

Microjuego Asteroids - Game Design Document

1. REPOSITORIO

<https://github.com/lsu64/Asteroidsv1>

2. INTRODUCCIÓN

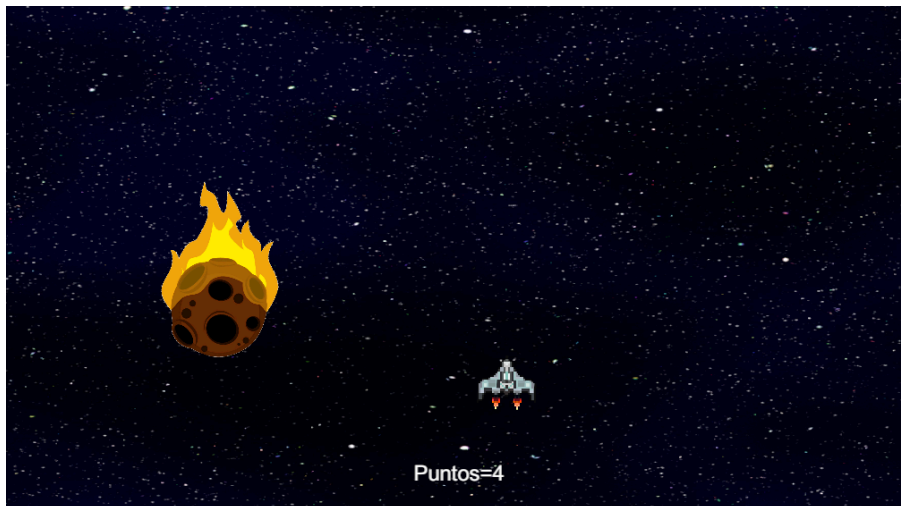
Este videojuego es una réplica del conocido “Asteroids” de la compañía Atari, lanzado en 1979. El concepto es sencillo, el jugador controla una nave que debe destruir los asteroides que aparecen para acumular puntos. Al mismo tiempo, también debe esquivarlos para evitar que su nave sea destruida.

3. ESCENAS

El videojuego consta una única escena:

3.1. Escena Principal

Esta pantalla está compuesta por los elementos del juego (nave, asteroides, balas, etc), además del background y la puntuación. También, en cualquier momento, se puede transicionar al menú de pausa pulsando la tecla P. Se puede reanudar el juego pulsando la tecla P de nuevo.



4. ASSETS Y GAME OBJECTS

En el desarrollo del microjuego tipo Asteroids en Unity, se han utilizado diversos Assets y Game Objects que conforman la estructura y funcionalidad del juego. Estos son:

4.1. Game Objects

4.1.1. Nave Espacial

Representa la unidad controlada por el jugador. Incluye componentes de movimiento, rotación y disparo de balas.

4.1.2. Asteroide

Obstáculos que se desplazan aleatoriamente por la pantalla. Cada asteroide está asociado a un Prefab.

4.1.3. Bala

Proyectiles disparados por la nave espacial para destruir asteroides. También implementados como Prefabs.

4.1.4. Canvas

Texto que muestra la puntuación del jugador.

4.1.5. Space background

Imagen de fondo que simula el espacio exterior, ubicada en la carpeta Sprites. Proporciona el ambiente visual del juego.

4.1.6. Main Camera

Cámara principal que captura la escena del juego.

4.1.7. Directional Light y Event System

Componentes nativos de Unity que manejan la iluminación global y los eventos del sistema, respectivamente, sin modificaciones personalizadas.

4.2. Assets

La carpeta Assets cuenta con:

