

Asteroids

Índice

Identidad del Juego:	3
Pilares del Diseño:	3
Resumen de Género/Historia/Mecánicas:	3
Características:	3
Interfaz:	5
Estilo de Arte:	5

Repositorio de github: <https://github.com/iboigues/Asteroids>

Release de github: <https://github.com/iboigues/Asteroids/releases/tag/1.0.0>

Identidad del Juego:

Destruye asteroides con el arma de tu nave. Esquiva los asteroides y dispara con precisión para sobrevivir y acumula puntos

Pilares del Diseño:

Mecánica de juego clásica y adictiva

El juego mantiene la esencia del clásico Asteroids, donde el jugador controla una nave espacial para destruir asteroides, con nuevas mecánicas nuevas

Dificultad escalable

La dificultad del juego va escalando conforme pasa el tiempo presentando mayores retos para el jugador.

Sistema de puntuación motivador

El juego presenta una puntuación que motiva la competitividad para superar tus récords o los de tus amigos.

Resumen de Género/Historia/Mecánicas:

Género:

Arcade clásico de supervivencia en el espacio.

Historia:

El enfoque del juego es puramente arcade por lo tanto carece de historia.

Mecánicas:

El juego consiste en evitar morir aplastado por un asteroide y destruir la mayor cantidad de asteroides con cuidado ya que al destruir un asteroide se crean 2 más pequeños y con el tiempo llegan más asteroides. Para ello el jugador cuenta con un arma que dispara balas capaces de destruir los asteroides y puede moverse por todo el espacio disponible dentro de la pantalla

Características:

Escenas:

El juego solo tiene una única escena donde se desarrolla toda la acción y tiene: el sprite del espacio, el jugador

Assets y GameObjects:

Prefabs:

- Bala: Tiene la lógica de colisión de asteroides, haciendo que las balas y los asteroides se desactiven. Aumenta la puntuación del jugador y actualiza el texto
- Asteroides grande y pequeño: Tienen la misma lógica diferenciándolos por una variable booleana indicando si es pequeño. Si no es pequeño al colisionar con una bala se divide en dos asteroides pequeños, que se les añade una fuerza en las direcciones abajo izquierda y abajo derecha, y desactivan el asteroide grande.

GameObjects:

- Jugador: Tiene la lógica de movimiento, con desplazamiento y rotación. También puede disparar el prefab de la bala. Si colisiona con un asteroide muere y si se va fuera de los límites del mapa es reposicionado. El jugador tiene un pequeño tiempo de enfriamiento por cada vez que dispara una bala
- Canvas: Este GameObject tiene dos hijos
 - UI: Contiene un campo de texto con la puntuación y un botón de pausa
 - Pause: Contiene un hijo Panel con dos hijos que son dos botones para reanudar la partida o salirse de Asteroids. A su vez este GameObject es el que contiene la lógica que pausa el juego y lo reanuda.

Los otros GameObjects son gestores de Scripts y se explicarán más adelante

Scripts

Los scripts del proyecto son el del jugador, de los asteroides, del menú de pausa y de la bala, cuya lógica ha sido explicada previamente.

ObjectPool:

- Este script es un Singleton. Desde el editor puedes definir dentro de una lista de 'Pool', clase privada dentro del script definida por tres parámetros: un string 'tag', un GameObject 'prefab' y un int 'size'.
- Cuando se activa este GameObject cicla por la lista de pools inicializando una cola para guardar los GameObjects e inicializando tantos GameObjects como se hayan indicado en el parámetro size y el GameObject que se inicializa es el prefab, después se insertan en un diccionario con la llave siendo el parámetro tag y el valor siendo la cola.
- También está suscrito a un evento que se invoca cada vez que se cambia de escena haciendo que se vuelva a inicializar las pools.
- Tiene dos métodos que devuelven un objeto de la pool, el primero devuelve el GameObject y el segundo devuelve un componente del GameObject solicitado, los parámetros de entrada de ambos son la tag del objeto solicitado, la posición y rotación que queremos que use.

EnemySpawner:

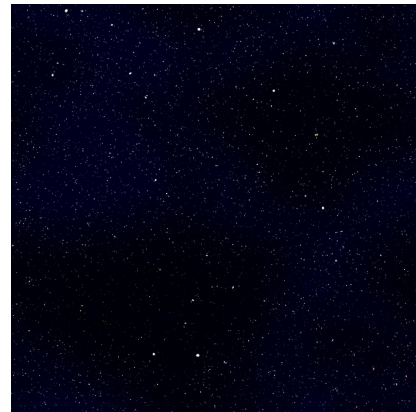
- El EnemySpawner tiene una lógica simple. Donde cada intervalo de tiempo, definido por la variable `_spawnRate` que son el número de meteoritos que aparecen por minuto.
- Cada vez que aparece un asteroide se calcula en qué momento tiene que aparecer el siguiente, se aumenta el ratio de aparición se calcula un número aleatorio para el eje x y se instancia el meteorito.

Interfaz:

Asteroids tiene dos métodos de entrada:

- Teclado: Con las teclas WASD puedes mover y rotar a tu jugador, pulsando la tecla espacio disparas balas y pulsando la tecla escape abres o cierras el menú de pausa.
- Ratón: Con el ratón puedes pulsar los botones que están en la interfaz, como el botón de pausa, el botón de “resume” o el botón de “quit” que sirven para reanudar la partida o para salir del juego respectivamente.

Estilo de Arte:



Estos son los sprites de la nave, del asteroide y del espacio que han sido entregados por el profesor de la asignatura