

---

# *Rapport de Stage*

---



BERRICHI Yassine

BTS SIO OPTION SLAM

LYCÉE TURGOT

# INTRODUCTION

Dans le cadre de ma formation en BTS SIO , option SLAM (Solutions Logicielles et Applications Métiers), j'ai effectué un stage de développement web au sein d'une entreprise commençant le 15 Mai 2023 qui s'est conclu le 7 Juillet 2023 , pour une durée totale de 8 semaines.

J'ai effectué ce stage au sein de l'entreprise HDM Network ASBL , en intégrant leur équipe et leur pôle développement web.

Durant ce stage, j'ai pu être assigné à plusieurs petites missions, qui m'ont permis d'accroître mes compétences en développement web et régulièrement pratiquer différentes technologies.

Réussir à compléter chacune de ses missions me donnerait alors ma première expérience professionnelle ainsi que de renforcer mon niveau en programmation web , ma compréhension du code ainsi que les fondamentaux que j'ai acquis durant toute cette première année de BTS SIO .

# HDM NETWORK ASBL

HDM Network ASBL, basée à Charleroi en Belgique, a été fondée par Quentin Mousset et David Van Goidtsnoven. Cette entreprise se spécialise dans la proposition de formations de qualité visant à développer les compétences professionnelles dans les divers domaines du numérique. Grâce à son engagement en faveur de l'excellence, HDM Network offre une expérience de formation enrichissante.

L'équipe de HDM Network compte entre 50 et 60 professionnels travaillant dans différents pôles clés. Parmi ces pôles, on retrouve les RH spécialisées dans le recrutement et le placement, ainsi que des experts en Web Development, Web Design et graphisme (illustrateurs). De plus, l'entreprise dispose d'une équipe dédiée aux partenariats B2B et d'une équipe spécialisée en SEO (Optimisation pour les moteurs de recherche).

Un aspect essentiel de l'activité de HDM Network est de mettre en relation ces différents pôles avec des entreprises intéressées à travailler avec elles sur différentes missions, en fonction des besoins spécifiques de chaque domaine. L'entreprise facilite ainsi la collaboration entre ses experts et les clients potentiels, garantissant une réponse adaptée aux demandes des entreprises en matière de compétences numériques.

Grâce à son approche centrée sur la qualité des formations et à sa capacité à créer des synergies entre les différentes équipes, HDM Network ASBL est reconnue comme un acteur majeur dans le développement des compétences numériques en Belgique.

# MON ROLE DANS HDM

## HORAIRES :

-Lundi à Jeudi : 8h30-12h / 13h-16h30

-Vendredi : 8h30-11h30 / 12h30-16h30

## MES MISSIONS :

Durant ce stage au sein du pôle WebDev de HDM que j'ai pu intégrer pour 8 semaines , voici le travail que j'ai effectué :

-Formations sur de nouvelles technologies liées au développement web

-La réalisation d'application web front-end et back-end

-Le Scraping de données

-La réalisation de site web/refonte

-La réalisation d'API

-La réalisation de jeux vidéos sur navigateur via librairies

## TECHNOLOGIES UTILISÉES :

-HTML

-CSS3

-JavaScript

-NodeJS

-React

# JAVASCRIPT

Durant mon stage au sein de l'entreprise HDM Network, j'ai eu l'opportunité de suivre une formation approfondie en JavaScript. Cette formation a été cruciale pour ma compréhension des principes fondamentaux de ce langage de programmation incontournable dans le développement web.

La formation en JavaScript m'a permis d'acquérir une solide connaissance des bases du langage. J'ai appris à manipuler les variables, à utiliser les fonctions pour encapsuler le code réutilisable, à mettre en place des boucles et des conditions pour contrôler le flux d'exécution, ainsi qu'à manipuler les tableaux et les objets.

Une des principales forces de JavaScript réside dans sa capacité à interagir avec le DOM (Document Object Model). Grâce à ma formation, j'ai acquis une compréhension approfondie de la manipulation du DOM, ce qui m'a permis de créer des interactions dynamiques sur les sites web. J'ai appris à sélectionner des éléments HTML, à modifier leur contenu, leurs attributs et leurs styles, ainsi qu'à réagir aux événements de l'utilisateur tels que les clics, les survols et les saisies clavier.

La formation en JavaScript m'a également permis de comprendre les concepts avancés de la programmation orientée objet (POO) appliqués à JavaScript. J'ai appris à créer des classes, des objets et des méthodes, et à utiliser l'héritage et le polymorphisme pour organiser et réutiliser le code de manière efficace. Cela m'a donné une meilleure compréhension des bonnes pratiques de développement et de la modularité du code.

Un aspect essentiel de la formation a été l'introduction aux bibliothèques et aux Framework JavaScript populaires tels que jQuery.

J'ai appris à utiliser jQuery pour simplifier les manipulations du DOM, effectuer des requêtes AJAX et animer les éléments de la page. Cette connaissance pratique de bibliothèques populaires a renforcé mes compétences en développement web et ma capacité à utiliser des outils éprouvés.

JavaScript c'est un langage polyvalent qui est utilisé aussi bien côté client (frontend) que côté serveur (backend). Au cours de ma formation, j'ai également exploré des concepts avancés tels que les événements asynchrones, les promesses et les fonctions fléchées. Ces connaissances m'ont permis de mieux comprendre et d'utiliser des fonctionnalités modernes de JavaScript.

L'importance de la formation en JavaScript réside dans le fait que ce langage est omniprésent dans le développement web. Il est essentiel pour créer des fonctionnalités interactives, des applications web complexes et des sites web réactifs. En acquérant une maîtrise de JavaScript, j'ai développé une compétence fondamentale recherchée par les employeurs dans le domaine du développement web, ce qui m'arrange donc énormément.

En résumé, la formation en JavaScript m'a fourni une base solide dans la compréhension des concepts et des fonctionnalités clés de ce langage. Grâce à cette formation, je suis en mesure de développer des fonctionnalités dynamiques et interactives sur les sites web, d'améliorer l'expérience utilisateur et de contribuer au développement d'applications web modernes, et donc ainsi maîtriser le langage principal utilisé au sein du pôle WebDev HDM.

# REACT

J'ai également eu l'opportunité de suivre une formation approfondie en React, une bibliothèque JavaScript largement utilisée pour le développement de User Interface (UI) réactives.

La formation en React a été un aspect essentiel de mon stage, car cette bibliothèque est devenue un choix populaire pour la construction d'applications web modernes, surtout dans HDM. Grâce à cette formation, j'ai pu acquérir une compréhension approfondie des concepts fondamentaux de React et maîtriser les compétences nécessaires pour développer des applications web réactives et performantes.

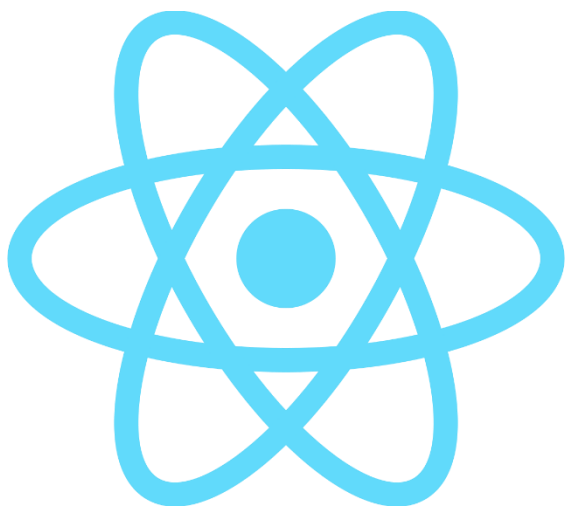
L'un des principaux concepts que j'ai appris dans ma formation en React est la notion de composants. Les composants sont les blocs de construction de toute interface React. J'ai appris à créer et à organiser des composants réutilisables qui peuvent être combinés pour construire des interfaces complexes. Cette approche modulaire facilite la maintenance du code et la réutilisation des éléments d'interface.

Un autre aspect important de la formation a été l'apprentissage du rendu conditionnel et de la gestion des states (états). J'ai appris à rendre dynamiquement certains éléments de l'interface en fonction des conditions définies dans l'application. Grâce à la gestion des états, j'ai été en mesure de mettre à jour les composants de manière réactive en réponse aux interactions de l'utilisateur.

