



UNIVERSITÉ PARIS 8 - VINCENNES À SAINT-DENIS

Licence informatique & vidéoludisme

Projet Final Algorithmique Avancée

”Univ-Fighter” émerge en tant que projet audacieux, s’inspirant des illustres jeux de versus fighting, avec une claire référence au célèbre Street Fighter. L’équipe de développement s’est attaquée à la tâche colossale de recréer l’essence même de ces jeux emblématiques tout en ajoutant sa propre touche unique.

**Mehdi Oudghiri
Ludovic Renault
Bastien Capiaux
Tomas Pereira Oliveira**

Date de rendu : le 18/12/2023

Groupe : L3-A

Contents

1	Présentation	2
2	Font-end	2
2.1	Menu	2
2.2	Les assets	3
2.3	La partie sonore	4
2.3.1	La musique	4
2.3.2	les bruitages	4
3	Back-end	4
3.1	Interface et Interaction	4
3.2	Dynamiques du Jeu dans Street Fighter	4
3.2.1	Règles et Paramètres de Combat	4
3.2.2	Création et Personnalisation des Personnages	5
3.3	Gestion et performance	5
3.4	Composants Additionnels	6
4	Conclusion	6

1 Présentation

Dans la poursuite de notre exploration du projet "Univ-Fighter", plongeons dans les mécanismes qui animent ce jeu de combat en versus fighting. Dans la section 2, nous examinerons l'implémentation du front-end (2.1) pour découvrir comment l'esthétique visuelle et l'expérience utilisateur prennent vie. Ensuite, nous nous aventurerons dans les coulisses avec la section 2.2, détaillant l'implémentation du back-end, où la logique du jeu se dévoile.

2 Font-end

2.1 Menu

Le menu constitue une composante essentielle de l'interface utilisateur de notre jeu "Univ-Fighter". Son design et son fonctionnement ont été réalisés en utilisant principalement des technologies front-end telles que HTML, CSS, et JavaScript.

Mise en Page et Style

Le corps de la page utilise un modèle de boîte flexible (flexbox) pour centrer le contenu à l'écran, créant ainsi une mise en page équilibrée. Le fond est illustré par une image de fond dynamique (BG2.gif) qui renforce l'aspect visuel du menu. Les éléments de menu sont positionnés de manière stratégique pour assurer une expérience utilisateur intuitive.

Effets Visuels

Le survol des boutons du menu déclenche des animations visuelles attrayantes. Chaque bouton est associé à une animation "jelly" qui apporte une légère déformation, ajoutant ainsi un effet visuel distinctif et engageant.

Sons et Curseur Personnalisé

Le menu est accompagné d'une ambiance sonore grâce à l'utilisation d'une piste audio en boucle (SmelloftheGame.mp3). De plus, un effet sonore distinct (dmc5_button.mp3) est déclenché lors du clic sur les boutons du menu, renforçant l'interaction utilisateur. Le curseur de la souris est également personnalisé pour offrir une expérience immersive supplémentaire.

Boutons et Liens

Chaque bouton du menu est conçu avec soin, associé à une icône et un texte descriptif. Les liens de navigation sont mis en œuvre en utilisant des éléments de type `ja<` pour assurer un accès rapide et fluide aux différentes sections du jeu.

2.2 Les assets

Les assets, éléments graphiques utilisés dans notre jeu "Univ-Fighter", jouent un rôle essentiel dans l'expérience visuelle globale. Ils ont été soigneusement sélectionnés et intégrés pour renforcer le thème du jeu et garantir une esthétique cohérente.

Les assets utilisés dans le jeu ont été obtenus à partir de différentes sources. Certains ont été créés en interne par l'équipe de développement, tandis que d'autres ont été acquis auprès de ressources en ligne. Nous avons par exemple utilisé une IA afin de nous représenter dans la partie crédits, tandis que nous avons tiré les spritesheets des deux combattants du site "<https://www.sprites-resource.com/>". Où encore les boutons qui ont pu être créés grâce au site "<https://www.textstudio.com/logo/street-fighter-logo-972>" qui permet de créer des images avec un texte se voulant d'un visuel similaire à street fighter. Qui nous ont ensuite servis d'assets pour les différents boutons du menu.

Les différentes images et sprites sont intégrés dans le code HTML à l'aide de balises `img`. Chaque image est associée à une balise avec l'attribut `alt` spécifiant le nom du personnage ou de l'élément graphique. Ces balises `img` permettent au code JavaScript d'accéder et de manipuler les images dynamiquement pendant l'exécution du jeu.

La gestion des animations dans "Univ-Fighter" repose sur un découpage judicieux de la sprite sheet, une image regroupant l'ensemble des frames nécessaires pour animer les personnages.

Chaque sprite sheet est soigneusement organisée pour garantir une gestion efficace des frames d'animation. Les frames associées à chaque action spécifique, telle que la marche, le saut, ou un coup particulier, sont regroupées dans des zones distinctes de la sprite sheet.

Chaque entrée dans le tableau, servant à découper les animations, représente les coordonnées (x, y, largeur, hauteur) d'une frame spécifique de l'animation. Ces coordonnées sont utilisées pour extraire dynamiquement la frame correspondante de la sprite sheet.

Cette approche offre une flexibilité et une modularité permettant d'ajuster facilement les animations en fonction des besoins du jeu.

