



Comprendre le monde,
construire l'avenir®

UNIVERSITE PARIS-SUD

Master Informatique 1ere Année

Année 2016-2017

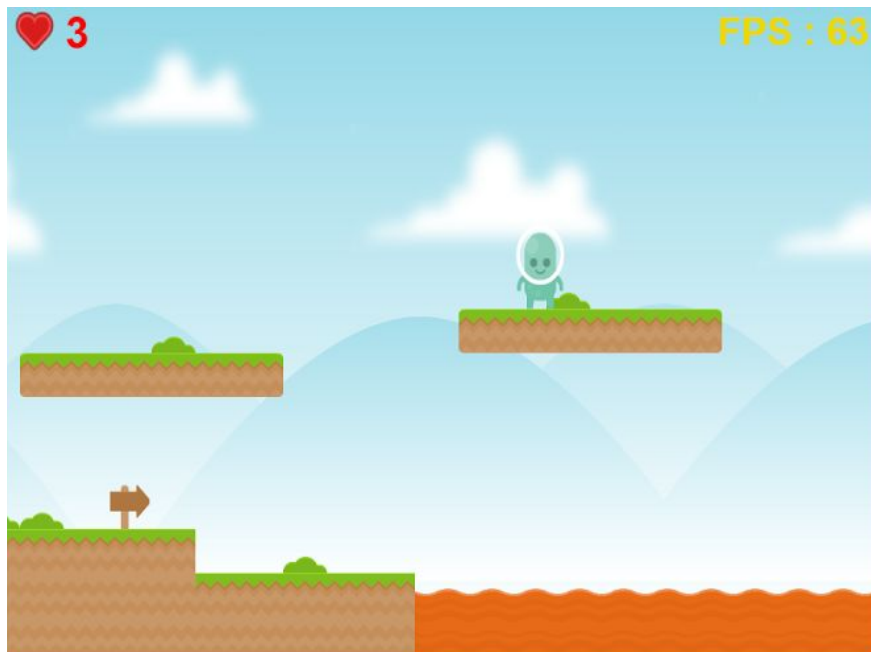
Projet de programmation Javascript

Rapport de programmation web avancée

par

Rémi BUTET
DU LI

Enseignant: Kim.Nguyen, *Université Paris-Sud*



Capture d'écran du jeu

Introduction

Dans le cadre de l'unité d'enseignement de programmation objet avancée de département informatique de l'université Paris Sud, il a été demandé aux étudiants de Master 1, d'effectuer un jeu de plates-formes en Javascript. Le projet était à réaliser en binôme.

L'objectif du projet était de réaliser un jeu de plates-formes similaire à Mario. Dans un tel jeu, un personnage, contrôlé par le joueur au moyen du clavier évolue dans un monde en deux dimensions, vu de profil. Le but du jeu est de parcourir un niveau en évitant une suite d'obstacles pour atteindre la sortie.

I. Les fonctionnalités implémentées

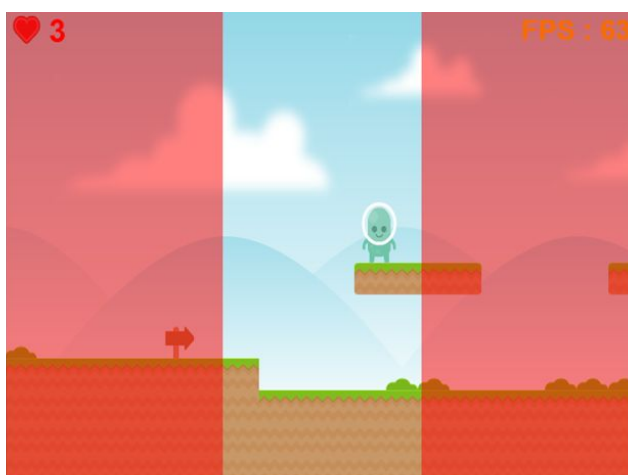
Pour notre projet avons décidé d'implémenter plusieurs fonctionnalités supplémentaires proposées dans le sujet et quelque fonctionnalité supplémentaire pour améliorer la qualité du jeu.

La première fonctionnalité étant un "défilement de l'arrière-plan" qui permet de donner un effets de profondeur au jeu. Pour cela nous avons superposé 2 images et nous les faisons défiler à une vitesse différente.