La formation en React m'a également permis de comprendre les cycles de vie des composants. J'ai appris à utiliser les méthodes de cycle de vie de React pour effectuer des actions spécifiques à différentes étapes du rendu et de la mise à jour des composants. Cela m'a donné un meilleur contrôle sur le flux d'exécution de l'application et la possibilité d'effectuer des tâches telles que l'appel à des API externes ou la mise à jour de l'état global de l'application.

Une autre caractéristique importante de React que j'ai explorée dans ma formation est la gestion de l'état global avec le contexte de React (Il est également possible de le faire , en utilisant par exemple des bibliothèques telles que Redux , mais j'ai juste utilisé le contexte). J'ai appris à centraliser l'état partagé de l'application et à le gérer de manière efficace, ce qui facilite la communication entre les différents composants et améliore la cohérence de l'application. Par exemple , le contexte ça aide à pouvoir appliquer un « mode sombre » sur toute l'application.

L'utilisation de React offre de nombreux avantages, notamment la virtualisation du DOM, qui permet de mettre à jour uniquement les parties de l'interface qui ont été modifiées, améliorant ainsi les performances. J'ai également appris à utiliser des outils de développement tels que Jest pour tester mon code petit à petit.



```
PASS src/pages/Results/index.test.js
✓ Ceci est mon premier test (1 ms)

Test Suites: 1 passed, 1 total
Tests:       1 passed, 1 total
Snapshots:   0 total
Time:        0.389 s, estimated 1 s
Ran all test suites related to changed files.
```

Exemple d'usage de la librairie React Testing Library pour les tests :



File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Line #s
All files	28.39	13.33	24.07	28.57	
src	0	100	100	0	
index.jsx	0	100	100	0	14
src/components/Card	0	0	0	0	
index.jsx	0	0	0	0	7-61
src/components/Error	0	0	0	0	
index.jsx	0	0	0	0	6-32
src/components/Footer	100	100	100	100	
index.jsx	100	100	100	100	
src/components/Header	0	0	0	0	
index.jsx	0	0	0	0	8-22
src/pages/Freelances	0	0	0	0	
index.jsx	0	0	0	0	7-62
src/pages/Home	100	50	100	100	
index.jsx	100	50	100	100	15-36
src/pages/Results	36.84	14.29	21.43	36.84	
index.jsx	50	20	30	50	15,19,35,42,74-112
style.jsx	0	0	0	0	4-46
src/pages/Survey	0	0	0	0	
index.jsx	0	0	0	0	10-102
src/utils/context	66.67	50	50	66.67	
index.jsx	66.67	50	50	66.67	21-26
src/utils/hooks	11.76	0	25	12.5	
index.jsx	11.76	0	25	12.5	5-25
src/utils/style	60	50	50	60	
Atoms.jsx	100	75	100	100	27
GlobalStyle.jsx	0	0	0	0	4-19
colors.js	100	100	100	100	

# SEO(Searching Engine Optimisation)

Cette formation s'est avérée essentielle pour comprendre les principes et les stratégies permettant d'améliorer la visibilité et le classement d'un site web dans les résultats des moteurs de recherche.

La formation en SEO , beaucoup plus simple que les autres à assimiler , m'a permis de comprendre l'importance du référencement naturel pour attirer du trafic organique qualifié vers un site web. J'ai appris les techniques fondamentales pour optimiser un site web afin qu'il réponde aux critères des moteurs de recherche et améliore sa visibilité dans les résultats de recherche.

L'un des éléments clés que j'ai explorés dans ma formation en SEO est l'optimisation des mots-clés. J'ai appris à effectuer une recherche approfondie des mots-clés pertinents pour le domaine d'activité d'un site web. Cela m'a permis de comprendre comment sélectionner les mots-clés appropriés et les intégrer de manière stratégique dans le contenu, les titres, les balises et les méta-descriptions pour améliorer la pertinence du site pour les recherches effectuées par les utilisateurs.

La formation en SEO m'a également permis de comprendre l'importance de la structure du site et de l'optimisation technique. J'ai appris à améliorer la structure de navigation du site, à optimiser les URL, les balises HTML et les en-têtes pour faciliter l'indexation par les moteurs de recherche. J'ai également acquis des connaissances sur l'optimisation de la vitesse de chargement du site, l'optimisation des

images et la mise en place de pratiques recommandées pour améliorer l'expérience utilisateur.

Un autre aspect crucial de la formation en SEO a été l'accent mis sur la création de contenu de qualité. J'ai appris à produire du contenu pertinent, informatif et engageant pour les utilisateurs, tout en prenant en compte les critères de pertinence des moteurs de recherche. J'ai compris l'importance de la recherche de mots-clés dans la création de contenu et comment structurer les articles, les pages et les publications de blog pour améliorer leur classement dans les résultats de recherche.

## MES MISSIONS :

Après avoir effectués les formations ci-dessus , j'ai pu enfin être assigné à plusieurs petites missions pour pratiquer le JavaScript :

1. Création d'un petit jeu 2D pour le SpiderWolf de HDM :

J'ai été assigné à la création d'un petit jeu en 2D afin d'ajouter un jeu leur spiderwolf , et également découvrir le Framework Phaser.



Technologies utilisées durant cette mission :

-Javascript

- Framework Phaser

- NodeJs

Logiciel utilisé :

- Visual Studio Code

Le jeu web en 2D que j'ai choisis à développer est inspiré du style de jeu de Vampire Survivor. Un jeu simple pour prendre en main Phaser et pratiquer le JavaScript , voici les tâches que mon code doit exécuté lorsqu'il sera abouti :

- Afficher un bouton pour lancer le jeu

- Afficher le personnage et des ennemis apparaissant autour du joueur de manière croissante

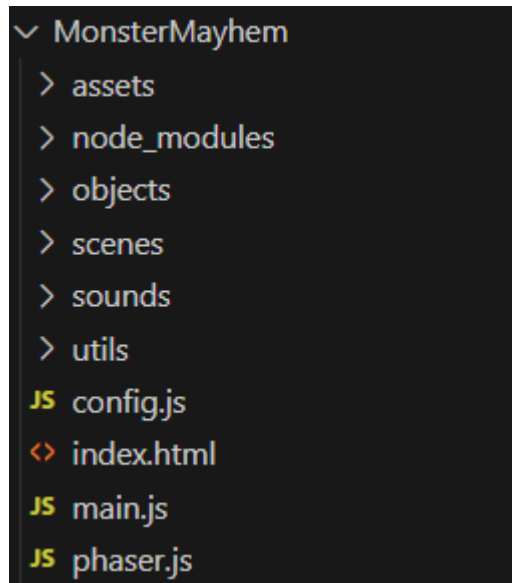
- Incrémenter un score et l'afficher quand on tue des ennemis

- Les ennemis laisseront tombé une « clé » qui lorsque ramassé, incrémente un score de pièces affichée

- Lorsqu'un ennemi touche le joueur , sa barre de points de vie baisse de 10 (Nombre de points de vie max : 100 , ce qui laisse 10 touches).

Pour commencer, j'ai utilisé NodeJS pour initialiser mon projet.

J'ai par la suite télécharger via le site de Phaser leur Framework pour pouvoir directement l'importer dans les fichiers qui auront besoin d'utiliser celui-ci. L'architecture du projet ressemblait donc à celle-ci :