2.3 La partie sonore

2.3.1 La musique

Le menu représente une composante essentielle dans l'expérience d'un jeu. Dans notre jeu Son design et son fonctionnement ont été réalisés en utilisant principalement des technologies front-end que sont l'HTML pour la jouer et le JavaScript pour modifier le volume. La musique accompagne le menu principale ainsi que les combats et démo réalisé, elle proviennent toutes du dernier opus de la série de jeu de combat "Guilty Gear", Gealty Gear Strive. Ces musiques entrainente permettent de sublimer l'expérience des joueurs.

2.3.2 les bruitages

Chaque actions de notre jeu est accompagné d'un bruitage associé pour renforcer l'immersion des joueurs, et cela dès le menu principale, en effet, chaque appui sur un bouton joue, en plus de sa fonctionnalité principal, un son. De plus, en combat chaque action des personnage s'accompagne d'un bruitage le représentant, la diversité de ces bruitages, de part leurs différence sonore mais aussi leurs origines (provenant de différent jeux, parfois très différent les uns des autres, certains venant de la séries de jeu Devil May Cry, d'autre de l'animé Dragon Ball Z ainsi que ceux venant de Street Fighter) crée ainsi une expérience divertissante et unique.

3 Back-end

3.1 Interface et Interaction

StatusBar.js assure que les joueurs ont des informations actualisées sur la santé des combattants. Les contrôles sont rendus intuitifs grâce à "controls" dans control.js .

3.2 Dynamiques du Jeu dans Street Fighter

3.2.1 Règles et Paramètres de Combat

Dans battle.js , elle est essentielle pour définir les règles fondamentales du combat. Elle spécifie des éléments tels que les points de vie, les dégâts des attaques, et les conditions de victoire. Parallèlement, stage.js joue un rôle clé dans la configuration des arènes. Cette fonction détermine les dimensions de l'arène, les obstacles et les particularités de chaque environnement de combat. Ces éléments ont un impact direct sur la stratégie et la jouabilité

3.2.2 Création et Personnalisation des Personnages

Caractéristiques Principales de "Fighter.js"

1. **Structure et Initialisation** : Le script définit la classe `Fighter` , qui initialise les propriétés clés des personnages, telles que la position, la vitesse, l'orientation, l'animation et divers états.
2. **Gestion des États et des Mouvements** : À travers un système d'états bien définis, le combattant peut passer de différentes actions comme marcher, sauter, se défendre et attaquer. Chaque état a ses propres méthodes d'initialisation et de mise à jour, permettant des transitions fluides et des actions réactives.
3. **Intégration avec d'Autres Modules** : Le script interagit avec d'autres parties du projet, comme les systèmes de contrôle d'entrée (`InputHandler.js`), les fonctions de collision (`collision.js`), et les constantes de jeu (importées de `dfight.js` et `stage.js`).
4. **Animations et Graphismes** : Il gère les animations du combattant en fonction de l'état actuel, assurant que les mouvements et les réactions sont visuellement cohérents et alignés avec les actions du joueur.
5. **Interactions de Combat** : Des fonctions telles que `updateAttackBoxCollided` permettent de gérer les interactions complexes pendant les combats, comme la détection des coups et la gestion des dommages.
6. **Flexibilité et Personnalisation** : La classe offre une base solide pour la personnalisation des combattants, permettant l'ajout facile de nouveaux mouvements, attaques, et comportements spécifiques.
7. **Gestion des Sons et Effets** : Le script intègre également des effets sonores correspondant aux différentes actions, ajoutant une couche d'immersion au gameplay.

En somme, "Fighter.js" est un composant clé dans le projet `Street Fighter`, Les fichiers spécifiques aux personnages, tels que `Bastien.js` , s'appuient sur cette base pour développer des combattants uniques. Chaque personnage possède des compétences et des animations propres.

3.3 Gestion et performance

La performance est surveillée via dans `FpsCounter.js` , assurant une expérience de jeu fluide. `gameState.js` et `fighterState.js` gèrent les aspects dynamiques du jeu, tels que les scores et l'état des combattants.

3.4 Composants Additionnels

Le jeu se construit autour de `index.js` et `indexDemo.js`, qui lancent les différentes versions du jeu en intégrant divers modules. Au cœur du jeu, `StreetFighter.js` pilote les mécaniques principales, tandis que dans `StreetFighterDemo.js` adapte ces mécanismes pour la version démo. En ce qui concerne l'interface utilisateur, `menu_buttons.js` facilite une navigation sans effort à travers les menus du jeu, et dans `camera.js` enrichit l'expérience visuelle avec des mouvements de caméra dynamiques et immersifs.

4 Conclusion

"Univ-Fighter" émerge comme un projet passionnant, imprégné de l'héritage du versus fighting et influencé par le légendaire Street Fighter. Conçu en HTML, CSS, et JavaScript par, le jeu offre une expérience visuelle soignée avec un front-end intuitif et des assets esthétiquement cohérents.

La dimension sonore, avec des musiques dynamiques de "Guilty Gear Strive" et des bruitages variés, renforce l'immersion à chaque étape du jeu. Côté back-end, la gestion des personnages à travers "Fighter.js" apporte une flexibilité remarquable, donnant vie à des combattants uniques.

En outre, "Univ-Fighter" ne se contente pas de reproduire, mais ajoute une touche personnelle. Chaque personnage, avec ses compétences distinctes, contribue à une expérience de jeu fluide. Ce projet incarne la passion de l'équipe, fusionnant habilement tradition et innovation dans le monde du versus fighting.