Z1KFKik2g8D4wrmYj4yein4rCk8&amp;index=11</a> Minuto 0:00 - 1:45
	Creacion y asociacion del script de Raycast al objeto "directing"	- Ver el Script en EXTENSIÓN DE CONFIGURACION DE RAYCAST	<a href="https://docs.unity3d.com/ScriptReference/Physics.Raycast.html">https://docs.unity3d.com/ScriptReference/Physics.Raycast.html</a> <a href="https://docs.unity3d.com/es/530/ScriptReference/Collider.html">https://docs.unity3d.com/es/530/ScriptReference/Collider.html</a>
	Creación de Audio Source en personaje principal	- Ver una captura en EXTENSIÓN DE CONFIGURACION DE RAYCAST	
	Asociación de canciones en las variables "Public" del script de Raycast	- Ver una captura en EXTENSIÓN DE CONFIGURACIÓN DE RAYCAST - Las canciones las guardaremos en una carpeta Assets -> Sounds	

## EXTENSIÓN: CREACIÓN DE PERSONAJE PRINCIPAL (FPSController):

Nuestro siguiente paso va a ser poder agregar un personaje principal controlado por ratón y teclado, es lo que se conoce en Unity como "FPS Controller":

3.1: Comenzamos agregando un Empty Object que hará el rol de nuestro personaje, para ello vamos a boton derecho raton-> create empty.  
3.2: La "Main Camera" es la encargada de mostrar el usuario la vista que va a tener en el juego (lo que sería la cámara del juego) para ello lo que vamos a hacer es que herede de nuestro empty object creado en el paso anterior para que todos los ajustes, movimientos, cambios... quese hagan en el personaje, impliquen tambien que la cámara lo siga. Para ello arrastramos la "main Camera" dentro del empty object (al que a partir de ahora llamaremos "First Person Player"). De forma que quede asi:



3.3: Sin embargo la cámara en el escenario no se encuentra en la posición del First Person Player, para terminar de lograr esto seleccionamos la cámara y en sus propiedades de Transform (Inspector) boton derecho raton-> reset.

3.4: Finalmente vamos a elaborar el script que usaremos para realizar el movimiento y que la cámara pueda girar. (lo guardamos en una carpeta nueva Assets -> Scripts)

Cosas importantes a destacar dentro del script:

- Input nos permite obtener la entrada de teclado o mouse o incluso mapear un input correspondiente a determinada tecla del ratón (mas adelante).
- Transform define todas las propiedades del cuerpo físicas y de movimiento.
- Las variables **public** son valores que son introducidos desde el inspector del objeto correspondiente al que está asociado el script en Unity a la hora de ejecutar el Script.

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class FPSController : MonoBehaviour
{
    public Camera camera;

    public float horizontalSpeed; // velocidad de rotación con el ratón eje x.
    public float verticalSpeed; // velocidad de rotación con el ratón eje y.

    // variables auxiliares para el movimiento del ratón
    float h;
    float v;

    // Start is called before the first frame update
    void Start()
    {
    }

    // Update is called once per frame
    void Update()
    {
        changeCamera();
        move();
    }

    void changeCamera()
    {
        h = horizontalSpeed * Input.GetAxis("Mouse X");
        v = verticalSpeed * Input.GetAxis("Mouse Y");

        transform.Rotate(0, h, 0);
        camera.transform.Rotate(-v, 0, 0);
    }

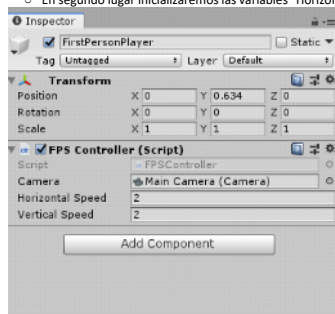
    void move()
    {
        if (Input.GetKey(KeyCode.W))
        {
            transform.Translate(0, 0, 0.3f);
        }
        else if (Input.GetKey(KeyCode.S))
        {
            transform.Translate(0, 0, -0.3f);
        }
        else if (Input.GetKey(KeyCode.A))
        {
            transform.Translate(-0.3f, 0, 0);
        }
        else if (Input.GetKey(KeyCode.D))
        {
            transform.Translate(0.3f, 0, 0);
        }
    }
}

```

RUTA: /src/Assets/Scripts/FPSController.cs

Tras esto arrastramos el script al inspector del objeto "FirstPersonPlayer" para asociarlo con el objeto.

- 3.5: Nos queda simplemente rellenar los valores "Public" por los correspondientes objetos y valores desde Unity:
- En primer lugar a la variable "camera" le asociaremos el objeto "MainCamera" en Unity
  - En segundo lugar inicializaremos las variables "Horizontal Speed" y "Vertical Speed" desde el Inspector.



Tras esto ya tenemos nuestro personaje que puede girar la cámara con el ratón y moverse con las teclas del teclado: "w", "a", "d", "s".

## EXTENSIÓN: CONFIGURACIÓN DE RAYCAST:

- Script de Raycast:

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class DirectorRaycast : MonoBehaviour
{
    // Variables
    public int i = 0;
    public int j = 0;
    public AudioClip[] canciones;
    public AudioSource fuenteAudio;

    // Start is called before the first frame update
    void Start()
    {
        // Inicialización de variables
        i = 0;
        j = 0;
        // Asignación de la fuente de audio
        fuenteAudio = GetComponent();
    }

    // Update is called once per frame
    void Update()
    {
        // Raycast horizontal
        RaycastHit[] hits = Physics.RaycastAll(transform.position, Vector3.right, 100);

        // Si hay hits, se ejecuta la siguiente acción
        if (hits.Length > 0)
        {
            // Seleccionamos el primer hit (el más cercano)
            RaycastHit hit = hits[0];

            // Obtenemos el nombre del objeto que ha sido hitado
            string nombreObjeto = hit.transform.name;

            // Si el nombre del objeto es "Mesa", se ejecuta la siguiente acción
            if (nombreObjeto == "Mesa")
            {
                // Seleccionamos la canción que se va a reproducir
                AudioClip clip = canciones[i];

                // Reproducimos la canción
                fuenteAudio.clip = clip;
                fuenteAudio.Play();

                // Incrementamos el índice de la canción
                i = i + 1;
            }
            else
            {
                Debug.Log("NO CHOCA" + j + "veces");
                j = j + 1;
            }
        }
    }
}

```

```

    }

    // Update is called once per frame
    void Update()
    {
        // Raycast horizontal
        RaycastHit[] hits = Physics.RaycastAll(transform.position, Vector3.right, 100);

        // Si hay hits, se ejecuta la siguiente acción
        if (hits.Length > 0)
        {
            // Seleccionamos el primer hit (el más cercano)
            RaycastHit hit = hits[0];

            // Obtenemos el nombre del objeto que ha sido hitado
            string nombreObjeto = hit.transform.name;

            // Si el nombre del objeto es "Mesa", se ejecuta la siguiente acción
            if (nombreObjeto == "Mesa")
            {
                // Seleccionamos la canción que se va a reproducir
                AudioClip clip = canciones[i];

                // Reproducimos la canción
                fuenteAudio.clip = clip;
                fuenteAudio.Play();

                // Incrementamos el índice de la canción
                i = i + 1;
            }
            else
            {
                Debug.Log("NO CHOCA" + j + "veces");
                j = j + 1;
            }
        }
    }
}

```

- RUTA: /src/Assets/Scripts/DirectorRaycast.cs
- Audio Source en el personaje y asociación de las canciones en las variables:

