|  |
| --- |
| DIVTEC |
| Rapport de fin de travail |
| Jeu de tir en javascript |

|  |
| --- |
| Brosy Arsène  28/03/2023 |

# Le résultat

## Description

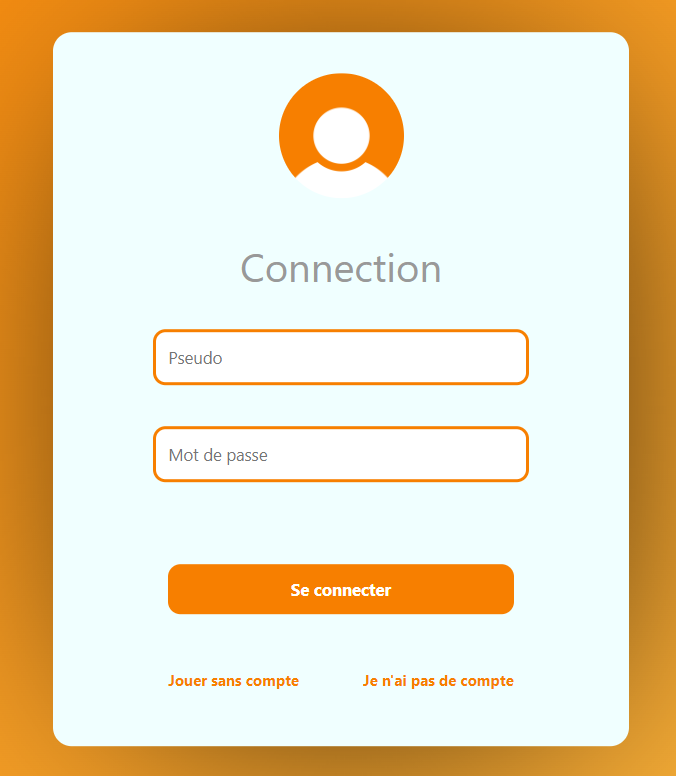
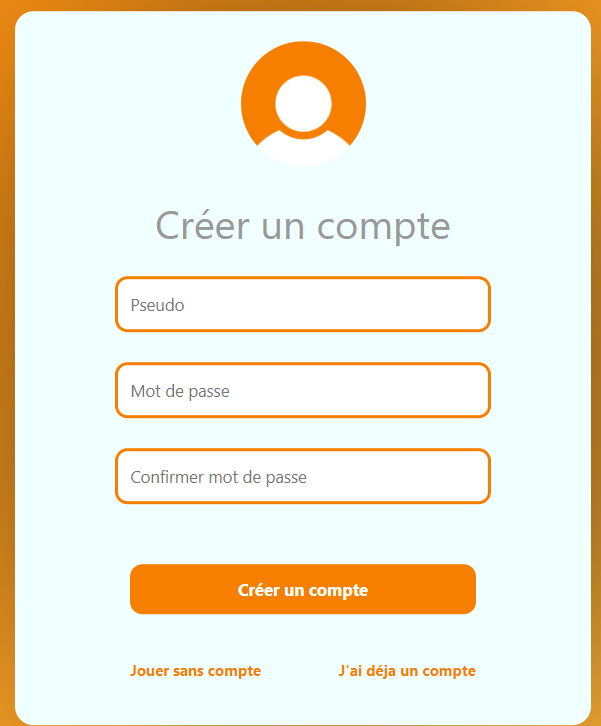
Le projet complet est un jeu jouable en multijoueur sur lequel on peut se connecter et incarner une araignée (on ne reconnait pas que c’est une araignée étant donné que je n’ai rien dessiné) et grimper sur les murs ou le toit puis tirer sur les autres joueurs connectés qui ont chacun 5 vies afin de les éliminer tout en essayant de survivre.

## Images

Une image contenant graphique

Description générée automatiquement

On voit ici notre joueur (en rouge) et les autres joueurs (en bleu)

Menu de connexion Menu de création de compte

## Dates

Création du projet : 30 janvier 2023

Fin de création : 28 mars 2023

# Technologies utilisées

## Canvas HTML

Le canvas est une balise HTML qui permet, via javascript, d’afficher des images, formes, etc. Je l’ai donc tout naturellement utilisée pour afficher mon jeu car cela me permet de calculer une image et des l’afficher en continu pour créer un flux vidéo.

## Javascript

Javascript est un langage de programmation utilisé par les navigateurs pour manipuler les éléments HTML d’une page. J’ai donc écrit un programme qui s’occupe de lire les inputs au clavier et à la souris de l’utilisateur pour calculer les physiques du jeu et ensuite d’afficher le jeu sur le canvas.

## Firebase

Firebase est une API qui met à disposition des petites bases de données gratuites. Le terme « base de données » n’est peut-être pas le plus adapté car il s’agit juste d’un fichier JSON synchronisé en temps réel. Je l’utilise donc pour pouvoir lui envoyer mes coordonnées et lire les coordonnées des autres joueurs afin d’interagir avec eux.

# Problèmes rencontrés

## Tourner le joueur

Le joueur doit tourner sur lui-même pour s’adapter à la surface et comme j’ai décidé tout calculer moi-même je n’avais que la possibilité de tourner le joueur sur son centre alors que dans mon cas j’avais besoin de le tourner sur son coin.

Solution :

Une image contenant texte, signe

Description générée automatiquement

J’ai donc créé une fonction qui tourne d’abord le joueur (il est donc dans la position bleu) puis, à l’aide de fonctions trigonométriques, calcule le vecteur ici en jaune et déplace le joueur pour obtenir la position finale (verte) et donc pouvoir tourer le joueur sur son coin

## Synchroniser le multijoueur

Au début cela pouvait sembler compliqué, mais en effectuant des recherches j’ai découvert la librairie Firebase qui me permet de créer des bases de données en temps réel qui m’ont donc permis des synchroniser les positions des joueurs.

Une image contenant texte

Description générée automatiquement

Coordonnées x, y et orientation d’un joueur (player12981)

## Synchroniser les balles

J’ai eu beaucoup de mal à synchroniser en même temps les positions et les balles.

Solution :

J’ai séparer les donnés en 2 fichiers (players et bullets).