



Ciclo: Animaciones 3D, Juegos y Entornos Interactivos

Curso: 2020/21

Módulo: Desarrollo de Entornos Interactivos Multidispositivo

09/12/2020

EXAMEN PRÁCTICO – 1ª EV

Grupo B

Antes de comenzar, lee atentamente estas instrucciones y visiona el vídeo de muestra, disponible en la plataforma. A continuación, realiza las siguientes tareas:

1.- GitHub

Accede a la siguiente URL con el repositorio que usarás para hacer el examen:

https://github.com/CIFP-Jose-luis-Garci/Examen_1TR_GRB

Realiza un Fork en tu cuenta de GitHub, y clónalo a tu disco duro. A partir de ese momento, trabaja con el clon de **TU** repositorio. Realiza commits cada vez que des por cerrada una tarea (aunque no esté concluida del todo), y al final del examen realiza un "Pull Request" al repositorio original, que será el que se evaluará. La hora límite de entrega la indicará el profesor.

2.- Creación y desarrollo del proyecto Unity

Crea en el repositorio clonado un Proyecto Unity que usarás para crear el videojuego del examen.

Llama al proyecto de Unity con tu apellido y nombre (no te olvides de copiar el archivo "gitignore" dentro de la carpeta del proyecto antes de realizar el primer *commit*).

Crea una escena con los elementos y los scripts necesarios para lograr un videojuego similar al del vídeo. Deberás añadir estos elementos:

1. Un plano que servirá de suelo (la textura la tienes entre los archivos del examen). El del vídeo está escalado cuatro veces el del plano por defecto de Unity (tiene un tamaño de 40x40)
2. Copia la carpeta con la nave que encontrarás en el repositorio a tu carpeta de assets y añade el modelo "Viper" a tu escena



- a. NOTA: deberás escalarlo, y si es necesario, crear un *empty object* como padre para poder girarlo en la dirección que desees sin que afecte a sus coordenadas.

3. Una esfera que hará de enemigo

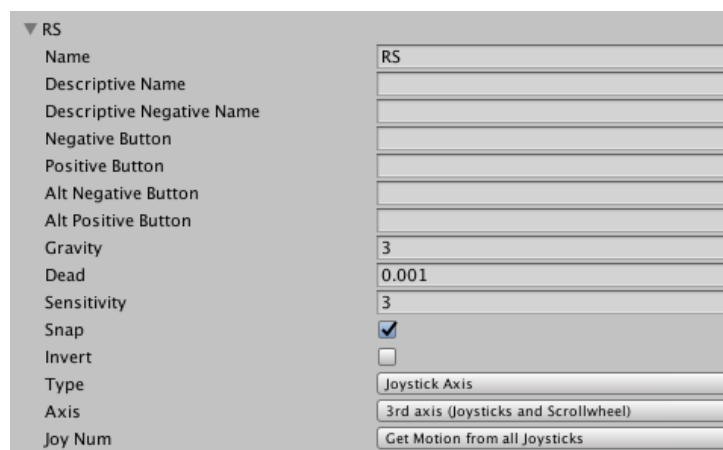
NOTA: te recomiendo ubicar el plano en las coordenadas (0,0,0)

3.- Cuestiones a resolver en el juego

Tu proyecto deberá resolver los siguientes retos:

A) Interactividad

- a. La nave deberá moverse por el espacio usando los mandos del joystick:
- Eje horizontal para moverse a izquierda y derecha
 - Eje vertical para moverse adelante. NO PUEDE IR HACIA ATRÁS
 - Botones LB y RB para subir y bajar (los códigos de tecla son "joystick button 4" y "joystick button 5" respectivamente)
 - Stick secundario par [rotar](#) en el eje Y
 - Deberás crear un eje nuevo en el Input Manager, con el nombre que elijas, asociado al eje 3 del joystick para conseguirlo, como se muestra en la imagen:



- b. Si la nave se sale de la zona restringida por el plano, o sube más de 20 unidades, saltará un aviso de ALERTA, tal y como muestra el vídeo



B) Seguimiento del jugador con la cámara

- La cámara seguirá a la nave tal y como se muestra en el vídeo: desde la distancia, ligeramente contrapicada, y modificando suposición en altura y profundidad siguiendo a la nave, pero en el movimiento lateral solo la seguirá apuntándola

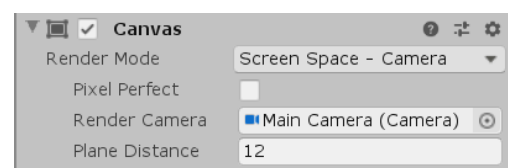
C) Creación de elementos (esferas) de forma aleatoria