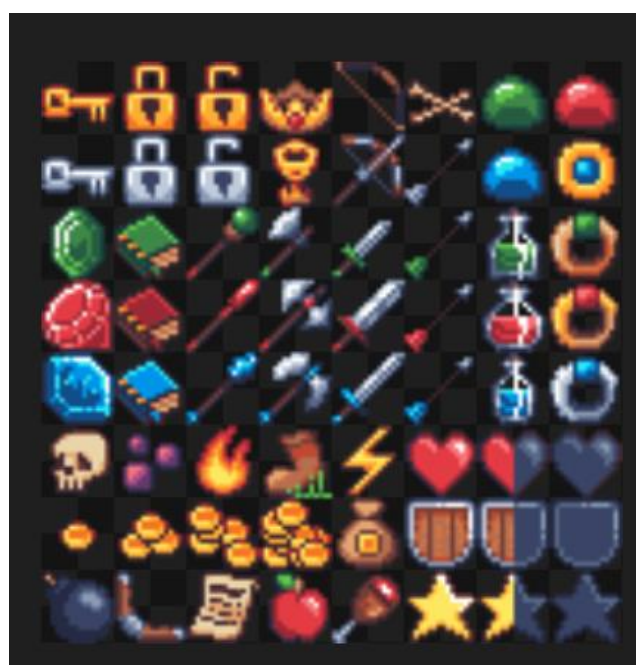
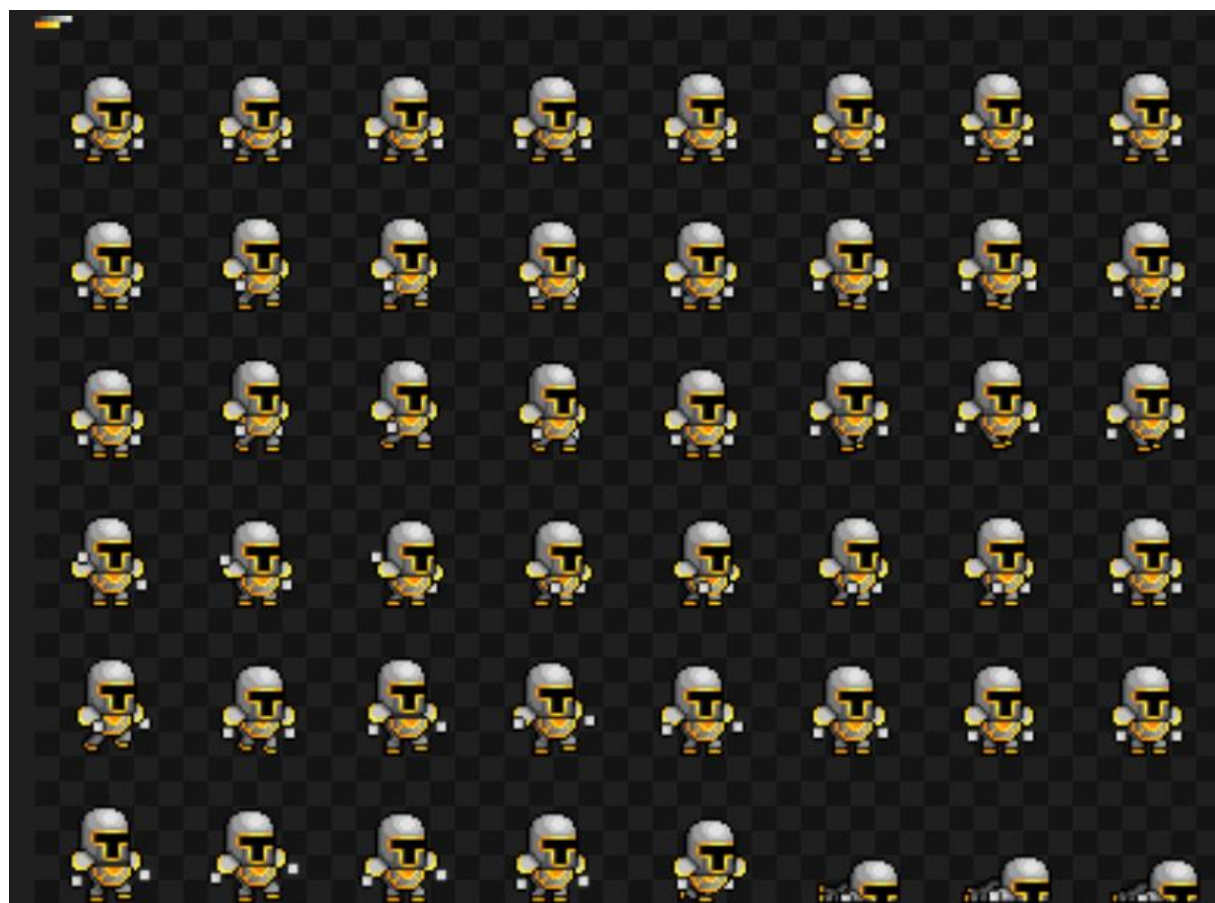


Une fois l'architecture et le cahier des charges fait , je pouvais donc commencer à coder.

Je me suis servi de ressources libres de droits sur des sites proposant des spritesheets 2D libre d'utilisations pour avoir l'ensemble de mes assets.

(Spritesheet = image comportant différentes petites images pour construire un décor ou représenter différentes phase de l'animations d'éléments)

Exemples des ressources utilisées durant ce projet :



Afin de gérer les différentes parties du jeu , j'ai décidé de créer une classe pour chaque partie , et donc par conséquent un fichier par classe comme ceci :

```
JS BootScene.js
JS GameScene.js
JS Item.js
JS main.js
JS Monster.js
JS phaser.js
JS Player.js
```

Le main.js lui , se servait donc de tout les autres fichiers pour mener à bien l'exécution du jeu.

Pour que vous compreniez comment fonctionne un jeu web en javascript sous le framework Phaser , voici un exemple avec des commentaires. Chaque jeu Phaser est composé de cette manière pour fonctionner correctement :

```
1 var config = {
2   type: Phaser.AUTO,
3   width: 1500,
4   height: 800,
5   physics: {
6     default: 'arcade',
7     arcade: {
8       gravity: { y: 0 },
9     },
10  },
11  scene: [BootScene, GameScene], // Utiliser un tableau pour définir les scènes
12 };
13
14 const game = new Phaser.Game(config);
15
16 function preload() {
17   // Préchargement des ressources communes à toutes les scènes (si nécessaire)
18 }
19
20 function create() {
21   this.scene.launch('BootScene'); // Lancer la scène BootScene
22 }
23
24 function update() {
25   // Mise à jour du jeu
26 }
27
28
29 game.scene.start('BootScene'); // Démarrer la scène BootScene
```



Une fois le corps de notre jeu créé , il suffit alors d'implémenter chaque fonctionnalité dans chacune des classes qui sont les suivantes :

Dans la classe Player :

- Le personnage
- Sa barre de vie
- Ses déplacements
- Gérer ses collisions avec l'ennemi , ses tirs , ses clés

Dans la classe Monster :

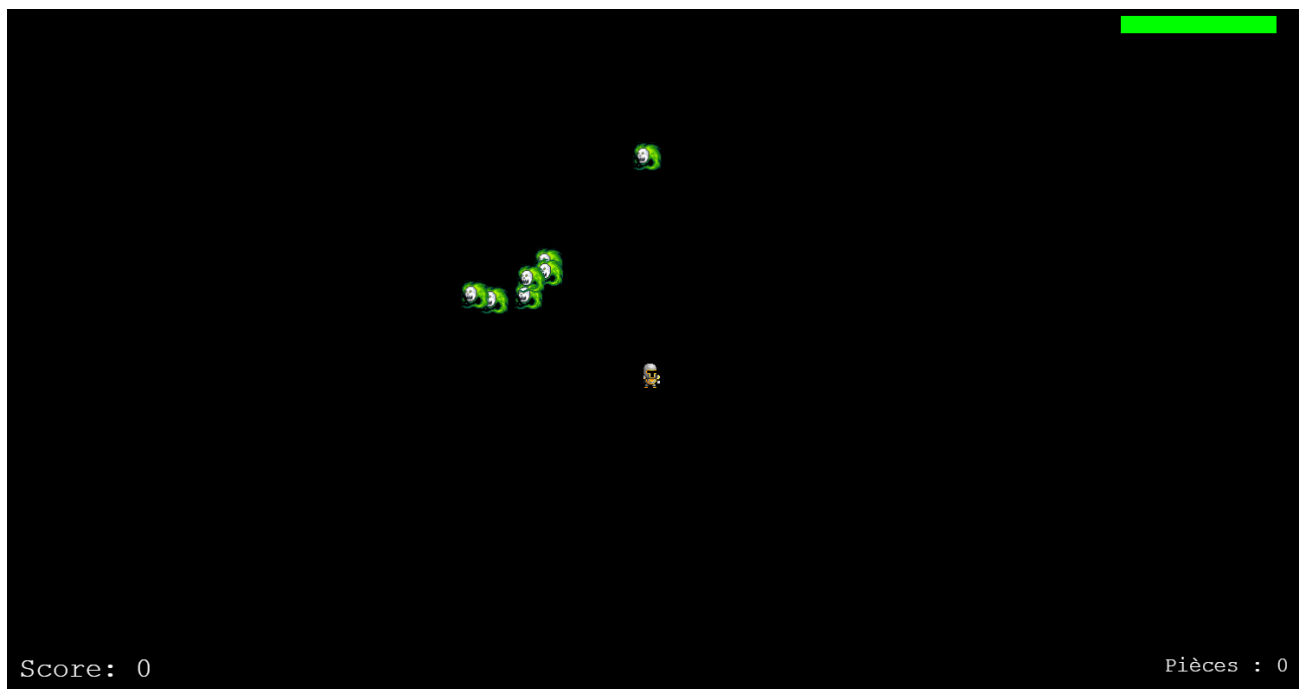
- Gérer les déplacements du monstre
- Gérer son nombre de points de vie
- Son nombre de dégâts qu'il va infliger
- Sa zone d'apparition , ainsi que son nombre
- Le fait qu'il dépose une clé à l'endroit de sa mort
- Le faire disparaître lorsqu'il subit assez de dégâts pour faire descendre ses points de vie à 0 ou moins

Dans la classe Item :

- Gérer l'apparition du bon item lors de la mort du monstre
- Gérer l'incrémentation du score de pièces lors du ramassage de l'item

Dans la classe main :

- Utiliser l'ensemble des fonctionnalités à fin de créer le jeu , et voilà le résultat !



