ASTEROIDS

<https://github.com/EddaBlanco/Asteroids/tree/main>

DOCUMENTACION:

-Escenas

-Assets

-GameObjects

-Mecanica

-Scripts

-Interaccion

Autores:

Marcos Morales García 210007

Edda Rosa Blanco Pavón 210310

Fundamentos de los videojuegos

-Escenas

Todo ocurre en la misma escena el SampleScene. Cada vez que queremos reiniciar el nivel porque hemos muerto o desde el menú de pausa hacemos un cambio de escena con el comando: ScenesManager.LoadScene(“SampleScene”);

A pesar de que no existe un cambio de escena, sí existe una ventana flotante que representa el menú de pausa.

-Assets

Los assets que hemos usado han sido los proporcionados en clase. Hemos creado varias carpetas para guardar los scripts, sprites, Prefabs y escenas. También hemos importado una serie de assets con assetstore de unity

<https://assetstore.unity.com/packages/2d/gui/puzzle-stage-settings-gui-pack-147389> el enlace de los assets que hemos utilizado para el menú de pausa.

<https://assetstore.unity.com/packages/2d/characters/2d-pixel-spaceship-two-small-ships-131545> el enlace a los assets para la bala nueva que hemos importado.

-GameObjects

Los GameObjects que hemos utilizado son los siguientes:

Ship: dentro de este gameObject tenemos otro objeto invisible llamado LanzaMisiles el cual es el spawner de las bullets que lanzará Ship. Ship tiene un Rigidbody, con gravedad 0 para simular el espacio, un Collider, para que al chocarse con los meteoritos se destruya y el script con su código.

Space: Es el objeto vacío de código, solamente posee un Sprite.

Main Camera: tiene un formato 2D orthpgraph y un script para que spawnen los enemigos.

Canvas: el elemento para que el jugador pueda interactuar con el juego.

También tenemos dos Prefabs los cuales son Bullet y Meteor para no tener q crear varios objetos.

-Mecanica

El juego consiste en conseguir puntuación acosta de eliminar meteoritos y evitar que estos choquen contigo, en cuyo caso el nivel se resetea (el contador, la posición y el spawn de meteoritos).

El juego tiene un menú de pausa que pone el Time.timeScale a 0 que para el tiempo de juego, al depender todos los movimientos del juego del Time.deltaTime, se pausa. Al reanudar el Time.timeScale se pone a 1 y todo retoma su mecánica inicial tal y como estaba antes del pause.

El Restart hace lo mismo que cuando un meteorito choca contra ti, vuelve a cargar la escena y se reinicia.

-Scripts

Ship:

- Pooling: Para el pooling hemos creado una lista de balas de 25 elementos la cual inicializamos en la función start. En el update obtenemos una bala de la lista, le damos posición y rotación y la activamos e incrementamos currentBulletIndex.

-Movimiento: El movimiento de la nave depende de dos variables, un impulso y una rotación, ambos dependen de Time.deltaTime.

-Espacio Infinito: El efecto de espacio infinito se logra mediante una comprobación de si la posición de la nave sigue dentro de unos límites en ejes x e y, en caso de salirse de los límites, se teletransporta al lado opuesto.

-SCORE: es una variable estática global que se usa para llevar el recuento de puntos.

PauseControl:

-Pause: El Time.timeScale se pone a 0 y el juego se para.

-Resume: El Time.timeScale se pone a 1 y el juego continua desde el punto parado.

-Restart: Reiniciamos la escena.

-Cerrar: Cerramos la aplicación.

Meteor:

-Collider: En colisión con la nave se reinicia la escena y el contador de SCORE se resetea a 0 igualmente.

EnemySpawner:

-Spawn: Los meteoritos aparecen de manera aleatoria a lo largo del eje x desde fuera del límite superior de la cámara, según evoluciona el tiempo de juego el número de meteoritos por minuto aumenta.

Bullet:

-Posición: Hace una comprobación de si las balas siguen dentro del espacio visionado por la cámara para que en caso de ya no estarlo poder desactivarlas.

-Collider: Si colisiona con un meteorito se desactiva la bala, se destruye el meteorito y por último se invoca el método IncreaseScore.

-Contador: el método IncreaseScore aumenta la puntuación SCORE en uno y llama a UpdateScoreText.

Texto: el método UpdateScoreText actualiza el texto de contador de puntos con el nuevo SCORE.

-Interaccion

Las interacciones de este juego son; movimiento de la nave al que le están asignado las teclas up,down,rigth,left .El disparo de las balas que se realiza con el espacio.

El hecho de que al salir de la pantalla y reaparecer por el otro extremo porta una gran sensación de espacio acotado con movimiento infinito.

La interacción con el canvas es el siguiente: hemos creado un botón de pause, también un panel el cual contiene el menú de pausa con otros tres botones que realizan las funciones de restart, resume y cerrar el juego. Esta interacción la puedes realizar tanto con el ratón para pulsar los botones y la tecla escape para parar y reanudar el juego.