

GDD del Miniproyecto

Oscar Gonzalez Strozzi

Abstract—Este documento recopila todo el desarrollo, implementación y diseño del miniproyecto para la asignatura. El juego trata de una nave espacial que tiene que ir esquivando meteoritos que van cayendo. El jugador deberá moverse por el espacio con un movimiento de 'tanque' y esquivar o disparar a los meteoritos sin morir.

El objetivo es llegar a la máxima puntuación posible.

I. INTRODUCCIÓN

El miniproyecto está enfocado en aprender a manejar y crear videojuegos en el entorno de **Unity** y familiarizarnos con las herramientas que usaremos.

En la sección 2 se explica el diseño del juego. En la sección 3 se detalla la (en este caso) simple y única escena que nos encontramos en este juego.

En las secciones 4 y 5 se detallan las mecánicas y los scripts utilizados. Por último, en la sección 6 se explica la jugabilidad.

II. DISEÑO

El diseño del videojuego es bastante sencillo.

Cuando inicias el juego, te sitúas en el espacio con una nave. Esta nave es la que el jugador deberá manejar para esquivar o disparar los meteoritos y obtener la máxima puntuación.

III. ESCENAS

El juego solo tiene una escena muy sencilla. En el centro de la pantalla tenemos la **nave** que va a manejar el jugador. Esta emite un sonido al disparar.

Por otro lado, tenemos los **meteoritos**, que van cayendo de manera aleatoria alrededor del jugador. También emiten su propio sonido.

Tenemos un fondo fijo con una textura del espacio. Para darle una apariencia de mundo infinito, el fondo sigue al jugador y se aplica un **offset a la textura** según la velocidad del jugador. Esto crea un efecto de movimiento del espacio, pero realmente solo tenemos una textura que sigue al jugador.

La **UI** es bastante básica, en la esquina inferior tenemos el marcador del jugador. Este aumenta cada vez que derriba un meteorito. En la esquina superior izquierda vemos un pequeño mensaje indicando el botón con el cual se pausa el juego.

Por último, tenemos un OST de fondo para darle más alegría e inmersión.

IV. ASSETS Y GAMEOBJECTS

Para el desarrollo del videojuego, disponemos de cuatro assets.

- **Nave:** el sprite/textura de la nave.
- **Meteorito:** el sprite/textura del meteorito.
- **Bala:** un cuadrado de color rojo que representa una bala.
- **Fondo:** una textura para usar en el fondo.

El asset de **Nave** se usa en un prefab al que llamaremos Player. Este contiene todos los scripts y componentes necesarios para su correcto funcionamiento.

Deberá tener un componente de **RigidBody** y un **Collider** para poder interactuar con el impacto de los meteoritos. También contendrá un **Script** al que le podremos cambiar atributos como la velocidad, el tipo de bala, desde donde sale la bala,...

A su vez, necesitara saber la cámara(para que esta la siga) con rotación fija y el spawner de meteoritos. Debemos incluirlo ya que queremos que el spawn de meteoritos siga al jugador. Lo mismo ocurre con el fondo.

El **fondo** es un GameObject Quad al que se le asigna la textura del fondo. Se asignara el modo de renderizado a repetir, para poder hacer offset infinito sobre la textura.

La aparición de meteoritos se hace mediante un Script al cual se le asigna a un GameObject vacío llamado **MeteorSpawner**.

El GameObject **OST** contendrá un AudioInput con el OST del juego.

Por ultimo tenemos las UI y los menús.

La **UI** por defecto solo incluye la puntuación y el label de tecla de Pausa. Tenemos otros dos menús, uno para la **Pausa** y otro para el **GameOver**.

V. MECÁNICAS

Las mecánicas de este minijuego son muy sencillas y esta basado en el movimiento básico de un tanque

Con la **A** y **D** rotas en la misma posición y con la **W** y **S** te desplazas hacia delante o hacia atrás, respectivamente. Además se pueden disparar balas con la tecla **espacio**.

El jugador tiene la opción de pausar el juego con la letra **P**. Esto abre una UI de Pausa, permitiendo al jugador descansar. Volviendo a pulsar P, se reanuda el juego.

VI. SCRIPTS

Para el desarrollo del videojuego debemos implementar varios Scripts.

Script Player: este Script es el que contiene toda la logica relacionada con la nave y el movimiento del jugador. Aqui se detalla el movimiento, como disparar. Ademas debe tener un contador de puntuación.

Script Background: se encarga de aplicar el offset del fondo para que vaya a la misma velocidad que el jugador.

Script EnemySpawner: Script que se encarga de hacer aparecer a los meteoritos de manera incremental y en posiciones aleatorias relativas al jugador.

Script Bullet: se encarga de la lógica de la bala. Esta debe detectar cuando golpea a un meteorito y destruirlo en su caso. Deberá tener una vida máxima de 3 segundos.

Script MenuController: este Script se encarga de mostrar/ocultar los distintos menús que tenemos en el juego. Debe parar el juego y congelarlo tanto cuando muere el jugador como cuando lo pausa.

VII. JUGABILIDAD

La jugabilidad es bastante intuitiva.

Tratar de sobrevivir a los meteoritos moviéndote con las teclas e ir disparando con el espacio para ganar puntos.

El usuario puede pausar el juego en cualquier momento(P) y reanudarlo.

Si un meteorito le cae y le mata, saldrá una UI de Gameover. El jugador podrá volver a jugar.

VIII. CONCLUSIÓN

El producto final del juego deberá contener todos los aspectos detallados en este documento previamente descritos.

El juego debe ser liviano y fácil de usar. Al arrancarlo deberá empezar inmediatamente la partida con la música a un nivel ´ de sonido moderado.

Se puede obtener el **código fuente** en el siguiente repositorio: <https://github.com/strozz1/minigame>

También se puede jugar a través de **Unity Play**: <https://play.unity.com/en/games/10bd5b8c-5471-480e-82bb-720683a56110/minigame-for-class>