Print des points de vie dans la console pour vérifier que la barre baisse bien ainsi que le nombre de points de vie :

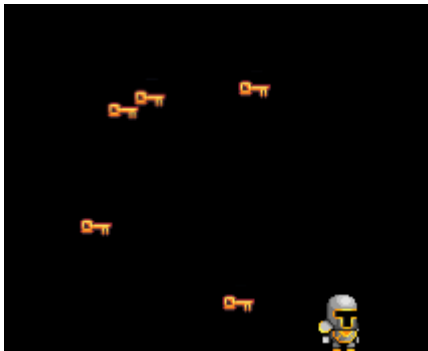
	GameScene.js:65
▶ <i>Player {scene: GameScene, health: 50, numItems: 0, healthBar: Graphics, numItemsText: Text}</i>	
45	Player.js:53
40	Player.js:53
35	Player.js:53
30	Player.js:53
25	Player.js:53
20	Player.js:53
15	Player.js:53
10	Player.js:53
5	Player.js:53
0	Player.js:53
Player died	Player.js:56

Le score et le nombre de pièces augmentent bien :

90

Pièces : 3

Les clés apparaissent avec succès a l'endroit même où l'on élimine le monstre :



DIFFICULTÉS RENCONTRÉES :

- Erreurs fréquentes lors d'ajout de nouvelles fonctionnalités
- Problèmes de compréhensions du framework Phaser au départ
- Manque de temps pour peaufiner le jeu et y ajouter des fonctionnalités conséquentes.

## 2. Refonte de site web

J'ai été amené dans un second temps à effectuer la refonte de la page d'accueil du site web de HDM Network ASBL.

Le défi ici, étant lancé au pôle WebDev était le suivant :

N'utiliser aucun framework , aucune librairie , rien.

Du HTML , CSS et Javascript en native , c'est-à-dire le langage de base sans rien d'ajouter.

De plus , l'objectif ici était de partir from-scratch , c'est-à-dire de partir d'un fichier vide , aucune bases de code rien. Nous devons également impérativement travailler le SEO du site et l'optimiser, c'était le plus important lors de cette mission.

Technologies utilisées :

- HTML

- CSS

- JavaScript

Logiciel utilisé :

- Visual Studio Code

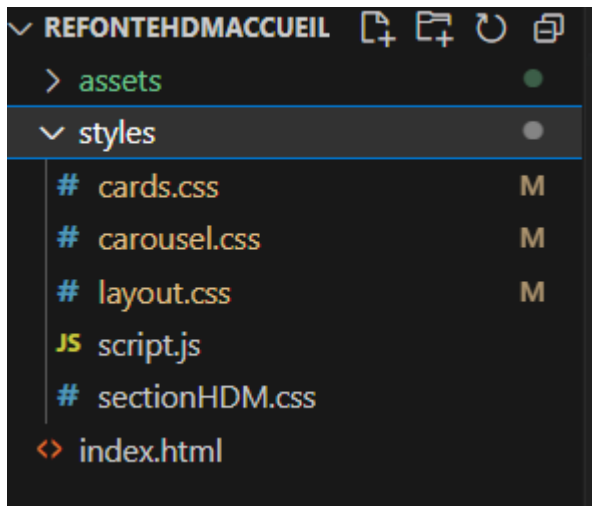
Le principal ici a été de reproduire la page d'accueil de base, mais non à l'identique. Il m'a fallu alors réfléchir et me mettre à la place d'un webdesigner cette fois ci, en utilisant par exemple figma pour faire des templates. Le travail ici était beaucoup plus orienté frontend.

Après réflexions , j'ai opté pour le cahier des charges suivant :

- Respecter la charte graphique blanc/bleu

- Utiliser un carrousel
- Faire des cards qu'on peut ouvrir pour les informations avec des textes trop longs
- Rendre plus moderne le tout en arrondissant les bords

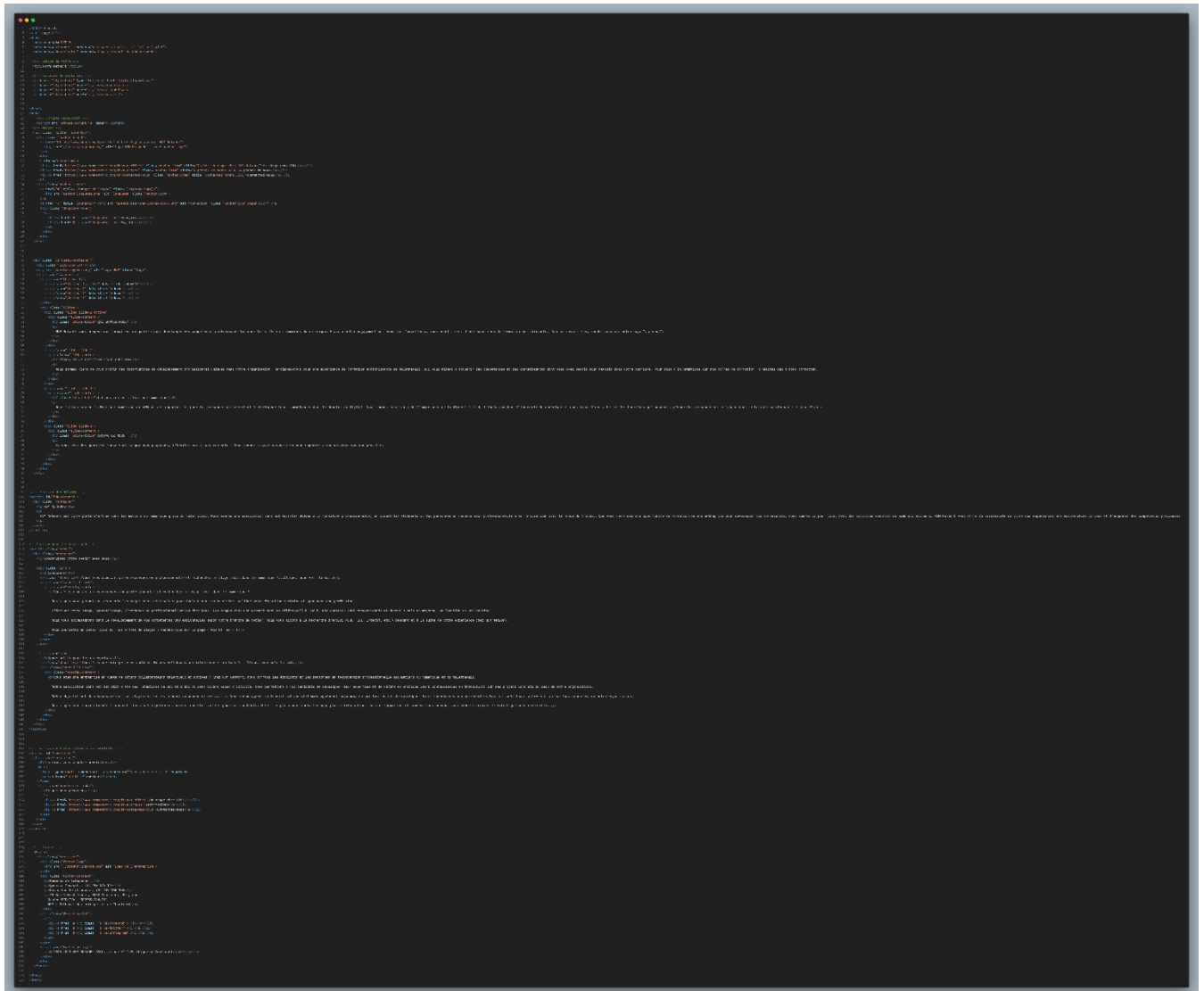
Voici l'architecture du code :



J'ai utilisé trois fichiers css distincts , avec dans le layout le principal du site tel que la navbar et les couleurs de la charte graphique , puis un fichier respectif pour les cards et le carrousel. Le fichier sectionHDM.css lui fait référence à une bande bleue contenant un texte, que je ne savais pas comment styliser alors j'ai fais fichier à part comme pour les autres parties du site, le considérant alors lui aussi comme une partie distincte de cette page d'accueil. Le script.js lui contiendra le code nécessaire aux animations relatives au carrousel, et diverses animations d'apparition/disparition de certains éléments du site, l'agrandissement des cards pour y afficher un texte complet etc.

