|  |
| --- |
| DivTec☺ |
| Création d’un jeu en C++ |
| Rapport de projet |

|  |
| --- |
| Nussbaumer Thibaud  20/11/2019 |

Table des matières

[1 Introduction 2](#_Toc26184674)

[2 Biographie 2](#_Toc26184675)

[3 But et contexte 2](#_Toc26184676)

[4 Glossaire 2](#_Toc26184677)

[5 Étapes du projet 3](#_Toc26184678)

[5.1 Choix du projet 3](#_Toc26184679)

[5.2 Bullet Hell 3](#_Toc26184680)

[5.3 GameFramework 4](#_Toc26184681)

[5.4 Conception de la scène 4](#_Toc26184682)

[5.5 Conception du style du jeu 4](#_Toc26184683)

[5.6 Conception du joueur 4](#_Toc26184684)

[5.7 Conception de l’ennemi 4](#_Toc26184685)

[5.8 4](#_Toc26184686)

[6 État du projet 4](#_Toc26184687)

[7 Améliorations possibles 5](#_Toc26184688)

[8 Ce que j’ai appris 5](#_Toc26184689)

[9 Conclusion 5](#_Toc26184690)

# Introduction

Ce document a pour but de retranscrire le déroulement du projet, les difficultés que j’ai rencontrées, comment ces dernières ont été résolues ainsi que mon ressenti.

# Biographie

Je m’appelle Thibaud Nussbaumer, j’ai 19 ans et j’habite à Alle. L’informatique m’intéresse depuis tout petit quand je regardais mon père travailler dans son bureau chez nous. Je suis venu à l’EMT car j’avais comme conviction de travailler dans l’industrie du jeu-vidéo plus tard mais cela à changer depuis car je ne m’y intéresse plus.

# But et contexte

Ce projet fait suite à celui des portes ouvertes, le but étant de créer un jeu 2d (dans le langage que l’on veut) d’une période allant du 30 septembre jusqu’au 21 décembre.

Ce projet nous permet de mettre en pratique nos connaissances ainsi que notre manière de se débrouiller.

# Glossaire

* **C++** : est un langage de programmation permettant la programmation sous de multiples paradigmes. Dans le cas de ce projet c’est de la programmation orientée-objets. C’est un langage libre de droit, donc tout le monde peut l’utiliser gratuitement. (Source [wikipédia](https://fr.wikipedia.org/wiki/C%2B%2B))
* **Qt**: est un IDE (Environnement de développement intégré) qui rassemble différents outils permettant d’améliorer la productivité des programmeurs. (Source [wikipédia](https://fr.wikipedia.org/wiki/Qt))
* **Sprite**: est une image que l’on peut déplacer sur l’écran. Il peut être animé en chargeant d’autres images les unes après les autres. (Source [wikipédia](https://fr.wikipedia.org/wiki/Sprite_(jeu_vidéo)))
* **Shoot ’em up**: est un genre de jeu vidéo dérivé du jeu d'action dans lequel le joueur dirige un véhicule ou un personnage devant détruire un grand nombre d'ennemis à l'aide d'armes de plus en plus puissantes, au fur et à mesure des niveaux, tout en esquivant leurs projectiles pour rester en vie. (Source [wikipédia](https://fr.wikipedia.org/wiki/Shoot_%27em_up))
* **Bullet Hell**: est un sous genre du Shoot ‘em up où un grand nombre de projectiles apparaissent à l’écran et constitue la difficulté de ce genre de jeu. (Source [wikipédia](https://fr.wikipedia.org/wiki/Shoot_%27em_up))

# Étapes du projet

## Choix du projet

Notre enseignant nous a donné les différentes directives à suivre pour le projet et nous a laissé le choix du langage de programmation. Ce-dernier devait être orienté objet et inclure une documentation automatique. Le choix du type de jeu nous était en quelque sorte libre dans le sens où c’était possible de le réaliser avant la fin de ce bloc d’atelier. J’ai donc choisi une mécanique de jeu « simple » en voulant faire un Bullet Hell. C’est un jeu où l’on doit affronter des ennemis qui tirent beaucoup de projectile de plusieurs façons.



Figure Image du jeu Touhou Project. Jeu type "Bullet Hell" très connu

J’ai décidé de développé ce jeu en C++ avec la bibliothèque GameFramework car il me semble un peu plus simple à faire, les plus grandes difficultés seront la gestion des ennemis et des balles d’après moi.