4.2.1. Los prefabs de la bala y el asteroide.

4.2.2. Diversos scripts en C# que se detallarán más adelante.

4.2.3. Los Sprites de la nave, el asteroide, la bala y el background

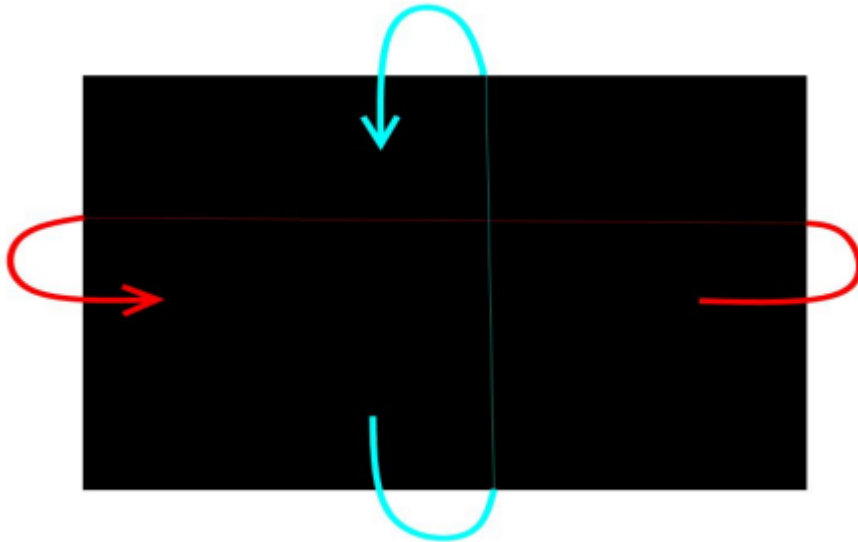
4.2.4. La escena principal.

5. MECÁNICAS

El jugador controla una nave que puede rotar hacia izquierda y derecha, además de desplazarse hacia delante o hacia detrás en el sentido en que está direccionada. La

nave no se detiene de inmediato al dejar de acelerar. En lugar de frenar bruscamente, su velocidad en la dirección en la que se desplazaba va reduciéndose de manera gradual. Una de sus habilidades principales es el disparo. Al pulsar el botón de disparo, se lanza una bala que se mueve en línea recta en la dirección apuntada por la nave hasta que impacta con un asteroide o sale del mapa.

El mapa tiene una particularidad. Al atravesar uno de sus límites, se aparece por el límite opuesto de la forma descrita en la siguiente imagen:



En el espacio hay también asteroides que destruyen la nave al impactar con ella. La trayectoria de los asteroides es siempre en línea recta hacia abajo y a una velocidad constante. Además, solo puede haber un asteroide en escena al mismo tiempo.

Al destruir un asteroide, se obtiene un punto. En la pantalla de juego puede verse en todo momento la puntuación actual.

6. SCRIPTS

Hay cuatro scripts diseñados para implementar la funcionalidad del juego:

6.1. Nave espacial (ShipBehaviour.cs)

Este script controla el comportamiento de la nave espacial. Maneja el movimiento inercial usando fuerzas para acelerar y rotar la nave según las entradas del jugador. Además, permite disparar balas desde un cañón cuando se presiona la barra espaciadora. El script también detecta colisiones: si la nave choca con un asteroide, se reinicia la escena. También ajusta la posición de la nave para que reaparezca al cruzar los límites de la pantalla.

6.2. Balas (BulletMovement.cs)

Este script controla el movimiento y comportamiento de las balas disparadas por la nave. Desplaza la bala en la dirección dada por la nave a una velocidad constante y destruye el objeto tras un tiempo máximo. Si la bala colisiona con un asteroide, aumenta la puntuación, destruye el enemigo y la bala. También actualiza el texto de la puntuación en la interfaz de usuario.

6.3. Asteroides (EnemySpawn.cs)

Este script genera meteoritos a intervalos regulares. La frecuencia de aparición de los meteoritos aumenta con el tiempo gracias a un incremento gradual de la tasa de generación. Los meteoritos aparecen en posiciones aleatorias dentro de los límites definidos por el mapa, y se destruyen automáticamente después de un tiempo máximo.

6.4. Pausa (PauseGame.cs)

Este script gestiona la pausa y reinicio del juego. Al presionar la tecla **P**, el juego se pausa o reanuda alternadamente. Además, al presionar **R**, el juego recarga la escena actual, reiniciando el estado del juego.

7. INTERACCIÓN

En el juego, el jugador controla una nave espacial con movimiento inercial. Utiliza las **teclas de dirección** o **WASD** para rotar la nave y acelerar en la dirección deseada. Al presionar la **barra espaciadora**, dispara balas para destruir meteoritos que aparecen aleatoriamente en la pantalla. Además, el jugador puede pausar o reanudar el juego con la tecla **P** y reiniciar la partida en cualquier momento con la tecla **R**.