Concernant le code , on y retrouvera du html basique , contenant chacune des informations de la page d'accueil originale , entourées de div pour styliser le tout.

(Oui , le code est assez conséquent alors j'utilise l'extension de VSCODE « codeSnap » pour prendre une capture d'écran du code sélectionné automatiquement depuis le logiciel)



Le plus intéressant durant cette mission a été l'optimisation du SEO :  
C'est ici que j'ai pu apprendre pour la première fois à optimiser ceci

pour un site web , et c'est une pratique très utile pour le référencement de celui-ci .

Par exemple , ici :

- structurer l'url pour que ce soit lisible , avec des mots clés pertinents.
- Utiliser des balise title pour décrire le contenu de la page avec des mots clés importants
- Des balises header (h1 h2 h3 etc) pour structurer le contenu de manière hiérarchique
- Balises alt pour les images pour décrire le contenu de celles-ci pour permettre au moteur de recherche de comprendre le contenu visuel et améliorer son accessibilité

Voici le résultat :

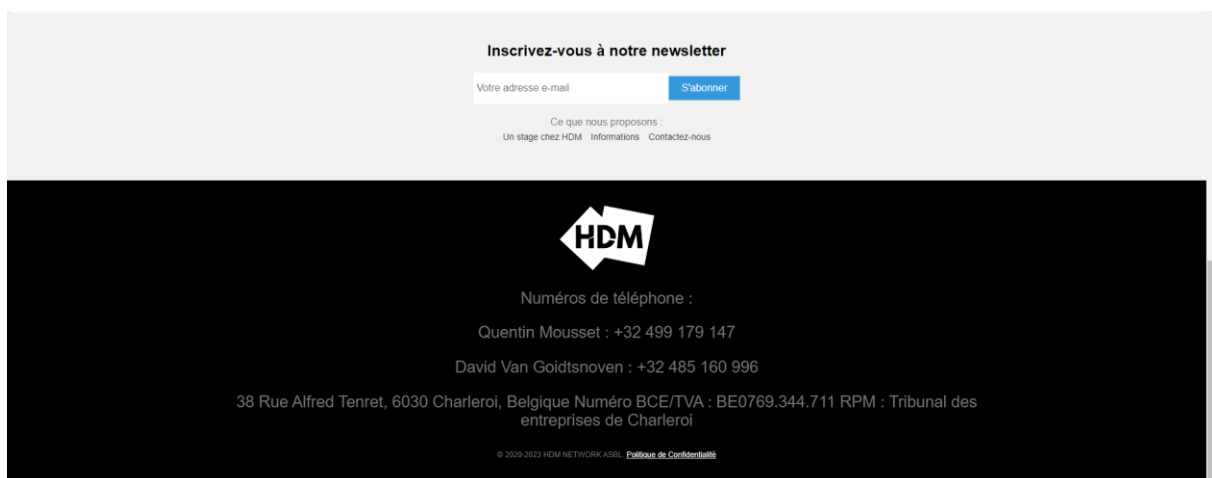




Lorsque l'on clique sur une card , par exemple celle appelée « Candidats » :

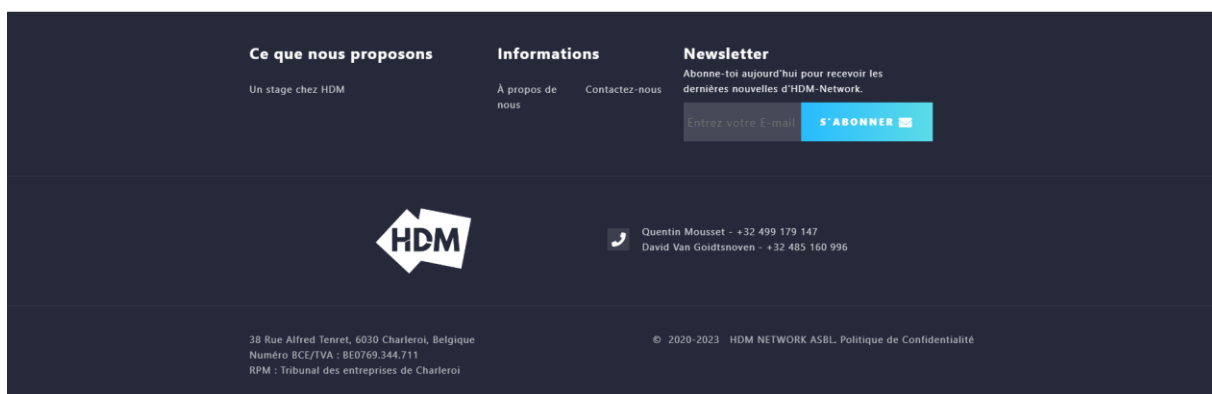
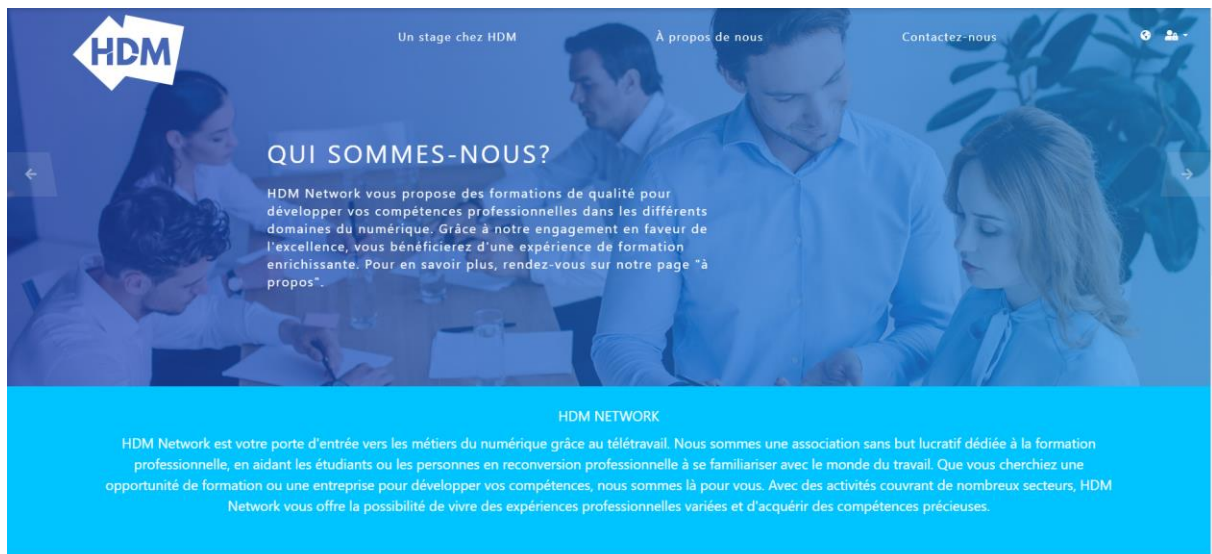


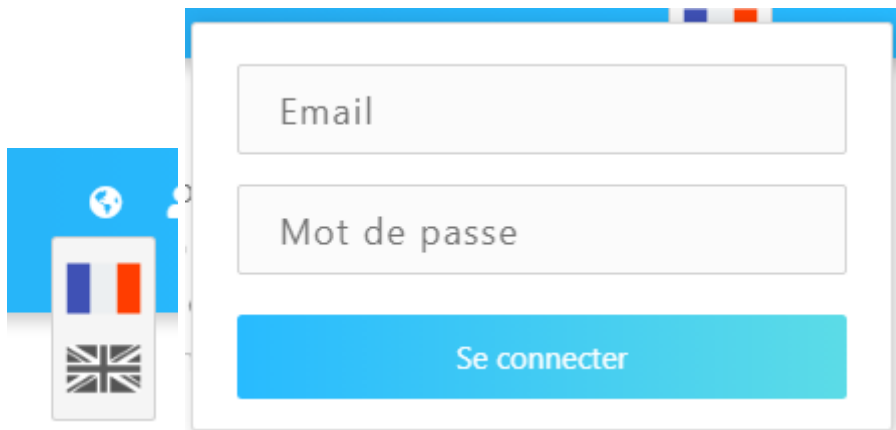
Footer :





La page d'accueil de base du site HDM Network :





Difficultés rencontrées :

-Je n'ai pas réussi à faire le formulaire de connexion déroulant ainsi que le changement de langue comme ci-dessus , malgré maintes essais via javascript, je n'ai pas réussi à implémenter ces deux choses sur ma version de la page d'accueil.