Je ne compte pas atteindre un niveau très élevé de Bullet Hell, je compte d’abord pouvoir gérer minimum 2 types d’ennemis avec différents types de projectiles pour avoir quelque chose de potable à rendre à la fin.

## Bullet Hell

Le « Bullet Hell », « manic shooter » ou « danmaku » en japonais, est un dérivé du shoot ‘em up. Il se caractérise par un grand nombre de balles sur l’écran du jeu. Ce style de jeu est apparu vers le milieu des année 1990 avec comme précurseur du style « Batsugun » sorti en 1994. Le plus connu d’entre eux reste « Touhou Project » sorti en 1996 et développé par un japonais surnommé « ZUN » ancien employé de Taito (entreprise ayant développé Space Invaders).

## GameFramework

J’utilise l’IDE Qt Creator pour ce projet car c’est celui que je connais le plus. Je reprendrai la bibliothèque GameFramework du professeur parce que je ne suis pas très doué en programmation et cette dernière m’aidera dans le développement de mon jeu.

## Conception de la scène

Pour respecter le style des « Bullet Hell » j’ai décidé de faire une fenêtre plus grande verticalement qu’horizontalement, cette dernière mesure 720px sur 900px. Elle n’est pas redimensionnable. La scène est un peu plus petite et mesure 700px sur 880px.

## Conception du style du jeu

J’ai décidé de faire une sorte de jeu de mot avec le genre du jeu en appelant mon jeu « BulletHell ». Le style graphique suit la même logique, j’ai décidé de faire un design représentant les enfers.

## Conception du joueur

Pour le joueur j’ai décidé de faire simple en reprenant une SpriteSheet déjà existante d’une sorte de gargouille.



Image contenant les Sprites du joueur

Ces sprites sont tirés d’un jeu de la Super Nintendo.

## Conception de l’ennemi

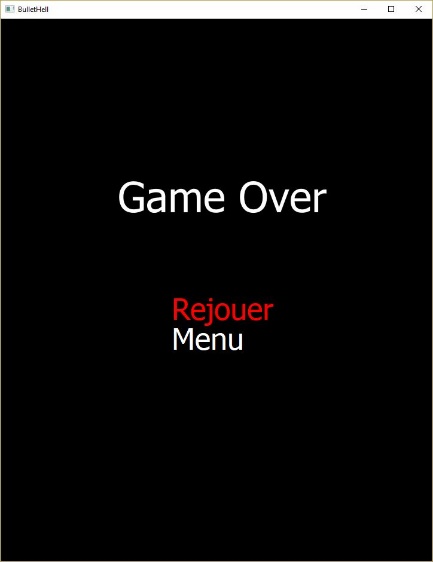
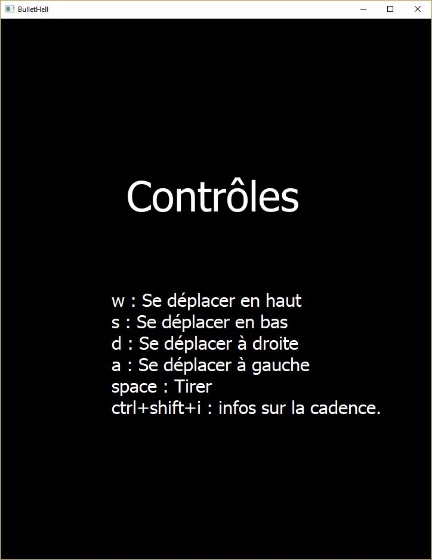
Comme dans tous les jeux vidéo, il existe un ennemi de base tout bête avec des déplacements simplifiés. J’ai donc décidé d’en faire un aussi afin de me faire la main sur la bibliothèque du professeur et j’ai créé cet ennemi.

Il se déplace de gauche à droite, change de direction lorsqu’il percute les limites de la scène et tir toutes les secondes.

Sprite de l'ennemi

## Conception des menus

Pour les menus Jonas m’a aidé avec son code, par la suite je les ai adaptés à mes besoins et j’ai dû modifier le code pour que ça fonctionne.



Il n’y a pas besoin d’explications pour les menus car les options de ceux-ci sont assez intuitives et explicites.

# État du projet

# Améliorations possibles

# Ce que j’ai appris

# Conclusion