



Ciclo: Animaciones 3D, Juegos y Entornos Interactivos

Curso: 2020/21

Módulo: Desarrollo de Entornos Interactivos Multidispositivo

Nombre y apellidos:

Andrés Segade Vedenova

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

Usando los inputs de Unity para poder elegir fácilmente los botones que activan el movimiento y usando condicionales le cambiamos la velocidad y la dirección según necesitemos.

Seguimiento del jugador con la cámara

Cogiendo el componente transform del jugador para ponérselo a la posición de la cámara como su base para luego añadir o quitar distancia según queramos. Luego le añadimos un LookAt y la cámara mirará siempre al jugador.

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

Creamos por lo menos una variable que sirva de número aleatorio. Ese número se lo damos a un vector 3 en x, y o z, según nos convenga, que luego



sumaremos a la posición inicial del instanciador. Para que aparezcan a intervalos usamos una corrutina que active el método de crear enemigos cada cierto tiempo añadiéndole un `yield return waitforseconds` y ahí le ponemos un número o una variable que hayamos creado

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

Creamos variables serializadas para acceder a los textos, hacemos un `GetComponent` y en `Update` especificamos el texto que llevarán usando `nombreTexto.text = "Texto"`. Para hacer todo esto necesitamos añadir la librería de `using UnityEngine.UI`

Colisiones

Usamos un método `void OnTriggerEnter(Collider Other)`

A ese método le ponemos un `if` en el que especificamos lo que tiene que pasar cuando se choque con otro objeto que podremos buscar según su nombre o tag por ejemplo. Algunos ejemplos comunes de lo que se le puede añadir a los resultados de la colisión son:

Destruir el `gameObject`, Ocultar su `mesh`, Activar sonido, Instanciar una animación, Mandar un mensaje a pantalla, etc.

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.

