

Projet de fin de semestre : Shooter

1 Introduction

J'ai développé un jeu 2D intitulé Shooter. Il s'agit d'un jeu de type arcade dans lequel le joueur doit

survivre face à des vagues successives de monstres et tenter de réaliser le meilleur score possible.

L'objectif principal de ce projet est d'appliquer des notions de programmation bas niveau, de gestion

des ressources, et de conception modulaire via la SDL.

2 Installation et Pré-requis

Pour faire fonctionner le jeu, les éléments suivants sont nécessaires :

- La bibliothèque SDL2 (ainsi que SDL_image pour le chargement des textures),
- Le Makefile fourni permet de compiler le projet facilement via la commande make.

Un alias a également été ajouté pour regrouper les commandes de compilation, d'exécution et de nettoyage des fichiers intermédiaires.

3 Présentation du Jeu

Le gameplay est simple et accessible : le joueur contrôle un personnage qui doit éliminer des monstres

apparaissant en boucle à l'écran. La partie prend fin lorsque le joueur est tué ou choisit de quitter.

Le jeu est ponctué par un événement spécial : la pluie de comètes, qui survient après un certain temps

de jeu pour augmenter la difficulté.

Le score est basé sur le nombre de monstres éliminés, chacun apportant un certain nombre de points selon son type.

Type de monstre :

1. Mummy

- Moins de vie par rapport a Alien 100 hp
- vitesse aleatoire entre 1 et 7
- moin faible que alien attaque 0,3
- Plus nombreux que alien précisément 3

2. Alien

- Plus fort que Mummy 250 hp
- vitesse de 1
- attaque 0,5
- une seul apparison

Les comets :

- Apparisson aleatoire
- inflige des dégât si tombe sur le joueur
- augment un peut la difficulté du jeu

4 organisation du projet

Projet/

|--- assets/

|---src/

|--- main.c

|--- game.c

|--- game.h

|--- player.c

|--- player.h

```
|--- monster.c  
|--- monster.h  
|--- comet.c  
|--- comet.h  
|--- projectile.c  
|--- projectile.h
```

5 bugs et TODO

Bugs

- Le jeu ne contient pas de methodes pour sauvegarder le meilleur score du joueur
- Le joueur à des moment ne collide pas avec les monstres donc il rentre dans le tas

Todo

- Faire les animations
- Changer peut-être le fonctionnement du jeu en une vague successives de monstres qui seront de plus en plus fort

6) Remerciements

Ce projet a été inspiré d'une série de tutoriels vidéos réalisés par Graven Développement sur

YouTube, initialement proposés avec Pygame (Python). J'avais déjà réalisé une première version de ce jeu en Python, et ce projet m'a permis de reproduire et adapter l'ensemble en langage C avec SDL,

une bibliothèque beaucoup plus bas niveau.

7 Ressources

- Zeste de Savoir - Tutoriel "Utiliser la SDL en langage C" Pour une introduction rapide à la SDL.
- Site officiel de SDL - <https://www.libsdl.org/> Documentation et guide d'installation.
- Chaîne YouTube de Graven - Développement de jeux en Python avec Pygame Source d'inspiration initiale.
- ChatGPT - Assistance à la correction et à l'ajout des commentaire