

Projet 5 – AMIMI Ahmed

Sommaire :

- Le jeu :
 - Prérequis
 - Lancement du jeu
 - Comment jouer ?
 - Présentation globale du jeu
- Fonctionnement et principe du jeu :
 - Introduction
 - Idée générale du Ray casting
 - Le Ray casting
 - Déplacement du joueur
- Création de Map (carte) :
 - Présentation
 - Utilisation

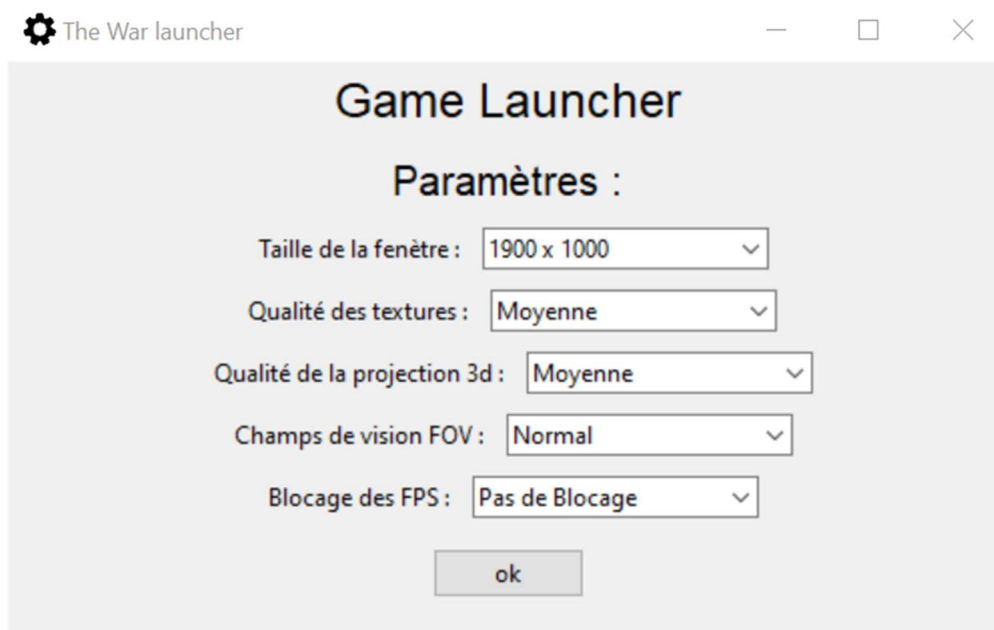
Le Jeu :

Prérequis :

Pour pouvoir jouer au jeu il faut avoir installer une version récente de python 3, ainsi que les bibliothèques requises au bon fonctionnement du jeu disponible en cliquant seulement sur le .bat fournis. Il faut être sur *windows* pour pouvoir lancer l'exécutable *main.exe*, sinon cliquer sur le fichier *main.pyw* depuis les fichiers source du jeu, sinon ouvrir le fichier source *Game_SourceCode* depuis visual studio code et lancer *main.pyw*.

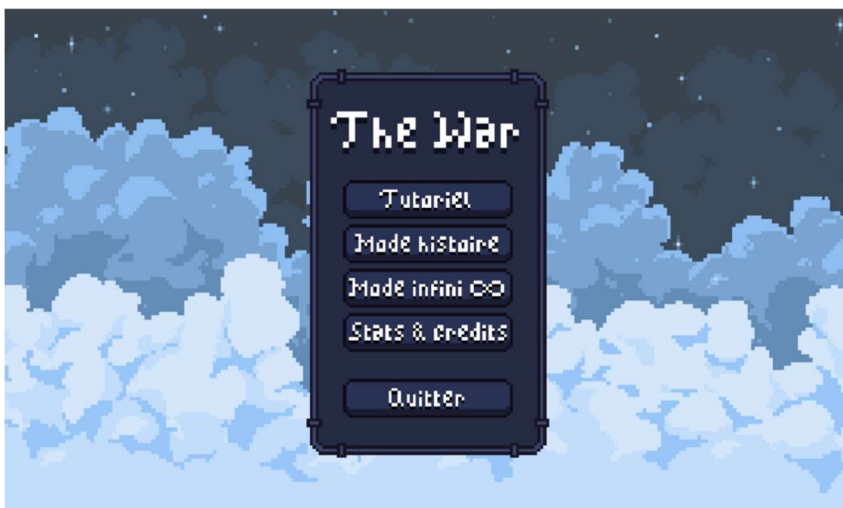
Lancement du Jeu :

Après avoir lancer le jeu, la fenêtre de lancement devrais s'afficher



Choisir les différents paramètres pour le lancement du jeu, un lancement optimal serait avec les paramètre de textures et de projection 3d sur *Moyenne*, un champs de vision *normal* et un blocage des fps a *60*.

Après avoir appuyé sur ok le menu se lance :



Plusieurs choix s'offrent à vous, le **tutoriel** qui apprend les bases du jeu, le mode **histoire** composé d'un seul niveau pour l'instant, **3 sauvegardes** seulement sont disponibles pour ce mode, la **possibilité de supprimer une sauvegarde n'est pas encore implémentée** (le seul moyen de supprimer les données de jeu c'est par l'outil développeur *Game_SourceCode\tools\cleardata.pyw*), Le **mode infini** qui se joue à l'infini avec des mondes et des positions de monstres aléatoires, (deux niveaux sont disponibles pour l'instant), ici c'est le score qui compte, le **meilleur score** que vous aurez fait, est stocké dans le dernier menu, ou les **statistiques** du joueur sont affichées.