- Al comenzar el juego, aparecerán 20 esferas repartidas aleatoriamente en el espacio tridimensional, pero dentro del terreno de juego (máximo 20 unidades de altura)
- Irán apareciendo esferas de forma aleatoria en el terreno de juego a intervalos regulares pero cada vez más rápido:
 - El intervalo de tiempo de creación debe depender del nº de esferas que hay en escena (por defecto, 4 segundos, si hay más de 5 esferas se crearán cada 2 segundos, y si hay más de 10, cada 1 segundo)

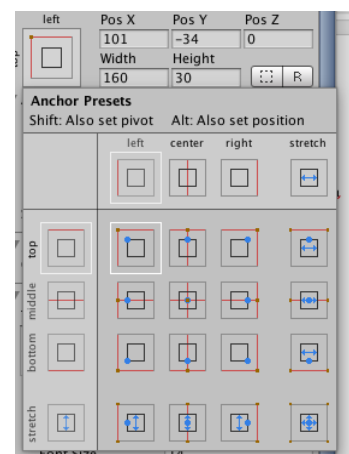
D) Tanteos: añade 2 textos a la pantalla que contengan la siguiente información:

- Tiempo transcurrido en el juego en segundos
- Nº de esferas que hay en la escena
- Alerta que aparecerá cuando el jugador se salga del terreno, también en altura

NOTA: Si quieres que el texto se ajuste a la vista de la cámara, deberás configurar el *canvas* como Render Mode: Screen Space – Camera, y arrastrar la cámara a la casilla de “Render Camera” (deberás ajustar la distancia para que no la tape el suelo):



También puedes hacer que los textos se ajusten a la pantalla. En el ejemplo de la derecha, el texto se ajustará a la esquina superior izquierda de la pantalla:





E) Colisiones:

- a. Cada vez que la nave choque con una esfera, ésta desaparecerá
 - i. Al desaparecer, deberá actualizarse el nº de esferas que aparece en la UI
- b. Si la nave choca con el suelo, dejará de verse, y dejará de estar controlada por el joystick

F) Sonidos: añade componentes de AudioSource para lograr estos sonidos:

- a. Un sonido de cabina constante: comienza a sonar automáticamente ("play on awake") y en bucle ("loop")
- b. Cada vez que se destruye una esfera, ejecuta el sonido de una explosión.

Recuerda hacer *commits* de forma regular para seguir el proceso de desarrollo.

Entrega y evaluación

Cuando se termine el plazo para realizar el examen, realiza un Pull Request a la cuenta original del repositorio, incluyendo el proyecto Unity y el examen teórico.

Copia la URL del Pull Request que aparecerá tras realizarlo (lo puedes encontrar en la URL: https://github.com/CIFP-Jose-luis-Garci/Examen_1TR_GRB/pulls) Entra en la plataforma, y en la entrega del examen escribe la URL copiada.

Criterios de calificación del examen práctico

1. Git Hub (**1 punto**)
 - a. Realizar correctamente el Fork y el Pull Request, así como el clonado del repositorio
 - b. Hacer *commits* de forma correcta y cada vez que se produce un cambio significativo en el proyecto
2. Organización del proyecto (**0,5 puntos**)
 - a. Los *assets* son organizados y nombrados de forma correcta en el proyecto



- b. El control de los elementos de la escena es correcto
 - c. La organización del código es adecuada
- 3. Interactividad y movimiento (**3 puntos**)
 - a. La nave se mueve correctamente con el joystick, y cuando se sale del plano es detectado
 - b. La cámara sigue correctamente al jugador, tal y como se ve en el vídeo de muestra
- 4. Control de tiempo (**1,5 puntos**):
 - a. Se crean las corrutinas adecuadas para controlar el tiempo de ejecución y los intervalos de creación de enemigos
- 5. Instanciar elementos (**1 punto**)
 - a. Uso correcto de los *prefabs* y su instanciación
 - b. Creación de ubicaciones aleatorias correctamente dentro del plano
- 6. Acceder a variables y métodos de otro GameObject (**1 punto**)
 - a. Se crean correctamente atributos y métodos públicos
 - b. Se accede desde un GameObject a las variables de otro y se ejecutan sus métodos
- 7. Colisiones (**1,5 puntos**)
 - a. Se añaden los componentes necesarios para detectar colisiones
 - b. Se realizan las tareas necesarias tras producirse una colisión
- 8. UI – User Interface (**1 punto**)
 - a. Se crea un canvas en la escena con los datos requeridos
 - b. Se actualizan correctamente los textos del canvas
- 9. Sonidos (**0,5 puntos**)
- 10. Materiales (**0,5 puntos**)
 - a. Se añade una textura de metal al suelo y una de color metálico a las columnas, tal y como aparece en el vídeo

La nota final del examen será un nº decimal entre 0 y 10 puntos