



Ciclo: Animaciones 3D, Juegos y Entornos Interactivos

Curso: 2020/21

Módulo: Desarrollo de Entornos Interactivos Multidispositivo

Nombre y apellidos: Alberto Sánchez Solana

EXAMEN TEÓRICO – 1ª EV

Grupo B

Escribe tu nombre y apellidos en la cabecera de este documento, y a continuación explica qué herramientas vistas a lo largo del curso, tanto de Unity como de programación, utilizarías para lograr los objetivos planteados en el ejercicio práctico

IMPORTANTE: no te limites a enumerarlas, explica por qué usarías esas y no otras:

Cada apartado valdrá 2 puntos, y la nota final representará el 25% de la nota final de la evaluación.

Interactividad y gestión del movimiento

Para el movimiento de la nave dentro de un metodo donde he agrupado todos los movimientos para que se lancen en el update, he utilizado dos métodos: el Input.GetAxis para el movimiento en horizontal el cual le he restringido el valor negativo para que no vaya hacia atrás. Seguramente de problemas en el momento que gire la nave ya que cambia de dirección... pero no he dado para más.

El movimiento de los botones LB y RB, he creado la interacción con el Input.GetButton ya que es el específico para el mando y permite que la acción se realice con tener mantenido el botón (no hay que soltar y apretar de nuevo para realizar la acción).

Para ambos he usado el Vector.up, Vector.down y Vector. Left ya que permiten de forma sencilla mover nuestro elemento cambiando entre 1 y -1 para caambiar su sentido.



Seguimiento del jugador con la cámara

Me falta la rotación inicial que tiene al mirar a la nave pero con el lookAt se consigue seguir con la mirada a la nave y modificando su posición poniendo la posición en y de la nave se consigue que el movimiento en el eje vertical sea mutuo.

Creación de elementos (enemigos) de forma aleatoria y a intervalos

Para crear múltiples elementos que hagan de obstáculos he usado un prefab. Un objeto que he llevado fuera de la escena y he guardado en el proyecto desde donde podré instanciar todos los que quiera y las cualidades que quiera, solo modificando a este primero.

Para meterlo en la escena se crea su script en un empty object donde estará la posición inicial que junto con unas variables aleatorias pero con límites, dejarán por todo el mapa los obstáculos y en el método start creo un bucle de 20 para que me aparezcan este número al principio.

Para los obstáculos que van saliendo según se desarrolla el juego, he creado una corrutina para fijar un tiempo de 4s base a que vayan apareciendo. Este intervalo se va modificando según van apareciendo las esferas. Iba a hacerlo usando la variable que va aumentando en el bucle dentro de la corrutina y la relación matemática con la mitad de tiempo que no he podido resolver todavía.

User Interface (tiempo transcurrido, nº de columnas y alerta)

Se crearía una serie de textos en nuestra escena y se les añadiría al script de movimiento de la nave (la alerta la pondría con un texto que saldría a pantalla cuando el Input.GetAxis de ambos ejes registre un número superior al limitado por el tamaño del plano) o al de colisionar (una corrutina que cuente los segundos y el número de esferas que se van creando, con la variable de la corrutina de instanciar enemigos).

Colisiones

En la nave he puesto otro script para las colisiones donde uso el método OnCollisionEnter para que se produzca la colisión física entre los elementos. Dependiendo de cual sea el nombre del objeto con el que choca en la nave se producirá una acción distinta.



Antes de todo esto por supuesto, he añadido un Capsule Collider para que pueda producirse el choque.

Entrega

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

Apellidos_nombre_ExTco1EV.pdf

Guárdalo dentro del repositorio, y súbelo en un *commit* de GitHub, el cual se acompañará al *Pull Request* del final del examen.