



Tarea 1b

Reporte de Documentación

Curso: CC3501-1

Integrante: Sebastián Salinas

Profesor: Daniel Calderón

Auxiliares: Alonso Utreras

Nelson Marambio Q.

Ayudantes: Beatriz Grabolosa M.

Heinich Porro Sufan

Nadia Decar

Tomás Calderón R.

Fecha: 9 de Mayo de 2020

1. Breve descripción del problema y Arquitectura del Programa

Para esta tarea yo elegí realizar la tarea1b, en la cual se pedía crear un programa que simulara el clásico juego Space Invaders, en el cual eres el piloto de una nave de ataque que participa en una guerra espacial, la idea es eliminar a todos las naves enemigas que se aproximen a tu nave, la cual contiene un cañón de largo alcance que dispara rayos laser.

Para implementar este programa se ocupó como base cuatro de los modulos presentados por el profesor, estos son `basic_shapes.py`, `easy_shaders.py`, `scene_graph.py` y `transformations.py`, además se crearon cuatro modulos aparte que manejaban ciertos aspectos del juego.

`Fondo.py`, en el cual se creó la función `createTextureBackground()` que mediante texturas y nodos creaba la escena del fondo del juego, en la cual se puede apreciar el espacio, el sol y el planeta tierra.

`class_naves.py` y `class_bullet.py` que son modulos en los que se crearon clases que manejaban las características de las naves y las balas respectivamente, como por ejemplo las vidas de las naves o la velocidad de las balas, entre otras. Además de también mediante texturas y nodos crear estos gráficos de escena que se ocuparían posteriormente para dibujar los objetos.

Por último el modulo `hitmarker.py`, que gracias a la ecuación de la distancia entre dos puntos retornaba `True` si es que esta distancia era menor que 0,075 y `False` si no, todo esto con el objetivo de ver si es que existía la colisión entre los objetos bala y nave que se estuvieran dibujando en la escena, para proseguir (en el caso de que el boolean fuera `True`) eliminando estos objetos debido al impacto de la bala sobre la nave.

Es así que con todos estos módulos se procedió a implementar la función principal del programa llamada `space-wars.py`, la cual maneja toda la lógica del juego. Este módulo sigue el patron de diseño Modelo-Vista-Controlador, el controlador maneja la mecánica de movimiento y disparo de nuestra nave. En el modelo se crea el fondo, la animación de victoria y de "Gameover", además se crean los objetos nave, enemigos y balas, y se definen las interacciones entre ellos, como su posición en la escena, su movimiento, su tiempo de disparo, etc. Y en la vista se procede a dibujar estos objetos, donde la mayoría de las veces estos varían mediante animaciones que dependen del tiempo transcurrido al ejecutar el programa.

