



Ciclo: Animaciones 3D, Juegos y Entornos Interactivos

Curso: 2020/21

Módulo: Desarrollo de Entornos Interactivos Multidispositivo

Nombre y apellidos: César García Viana

EXAMEN TEÓRICO

Grupo A

Escribe tu nombre y apellidos en la cabecera de este documento, y a continuación explica qué herramientas vistas a lo largo del curso utilizarías para lograr los objetivos planteados en el ejercicio práctico, explicando por qué esas y no otras:

Interactividad y restricción de movimiento

La manera más correcta de mover un personaje, sería mediante el uso de Rigid Bodies, utilizando fuerzas que empujen el personaje desde un lado u otro. En este caso, no estamos utilizando el mundo físico, por lo que utilizando las propiedades de transform (concretamente 'translate', nos valdría). Por ejemplo, en Unity con utilizar un método propio Vector3.up (y multiplicando por Time.deltaTime y la velocidad que asignes al objeto)

Seguimiento del jugador con la cámara

Existen varios metodos para hacer que la cámara siga al jugador. Uno de ellos es el método LookAt(), en él la cámara se mantiene estática pero rota mirando al gameObject que tenga asignado. Otra manera es asignarle los valores de ese gameObject, consiguiendo que le siga (pudiendo modificar esos valores con sumas o restas para colocarlo en la posición deseada). A esta última forma se le puede añadir suavidad (habría que crear nuevas variables como la de velocidad de suavidad), que, mediante un método propio de unity, que necesita de: la posición del objeto, de la cámara, una velocidad de suavizado y una variable de velocidad de camara, consigue suavizarla.

Creación de elementos (columnas) de forma aleatoria

Primero tenemos que tener ese objeto (columna en este caso) como un prefab, a continuación, un script asociado a un empty object, deberá contener un método que instancie dicho prefab (normalmente si se hace automático se le añaden valores aleatorios a su posición inicial me-



diante `random.range` (para darle un rango determinado). A continuación realizamos una corrutina que utilice dicho metodo y con un tiempo de espera que se indique con el `WaitForSeconds`. La corrutina se instancia en el start para que, al comenzar el juego, se creen dichos elemntos cada cierto tiempo, y con el componente aleatorio en el método que lo instancia

User Interface (tanteos)

para poder utilizar la UI en Unity hay que llamar a su librería (`UnityEngine.UI`). Se crea un texto, con su canvas, y se ajusta en pantalla. A continuación en algún script (en este caso lo he asignado al de crear columnas, ya que uno de los textos es un contador de columnas), se crea un `serializefield` para asignarlo en unity y poder modificarlo desde el código. En este caso he creado un cronómetro que registre y muestre los segundos que pasan en pantalla, y un contador de columnas que aumente cada vez que se instancie una.

Colisiones:

Para generar colisiones, es necesario al menos un `rigidbody` y un `collider`, con el `is trigger` activado en el `collider`, podemos crear un método que detecte cuando el `rigid body` (la esfera) entre en contacto con dicho `collider` (obstaculo), que destruya el obstaculo cuando, como hemos dicho, se activa.

Aunque no lo he conseguido, aquí tendríamos que hacer también que, cuando colisione y se destruya, el contador de columnas baje, según el número de columnas destruidas. La manera de hacerla sería, creo, llamar a la clase que tenga dicho contador, (interacción entre clases), y restar 1 a dicha variable, lo que deberá verse en pantalla.

Entrega y evaluación

Cuando tengas completo el documento, expórtalo a pdf con este formato:

Apellidos_nombre_ExTco1EV.pdf



Guárdalo en el misma carpeta que el documento Word, dentro del repositorio, y súbelo a un commit de GitHub.