

Projet Jeu

Space Raclure



La Providence (2019-2020)

BTS SN2

Brard Adrien / Lhermitte Nathan / Boury Aurélien / Jouen Matthias

SOMMAIRE

[Présentation du jeu 3](#_Toc27388973)

[Les objets techniques utilisés 4](#_Toc27388974)

[Les problèmes rencontrés 5](#_Toc27388975)

[Cahier des charges 6](#_Toc27388976)

[Conclusion 7](#_Toc27388977)

# Présentation du jeu

On a eu pour but de créer un jeu dans l’un de notre projet cette année.

Notre jeu est inspiré du jeu "Flappy Bird", il est codé en Java ; un langage que nous voulions approfondir/découvrir pour élargir notre connaissance.

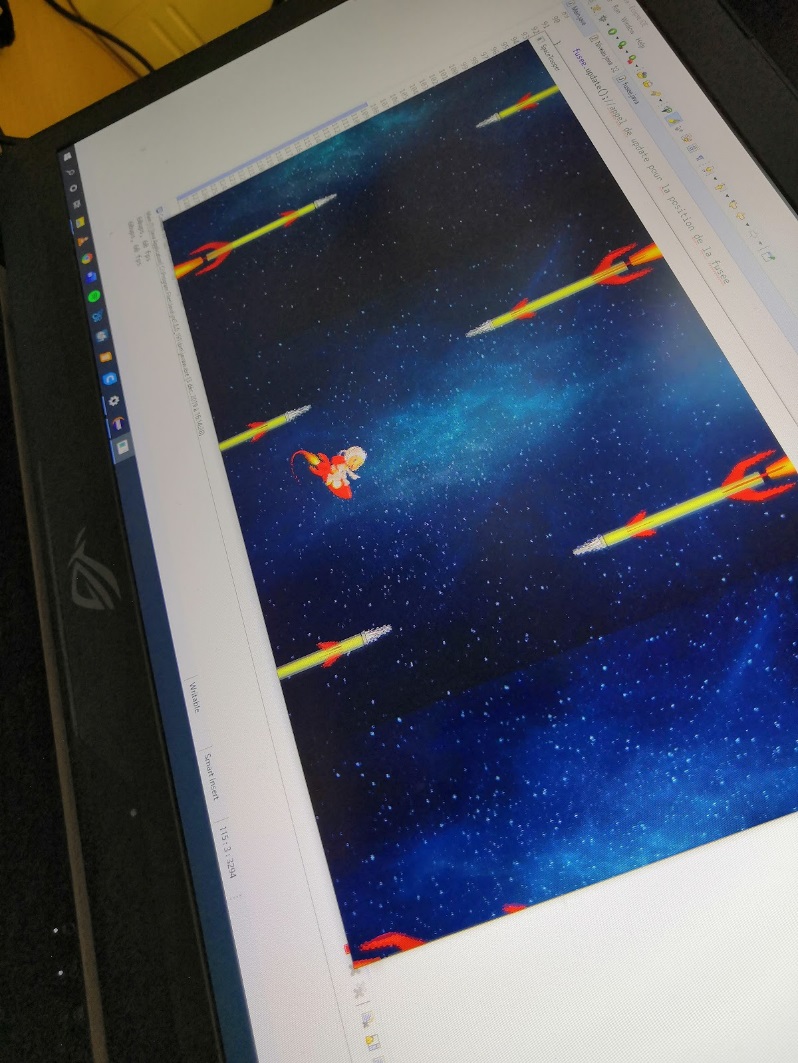
On a utilisé le Java pourquoi ? Car il nous fallait un langage adapté pour créer un jeu assez simple et en peu de temps. Nous avions donc le choix entre le C++ et le Java.

Nous avons choisi le Java pour plusieurs raisons :

* C’est un langage assez simple à comprendre et qui s’approche de ce qu’on connait déjà
* C’est un langage populaire c’est à dire que nous avons pu nous documenter un peu partout sur internet pour résoudre certains soucis.
* L’utilisation des bibliothèques facilite la tâche notamment pour des fonctions spécifiques au jeu.

Nous avons voulu reprendre ce jeu connu de tous en essayant de l'améliorer à notre manière grâce à des bonus multiples (malus ou bonus).

Ce "Flappy Bird" nouvelle génération se situe dans l'espace et vous incarnez un astronaute qui se déplace à travers des "missiles".



# Les objets techniques utilisés

Nous avons utilisé plusieurs bibliothèques pour ce projet :

* **LWJGL**
* **OpenGL**
* **GLFW**

La première bibliothèque que nous allons vous présenter est **LWJGL** acronyme de Lightweight Java Game Library, LWJGL est une bibliothèque open source multiplate-forme gérant l’image (avec OpenGl), l’audio (avec OpenAl) et le calcul parallèle (avec OpenCl). Elle est conçue pour réaliser des frameworks de jeu. Dans les avantages que nous propose LWJGL, nous avons notamment utiliser OpenGl et GLFW.

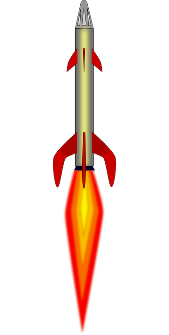
**OpenGL** (***Open******G****raphics* ***L****ibrary*) :  un ensemble normalisé de fonctions de calcul d’images 2D ou 3D. Cette interface de programmation est disponible sur de nombreuses plateformes.

OpenGL permet à un programme de déclarer la géométrie d'objets sous forme de points, de vecteurs, de polygones, de bitmaps et de textures. OpenGL effectue ensuite des calculs de projection en vue de déterminer l'image à l'écran, en tenant compte de la distance, de l'orientation, des ombres, de la transparence et du cadrage.

**GLFW** : c’est une bibliothèque offrant la possibilité de créer notamment des fenêtres applicatives avec OpenGl ou alors de recevoir des informations sur les touches appuyées par exemple.

On a aussi utilisé Photoshop qui n’est pas directement acteur du code, mais qui nous a servi à faire les textures comme le fond, les obstacles et le personnage / joueur.

**Photoshop** qu’est-ce que c’est ? C’est un logiciel professionnel de graphisme qui offre des possibilités quasiment infinies. Ce logiciel est utilisé par un grand nombre de professionnels : architectes, dessinateurs, illustrateurs, cartographes... Il permet de retravailler la photo, de réaliser un photomontage etc.… dans le but de professionnaliser la création graphique.



# Les problèmes rencontrés

# Cahier de test

|  |  |  |  |
| --- | --- | --- | --- |
| N° | Action | Description | Résultat (OK/NOK) |
| 1 | Lancer le jeu | Le jeu démarre |  |
| 2 | Le jeu doit être fluide | Jeu en 50 FPS / IMS min |  |
| 3 | Le joueur doit pouvoir se déplacer | Fonctionnement des touches (ESPACE) |  |
| 4 | Le joueur peut mourir | Au toucher de l’obstacle, le joueur meurt via les collisions |  |
| 5 | Réapparition | Reviens au début du jeu |  |
| 6 | Prendre un bonus | Ex : Passer à travers les obstacles |  |
| 7 | Prendre un malus | Ex : Les obstacles deviennent plus gros |  |
| 8 | Fermer le jeu | Il se ferme sans problème |  |

# Conclusion