-Difficultés également rencontrées lors du carrousel , je n'ai pas encore un excellent niveau en javascript et j'avais alors du mal à le faire de moi-même , même si j'avais à ma disposition via internet des centaines de templates de carrousel réutilisables.

### 3. Défi jeu 3D

Cette mission nous a été assignée par Quentin Mousset , nous lançant tous au défi de mettre au point un simple jeu FPS (first person shooter) afin de découvrir l'utilisation de la librairie A-Frame.

A-Frame est une bibliothèque JavaScript open source qui permet de créer des expériences de réalité virtuelle (RV) et de réalité augmentée (RA) dans un navigateur web. Elle a été développée par Mozilla et est

basée sur les spécifications WebVR (spécification qui définit une API JS pour accéder aux casques de réalité virtuelles , et donc créer des applications VR sur site web) et WebAR (Même chose mais axée sur la réalité augmentée).

A-Frame facilite la création de scènes 3D interactives en utilisant des balises HTML. Cela nous permet donc de créer rapidement des applications de RV et de RA sans se compliquer la tâche avec des technologies plus complexes.

La bibliothèque A-Frame est construite sur Three.js, qui est une bibliothèque JavaScript populaire pour la création de graphiques 3D dans un navigateur. Elle fournit une abstraction plus simple pour créer des scènes 3D en utilisant une syntaxe HTML déclarative.

Grâce à A-Frame, les développeurs peuvent ajouter des objets 3D, des modèles, des lumières, des sons et des interactions à leurs scènes VR/AR en utilisant des composants préfabriqués. Il est également possible d'utiliser des bibliothèques et des outils tiers pour étendre les fonctionnalités d'A-Frame.

A-Frame est compatible avec la plupart des navigateurs modernes qui prennent en charge la RV et la RA. Elle offre une bonne flexibilité pour la création d'expériences immersives et constitue un bon choix pour les développeurs qui souhaitent se lancer dans le développement de contenu VR/AR sur le web.

Voici donc l'ensemble des technologies que j'ai utilisé sur cette mission :

- HTML
- JavaScript
- Bibliothèque A-Frame

Logiciels utilisés :

- Visual Studio Code
- Blender

Le cahier des charges que l'on devait suivre :

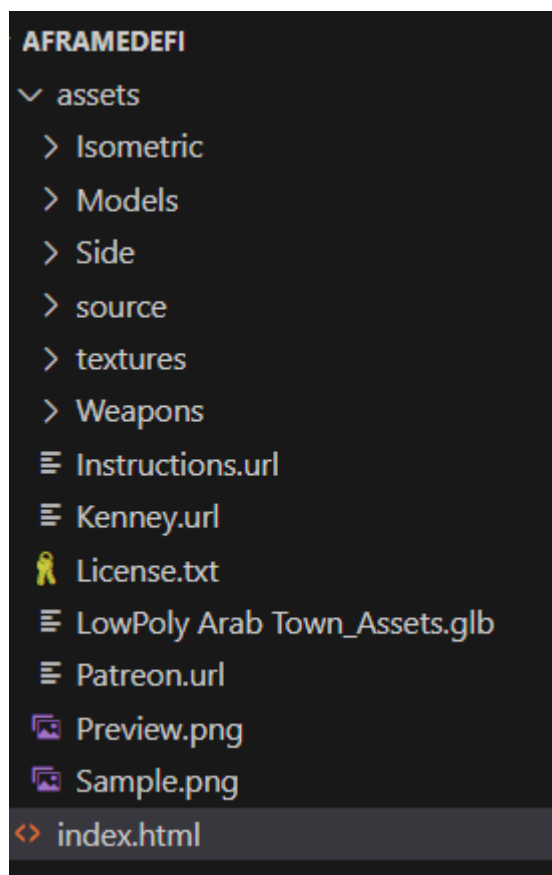
- Faire un jeu de tir a la première personne dans lequel on élimine des ennemis
- Chaque élimination apporte une augmentation de score
- Barre de vie du joueur qui baisse s'il se fait toucher
- Pouvoir se déplacer
- Avoir une ressource graphique pour l'arme du joueur
- Créer un environnement de jeu

Pour se faire , j'ai alors recherché comme pour le jeu 2D des ressources libres de droit pour créer mon environnement , et également avoir ma ressource graphique pour l'arme.

Concernant la barre de vie , j'ai choisis de simplement faire une barre verte qui baisse petit à petit comme pour le jeu 2D.

Pour les ennemis, j'ai créer via A-Frame des formes géométrique tel que des cubes ou des sphères rouges , pour les représenter , qui se déplacent tous vers le joueur et apparaissent autour de lui.

Voici l'architecture du projet :



Le plus conséquent du code sera le script javascript. Pour la partie purement HTML de A-Frame, il suffit de créer une scène et d'y ajouter les entités souhaitées tel que le joueur, l'ennemi et l'environnement de la manière suivante :

```

1 <a-scene>
2   <a-assets>
3     <a-asset-item id="gun" src="/assets/source/gun.glb"/></a-asset-item>
4   </a-assets>
5   <a-asset-item id="gun" src="/assets/source/gun.glb" class="environment" bounding-box static-body/></a-asset-item>
6
7   <!-- Player -->
8   <a-entity id="player" position="278 0 144" look-controls="pointerLockEnabled: true" camera look-controls wasd-controls geometry="primitive: box; width: 0.5; height: 1.8; depth: 0.5" aabb-collider="objects: .environment">
9     <a-entity id="gun" gltf-model="gun" scale="10 10 10" position="1 0 3" rotation="180 0 180"/></a-entity>
10    <a-cursor/></a-entity>
11
12   <!-- Bouton de démarrage -->
13   <a-entity id="startButton" position="278 0 144">
14     <a-box width="2" height="1" depth="0.2" color="blue"/></a-box>
15     <a-text value="Jouer" align="center" position="0 0 1" color="white"/></a-text>
16   </a-entity>
17
18   <!-- Barre de vie -->
19   <a-entity id="healthBar" position="1.5 2 3" look-at="#player">
20     <a-plane width="1.5" height="0.2" color="yellow"/></a-plane>
21     <a-plane id="health" width="1.5" height="0.2" color="red"/></a-plane>
22   </a-entity>
23
24   <!-- Score -->
25   <a-entity id="score" position="1.5 1.7 3" look-at="#player" text="value: Score: 0; color: #000000"/></a-entity>
26
27   <!-- Conteneurs pour les balles et les ennemis -->
28   <a-entity id="bullets"/></a-entity>
29   <a-entity id="enemies"/></a-entity>

```

Concernant le code javascript, j'ai simplement fait des fonctions différentes pour chaque aspect du jeu, tel que le tir, la création de

l'ennemi, la position de celui-ci , la mise a jour du score, la mise a jour de la barre de vie ainsi que le Game over.

Cependant, j'ai utilisé une méthode différente pour les collisions : J'ai créé un composant « personnalisé » A-Frame afin de mettre à jour la position de la balle à chaque frame et y gérer les collisions.

A-Frame nous permet de créer nos propres composants personnalisés en utilisant JavaScript, ce qui permet de manipuler les entités 3D et d'ajouter des fonctionnalités spécifiques à une scène VR/AR.

Un composant personnalisé est défini en tant qu'objet JavaScript qui contient des propriétés, des méthodes et des événements spécifiques. Il peut être attaché à une entité A-Frame dans une scène en utilisant une balise HTML personnalisée. Lorsque le composant est attaché, il modifie ou étend le comportement de l'entité en fonction de sa logique interne.

Par exemple, vous pourriez créer un composant personnalisé pour gérer le mouvement d'une entité lorsque l'utilisateur appuie sur une touche spécifique. Vous définiriez les propriétés et les méthodes nécessaires pour gérer ce comportement, puis vous attacheriez ce composant à l'entité souhaitée dans votre scène A-Frame.