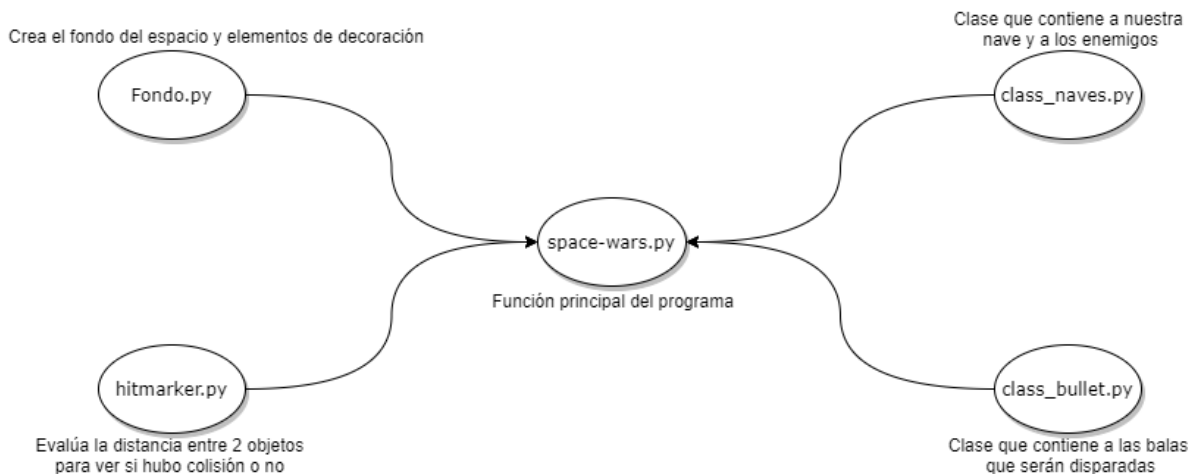


Figura 1: Diagrama.

2. Instrucciones de ejecución

Para iniciar el programa se pide un parametro N , el cual será la cantidad de enemigos que se crearan a lo largo del programa. El programa comenzará con máximo 3 enemigos en la escena, si es que estos son eliminados mediante nuestros disparos y quedan enemigos por dibujar, se hará una especie de respawn y el enemigo volverá a su posición inicial y se trasladará a la escena dando la impresión de que es un nuevo enemigo.

Cabe mencionar que el movimiento de nuestra nave se maneja al presionar las teclas A,S,D,W. A (para moverse a la izquierda), S (para atrás), D (para la derecha) y W (para adelante). Y que el disparo de nuestra nave se genera al presionar la tecla espacio.

3. Resultados



Figura 2: Entrada Inicial.



Figura 3: Escena 1.

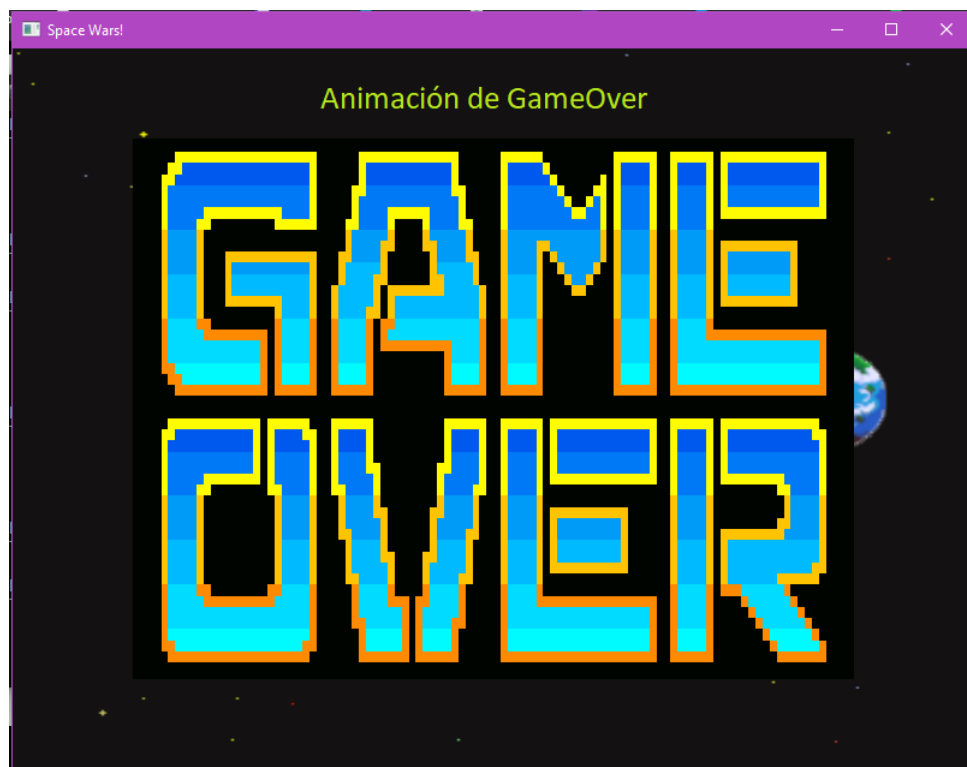


Figura 4: Game Over.