Comment jouer ?

Tout est expliqué dans le tutoriel, mais, reprenons :



Présentation globale du jeu :

Le jeu se présente comme un first person shooter (fps) où l'on incarne un soldat qui se téléporte de place en place et qui élimine les aliens qui ont envahi le monde et les différents espaces. Traversez différentes cartes originales dans un style retro pixel art. L'histoire est encore en développement, les futurs niveaux du mode histoire viendront plus tard au fil des mises à jours.

Fonctionnement et principe du jeu :

Introduction :

Le Raycasting est une technique de rendu permettant de créer une perspective 3D depuis une carte 2D. À l'époque où les ordinateurs étaient plus lents, il n'était pas possible de faire fonctionner de vrais moteurs 3D en temps réel, et le raycasting était la première solution. Le Raycasting peut aller très vite (sur les autres langages que python), car il suffit de faire un calcul pour chaque ligne verticale de l'écran. Le jeu le plus connu qui utilise cette technique est bien sûr Wolfenstein 3D.

« Wolfenstein 3D est un jeu vidéo de tir à la première personne développé par id Software et publié par Apogee Software le 5 mai 1992 sur MS-DOS. Il s'inspire des jeux Castle Wolfenstein et Beyond Castle Wolfenstein développés par Muse Software dans les années 1980. Le joueur y incarne l'espion B. J. » - Wikipedia



Le moteur de raycasting de Wolfenstein 3D était très limité, lui permettant de fonctionner même sur un iAPX 286 : tous les murs ont la même hauteur et sont des carrés orthogonaux sur une grille 2D, comme on peut le voir sur cette capture d'écran d'un mapeditor pour Wolf3D :

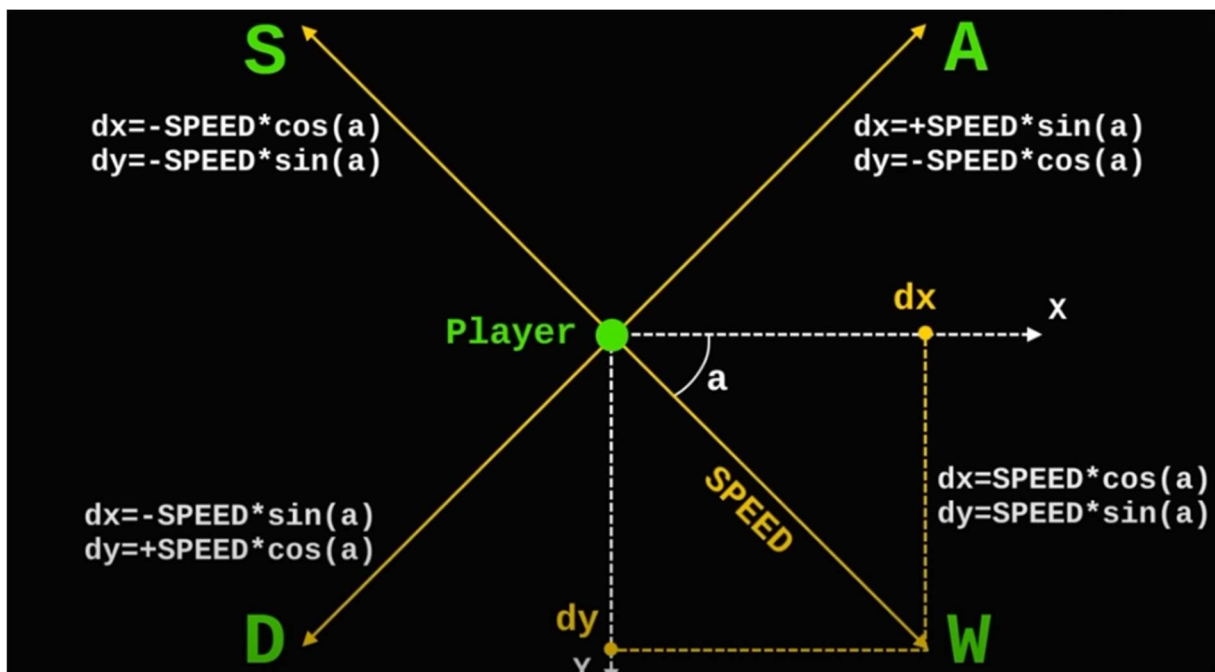


La gestion de hauteur n'est pas gérée par le moteur graphique par exemple comme les escaliers, les sauts ou les différences de hauteur sont impossibles à faire avec ce moteur. Des jeux plus tard, tels que Doom et Duke Nukem 3D utilisaient également le raycasting, mais ce sont des moteurs beaucoup plus avancés qui permettent l'affichage des murs inclinés, des hauteurs différentes, des sols et plafonds texturés, des murs transparents.

En savoir plus : <https://guy-grave.developpez.com/tutoriels/jeux/doom-wolfenstein-raycasting/>

Déplacement du joueur :

Le déplacement du joueur se calcule ainsi :



Création de Map (carte) :

Présentation :

C'est un outil qui permet de dessiner les cartes, il n'est pas ouvert au public du fait de sa complexité de compréhension et ses bug causé par l'interpréteur Tcl de Tkinter et aussi a cause de sa faible protection (il faut savoir ce qu'on fait)

Utilisation :

Mode Texture : on dessine les textures

Mode Room : on créer une room en ayant au préalable sélectionné un ciel et un sol et on choisit la porte avec le clic molette

Mode entité : on dessine les objets et monstre, on peut choisir le point d'apparition du joueur avec double clic molette