J'ai donc défini dans ce composant une vitesse de déplacement du tir, et calculer la nouvelle position du tir en fonction de sa position actuelle et de sa direction multipliée par sa vitesse, dans les directions x , y et z.

Pour tester les collisions avec l'ennemi, j'ai alors récupéré les deux entités en récupérant leurs attributs avant de boucler avec un foreach pour vérifier que la position du tir approche une certaine distance l'entité représentant l'ennemi, et alors par conséquent supprime les deux entités.

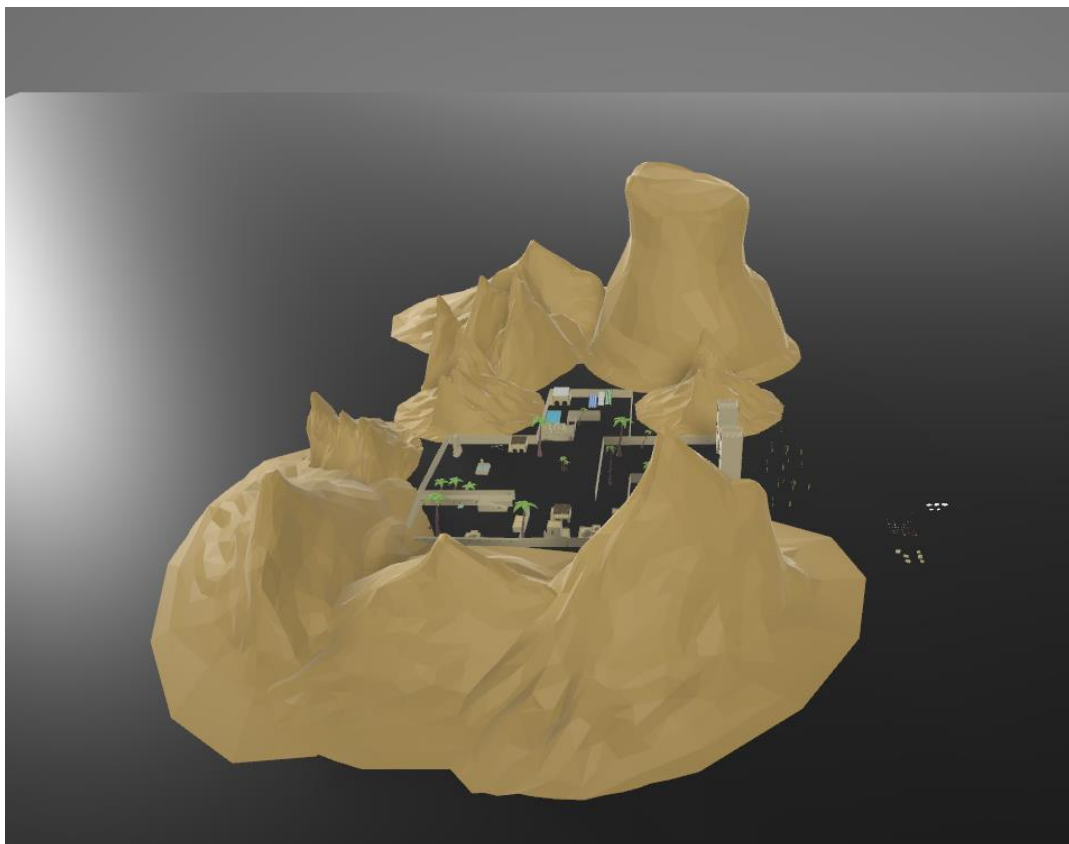
Voici le code représentant l'explication que je viens de vous donner :

```
1 AFRAME.registerComponent('movement', {
2   schema: {
3     direction: { type: 'vec3', default: { x: 0, y: 0, z: -1 } },
4   },
5
6   tick: function () {
7     if (!gameStarted) return;
8
9     var direction = this.data.direction;
10    var currentPosition = this.el.getAttribute('position');
11    var speed = 0.1; // Vitesse de déplacement de la balle
12    var activeBullets = 0;
13
14    // Calcul de la nouvelle position de la balle
15    var newPosition = {
16      x: currentPosition.x + direction.x * speed,
17      y: currentPosition.y + direction.y * speed,
18      z: currentPosition.z + direction.z * speed,
19    };
20
21    // Mise à jour de la position de la balle
22    this.el.setAttribute('position', newPosition);
23
24    // Test de collision avec les ennemis
25    var bullets = document.querySelectorAll('.bullet');
26    var enemies = document.querySelectorAll('.enemy');
27    var playerPosition = document.querySelector('#player').getAttribute('position');
28
29    bullets.forEach(function (bullet) {
30      activeBullets++;
31      enemies.forEach(function (enemy) {
32        if (bullet.getAttribute('position').distanceTo(enemy.getAttribute('position')) < 0.2) {
33          if (enemy.parentNode) {
34            // Collision détectée, supprimer la balle et l'ennemi
35            enemy.remove();
36            bullet.remove();
37
38            // Augmenter le score
39            score++;
40            updateScore();
41          }
42        } else if (enemy.getAttribute('position').distanceTo(playerPosition) < 0.5) {
43          // Collision détectée, baisse les points de vie
44          reduceHealth(10);
45        }
46      });
47    });
48
49    // Déplacement des ennemis vers le joueur
50    enemies.forEach(function (enemy) {
51      var enemyPosition = enemy.getAttribute('position');
52      var enemyDirection = new THREE.Vector3();
53      var averageSpeed = enemy.getAttribute('data-speed') / activeBullets;
54      enemyDirection.subVectors(playerPosition, enemyPosition).normalize();
55      var newEnemyPosition = {
56        x: enemyPosition.x + enemyDirection.x * averageSpeed,
57        y: enemyPosition.y + enemyDirection.y * averageSpeed,
58        z: enemyPosition.z + enemyDirection.z * averageSpeed,
59      };
60      enemy.setAttribute('position', newEnemyPosition);
61    });
62  },
63 });
```

Une fois toutes les fonctionnalités principales du jeu fonctionnelles, j'ai donc cherché à faire l'environnement de jeu. Après avoir trouvé les ressources graphiques nécessaires, j'ai installé le logiciel « Blender ».

Blender est un logiciel de modélisation 3D, d'animation et de création d'effets visuels. J'ai donc cherché à installer ce logiciel gratuit pour modéliser mon environnement puisqu'il est notamment très utilisé dans l'industrie des jeux vidéo, conception de produit, architecture ou même animation. Son interface étant simple à prendre en main, c'était le logiciel optimal à choisir afin de modéliser petit à petit mon environnement de jeu.

Voici à quoi ressemble donc l'environnement de jeu et l'arme utilisée pour les ressources graphique de mon jeu 3D :







#### DIFFICULTÉS RENCONTRÉES :

- Énormément de problèmes lors de l'implémentation des collisions.
- Incompréhension totale de comment fonctionnait les composants de base de A-Frame en lien avec les collisions, que je n'arrivais pas à appliquer à mon projet.

#### 4. Scrapping de données TripAdvisor



Nous avons été chargés de Scrapper les données d'une liste de restaurants situés à Charleroi, répertoriés sur le site de TripAdvisor.

La mission a été donnée par groupe de 2, et j'ai donc été accompagné d'un autre membre du pôle développement Web afin d'accomplir cette tâche.

Les différentes étapes du projet sont les suivantes :

Téléchargement de la liste des établissements : Nous commençons par télécharger la page principale de TripAdvisor qui contient la liste des établissements. Nous utilisons une fonction `downloadPageWithTimeout` pour télécharger la page avec un délai d'attente (timeout) de 10 secondes pour éviter les temps d'attente excessifs.

Extraction des liens des pages d'établissement : Une fois que nous avons téléchargé la page principale, nous utilisons le package `node-html-parser` pour extraire les liens des pages d'établissement à partir

du code HTML de la page. Nous utilisons un sélecteur personnalisé pour cibler les éléments HTML contenant les liens.

Parcours des pages d'établissement : Nous itérons sur chaque lien extrait et téléchargeons le contenu HTML de la page d'établissement correspondante en utilisant la fonction `downloadPageWithTimeout`. Encore une fois, nous utilisons un délai d'attente de 10 secondes pour chaque téléchargement.