La seconde fonctionnalité est le "jeu complet", cela permet au joueur d'avoir plusieurs vies et de le faire jouer sur plusieurs mondes. Nous avons aussi mis en place un menu d'accueil qui lui permet de patienter pendant les temps chargements. Une fois que joueur a fini un niveau il passe directement sur le monde suivant. Il dispose d'un certain nombre de vies avant de perdre.

La troisième fonctionnalité est la "vitesse de course", qui permet au joueur de courir et pouvoir faire des sauts plus grands. Pour cela le joueur doit appuyer simultanément sur la touche CTRL et la touche de direction.

Nous avons de plus implémenté de nombreux autres petits fonctionnalités qui permettent de rendre le jeu plus fluide et plus attractif. Comme par exemple les panneaux et les buissons qui sont là uniquement pour faire un effet de profondeur et pour guider le joueur. Pour aussi optimiser le jeu nous avons décidé de créer 2 zones qui permettent de faire défiler le monde dans la bonne direction et une troisième qui permet au joueur de se déplacer dedans sans à avoir à réactualiser tout le décor :



Les différentes zones

II. L'organisation du programme

Lorsque nous avons commencé à développer le programme, nous nous sommes mis d'accord sur les informations qui devaient se trouver le fichier du monde. Nous avons décidé de rendre le jeu au maximum personnalisable, c'est pour cela qu'il y a beaucoup d'informations dans le fichier du monde. Concernant le joueur, il est possible de choisir sa position de départ, sa taille et ainsi qu'une liste de textures selon les situations. Pour le background, il est possible d'indiquer une liste d'images à superposer et leurs vitesses de déplacement. Pour les blocs, il est possible d'en créer un nombre infini et de leur définir des caractéristiques personnalisées (la texture, le damage, la collision...).

Pour les variables qui devaient être identique dans tous les mondes, nous avons créé un fichier "constants.js" qui les regroupe ainsi que les messages et les touches de directions. Cela permet de corriger et modifier simplement le fonctionnement du jeu.

Pour réaliser le déplacement du joueur, nous avons réutilisé une partie du code que nous avons utilisé en TP. Il nous a permis de gérer la vélocité du joueur et les collisions avec les blocs. Mais nous avons dû l'adapter pour pouvoir courir et optimiser les performances.

III. Les difficultés rencontrées

Pendant le développement du projet nous avons rencontré quelques difficultés mais nous avons réussi à tous les surmontés.

La première difficulté que nous avons rencontrée fut l'adaptation du code vu en TP. Le joueur pouvait se déplacer dans le monde mais il pouvait arriver qu'il perde sa vélocité latérale. Pour corriger cela nous avons dû implémenter une fonction qui vérifie s'il est en collision avec le sol, avant de lui faire perdre sa vélocité.

Nous avons ensuite rencontré des problèmes lors de la gestion de plusieurs touches enfoncées simultanément. Nous avons dû mettre en place un historique des touches enfoncées et réaliser nos actions en fonction de celle-ci. De plus initialement notre jeu utilisait les touches Z, Q, S, D mais la majorité des personnes qui ont testé notre jeu se sont plaint de ne pas pouvoir utiliser les touches directionnelles.

La dernière grande difficulté que nous avons rencontré fut la gestion des messages. Il y a de nombreux types de messages (Chargement, Mort, Niveau suivant...). Il pouvait arriver que plusieurs arrivent en même temps donc il a fallu que l'on gère tous les cas et que l'on affiche unique le message le plus important.

Conclusion

Nous avons trouvé le projet très instructif et très intéressant mais pour des raisons de temps, nous n'avons pas pu implémenter toutes les fonctionnalités que nous souhaitons. Donc il se peut qu'il y ait quelque variable non utilisé dans notre code.

Ce projet nous a permis d'apprendre à utiliser les dernières fonctionnalités de Javascript comme les Class et les Promise. Ce qui nous permet désormais de programmer beaucoup plus simplement en Javascript.

Mais ce que nous retenons le plus de ce projet c'est qu'il faut savoir faire un compromis entre les fonctionnalités que l'on peut implémenter tout en garantissant une bonne fluidité de jeu.