Extraction des informations des établissements : Une fois que nous avons téléchargé le contenu HTML de chaque page d'établissement, nous utilisons à nouveau le package `node-html-parser` pour extraire le nom et le numéro de téléphone de chaque établissement en utilisant des sélecteurs personnalisés. Nous créons ensuite un objet contenant ces informations.

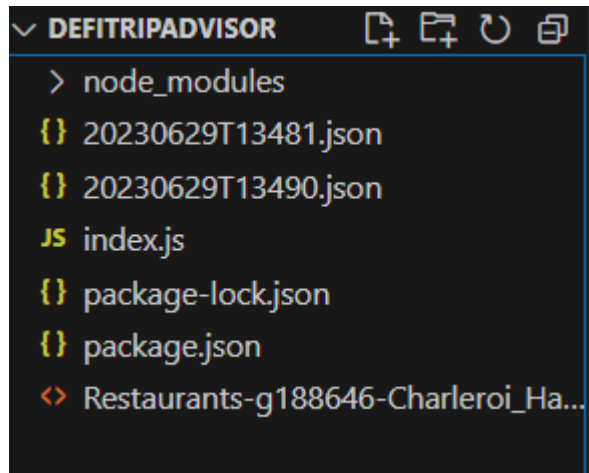
Stockage des informations : Nous stockons les informations extraites dans une structure de données appelée `outputData`. Il s'agit d'un tableau d'objets, où chaque objet représente un établissement avec ses informations. Nous filtrons également les données valides pour exclure les établissements pour lesquels les informations n'ont pas pu être extraites.

Enregistrement dans un fichier JSON : Enfin, nous convertissons la structure de données `outputData` en une chaîne JSON à l'aide de la fonction `JSON.stringify`. Nous créons un nom de fichier basé sur la date et l'heure actuelles, puis nous enregistrons la chaîne JSON dans un fichier JSON sur le disque.

C'est essentiellement le processus du projet. Nous utilisons des bibliothèques et des packages tels que `node-wget`, `node-html-parser` et `fs` pour faciliter les opérations de téléchargement, d'analyse HTML et d'écriture de fichiers.

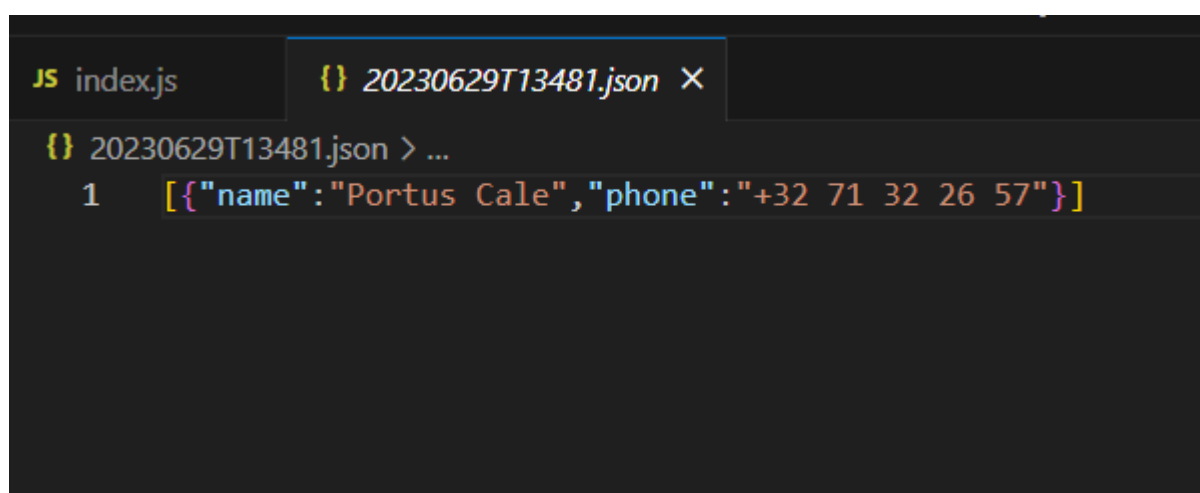
L'objectif principal du projet est de collecter les informations des établissements à partir du site TripAdvisor de manière automatisée

Voici donc un exemple de ce que donne l'architecture du projet, une fois le code exécuté :



Initialement, il n'y a que l'index.js ainsi que les package. Après exécution, la page est téléchargée et on peut donc voir un fichier HTML se créer, ainsi qu'un fichier JSON nommé par la date et heure du moment où le programme a été exécuté.

Par exemple ici , dans le fichier JSON correspondant au moment de l'exécution on retrouve alors les données extraites du site tels que le numéro de téléphone et le nom du restaurant d'un établissement :



Difficultés rencontrées :

-Énormément de timeout : C'est-à-dire que le temps de réponse du site était bien trop faible, et donc une fois sur 10 l'exécution du programme fonctionnait et les données étaient bien récupérées.

-Trop de requêtes : A force d'émettre des requêtes vers le site de Tripadvisor , celui-ci finis par nous bannir notre adresse IP comme si c'était une menace DDOS , un envoie d'une multitude de paquets via requêtes afin de faire crash leur serveurs.

Pour palier à ce problème , nous utilisons alors des proxys pour pouvoir continuer d'envoyer des requêtes et donc récupérer les données nécessaires.

-Problème d'autorisation : Selon la librairie, j'ai remarqué qu'il y avait des problèmes d'autorisation d'accès aux données du site. Par exemple, lorsque j'ai utilisé Axios, j'ai reçu une erreur avec le code d'état 403. Cela signifie donc que j'ai tenté d'accéder à une ressource à laquelle je ne suis pas autorisée d'accéder sur le serveur de Tripadvisor.

Le serveur renvoi donc comme réponse :

Status : 403 Forbidden

<HTML>

<HEAD>

<TITLE>Access Denied</TITLE>

</HEAD>

<BODY>

<H1>Access Denied</H1>

You don't have permission to access  
"http://fr.tripadvisor.be/Restaurants-g188646-  
Charleroi\_Hainaut\_Province\_Wallonia.html" on this server.<P>  
Reference #18.845d645f.1688043342.204ed6e3  
</BODY>  
</HTML>"

# CONCLUSION

En conclusion de mon stage au sein de l'entreprise HDM Network, j'ai acquis une expérience enrichissante et des connaissances approfondies dans plusieurs domaines clés. Les formations que j'ai suivies en JavaScript, React et SEO ont été des éléments cruciaux de mon apprentissage et ont renforcé mes compétences dans le domaine du développement web.

La formation en JavaScript m'a permis de maîtriser les fondamentaux de ce langage de programmation essentiel. J'ai acquis une compréhension solide des concepts de base, tels que les variables, les fonctions et les boucles, ainsi que des concepts avancés tels que la programmation orientée objet. Grâce à JavaScript, je suis en mesure de développer des fonctionnalités interactives et dynamiques sur les sites web.

La formation en React a été une étape importante dans mon parcours. J'ai appris à concevoir et à développer des interfaces utilisateur réactives et modulaires en utilisant cette bibliothèque populaire. Les concepts tels que les composants, le rendu conditionnel et la gestion de l'état m'ont permis de créer des applications web modernes et performantes. La maîtrise de React est devenue une compétence très recherchée sur le marché du travail.

La formation en SEO m'a ouvert les portes de l'optimisation pour les moteurs de recherche. J'ai acquis des compétences précieuses pour améliorer la visibilité et le classement d'un site web dans les résultats de recherche. J'ai appris à optimiser les mots-clés, à structurer le contenu, à créer des liens de qualité et à suivre les performances du site. Ces connaissances en SEO sont essentielles pour accroître la notoriété d'une entreprise en ligne et générer du trafic qualifié.

Au-delà des compétences techniques, ce stage m'a également offert une expérience professionnelle concrète au sein de l'entreprise HDM Network. J'ai pu travailler au sein de différentes équipes et collaborer avec des professionnels expérimentés. J'ai acquis une compréhension approfondie du fonctionnement d'une entreprise dans le domaine du numérique et j'ai pu développer mes compétences en matière de communication, de gestion de projets et de travail d'équipe.

Dans l'ensemble, ce stage m'a permis de consolider mes connaissances techniques et de développer mes compétences professionnelles. Les formations en JavaScript, React et SEO m'ont préparé pour des opportunités futures dans le domaine du développement web et de l'optimisation des sites. Je suis reconnaissant envers l'équipe de HDM Network pour leur encadrement et leur soutien tout au long de cette expérience. Ce stage a été une étape importante dans mon parcours professionnel et a renforcé ma passion pour le domaine du numérique.