

Projet: Castlevania

Moteur de Jeux

Auteurs :

Damien GEOFFROY L3-B

Viet NGUYEN L3-B

Wilfried BEN BRAHIM L3-A

Jonny MATHANARUBAN L3-A

Enseignant :

N. Jouandeu

LICENCE INFORMATIQUE 3

UNIVERSITÉ PARIS 8
2023

INTRODUCTION:

Dans le cadre de notre projet en *Moteur de Jeux*, nous avons dû concevoir un jeu vidéo principalement codé en JavaScript. L'objectif était que le jeu soit esthétiquement plaisant, intègre de la musique et puisse être lancé hors ligne sur plusieurs machines différentes. Pour ce projet, notre choix s'est porté sur le développement d'un jeu de plateforme en 2D de type Castlevania.

Castlevania, une franchise de jeux vidéo débutée en 1986 sur le Famicom Disk System, équivalent de la Nintendo Entertainment System (NES), propose une expérience où le joueur incarne un héros devant souvent faire face à un ennemi puissant, lié de près ou de loin à Dracula, considéré comme l'entité représentant le mal dans ce monde.

Notre décision de créer un jeu de plateforme inspiré de Castlevania découle de son style gothique, perceptible tant au niveau des graphismes, du décor, notamment, que de la police d'écriture et de la musique qui l'accompagne. Et c'est cela que nous avons souhaité retranscrire dans notre projet afin de pouvoir plaire aux fans de ces jeux.

JavaScript a été une des contraintes majeures de ce projet. Pour clarifier, JavaScript est un langage de programmation qui permet d'ajouter des fonctionnalités interactives aux sites web. Dans le contexte de notre jeu, cela nous a permis d'implémenter divers mécanismes de jeu et de proposer une expérience utilisateur fluide et dynamique.

En effet, grâce à JavaScript, il est possible de développer des jeux à la fois en 2D et en 3D en prenant en charge des aspects cruciaux tels que la gestion des collisions, la manipulation du temps et le traitement simultané de plusieurs images. Cette flexibilité s'avère particulièrement adaptée à notre public cible, composé de joueurs familiers avec l'univers de Castlevania et habitués aux jeux sur Famicom/NES.

Les joueurs de Castlevania recherchent une expérience de jeu qui respecte les codes classiques, hérités des anciens jeux et JavaScript, en tant que moteur de développement, nous offre la capacité de créer un jeu fonctionnel et complet qui correspond aux attentes de ces joueurs.

Par conséquent, notre choix de travailler avec JavaScript n'est pas uniquement dicté par ses possibilités techniques, mais aussi par la volonté de proposer une expérience de jeu qui résonne avec les souvenirs et les attentes des joueurs de Castlevania, offrant ainsi un équilibre subtil entre fonctionnalités modernes et respect des codes intemporels.

Dans ce rapport, nous explorerons notre démarche lors de la conception de notre jeu vidéo de plateforme 2D, inspiré de Castlevania. Nous aborderons la création du jeu, les défis que nous avons rencontrés, et enfin, nous présenterons les résultats obtenus au travers d'une brève présentation.

CONCEPTION ET DEVELOPPEMENT:

Bien que notre rapport de travail soit très fourni et détaillé, nous ne pourrions pas explorer en détail chaque fonction de chaque fichier dans cette partie. Cependant, nous avons la possibilité de vous fournir une explication claire de l'architecture du projet et de l'utilité de chaque fichier. Mais nous pourrions expliquer l'arborescence de notre projet ainsi chaque fichier se retrouve dans un fichier qui lui est dédié de manière logique. Chaque niveau possède ses propres informations qui seront séparé de celle des menus, couplé avec l'utilisation de commentaires dans notre code ceci a facilité la programmation asymétrique.

Voici l'arborescence de notre projet

- **HTML:**
 - *"index.html"* : main menu du jeu
 - *"game.html"* : lancer le jeu
- **CSS:**
 - *"style.css"* : style du menu
 - *"game_ui.css"* : style du jeu
- **JAVASCRIPT** (path: js/):
 - *"menu/"* contient tous les fichiers JavaScript liés à la configuration du menu
 - *"menu_background.js"* : Exécuter l'image d'arrière-plan pour le menu principal
 - *"menu_buttons.js"* : Exécuter les boutons
 - *"menu_credits.js"* : Exécuter la page crédits du menu
 - *"menu_settings.js"* : Exécuter la page des paramètres du menu
 - *"menu_pause.js"* : Exécuter la page de pause du jeu
 - *"game/"* contient tous les fichiers JavaScript du jeu
 - *"levels_setup.js"* : gérer les niveaux de jeu
 - *"stats.js"* : gérer les attributs, les données pour sauvegarder et charger le jeu
 - *"baselevel/"* exemple pour exécuter un niveau du jeu
 - *"controller.js"* : le contrôleur alerte uniquement l'utilisateur chaque fois qu'il appuie sur une touche, mais il définit également la classe `ButtonInput`, qui est utilisée pour suivre l'état des boutons
 - *"display.js"* : gestionnaire d'événements de redimensionnement d'écran et gère également dessiner les couleurs dans le tampon, puis dans l'écran
 - *"game.js"* : logique pour le niveau du jeu, progressivement modifiez
 - *"engine.js"* : il s'agit d'une boucle de jeu à pas de temps fixe
 - *"main.js"* : Il s'agit de la configuration de base ou "squelette" de programme. Il comporte trois parties principales : le contrôleur,

l'affichage et la logique du jeu. Il dispose également d'un moteur qui combine les trois parties logiques qui sont par ailleurs complètement distinctes. Séparer le code en logiques groupées est aussi un principe de programmation orientée objet, qui se prête à un code compréhensible et maintenable ainsi qu'à la modularité.

- *“lv1/”* Le premier niveau du jeu (Les fichiers sont déjà expliqués dans ***baselevel/***)
 - *“controller-01.js”*
 - *“display-01.js”*
 - *“game-01.js”*
 - *“main-01.js”*
- *“lv2/”* Le 2ème niveau du jeu :
 - *“controller-02.js”*
 - *“display-02.js”*
 - *“game-02.js”*
 - *“main-02.js”*
- *“effects/”* contient tous les fichiers JavaScript liés aux effets visuels du projet
 - *“butterfly.js”* : Exécuter au hasard de 1 à 4 papillons en mouvement sur l'écran
 - *“fireflies.js”* : Exécuter de petits cercles de lumière qui se déplacent de manière aléatoire sur l'écran
 - *“fog.js”* : Exécuter l'image de brouillard en mouvement en bas de l'écran
 - *“transition.js”* : L'écran s'assombrit progressivement pour changer de scène
- *“sound/”* contient les fichiers JavaScript liés à l'audio du projet
 - *“music.js”* : contient de la musique de fond et des fonctions pour les utiliser
 - *“sound.js”* : contient des sons et des fonctions pour les utiliser

RESULTATS ET DEMONSTRATION:

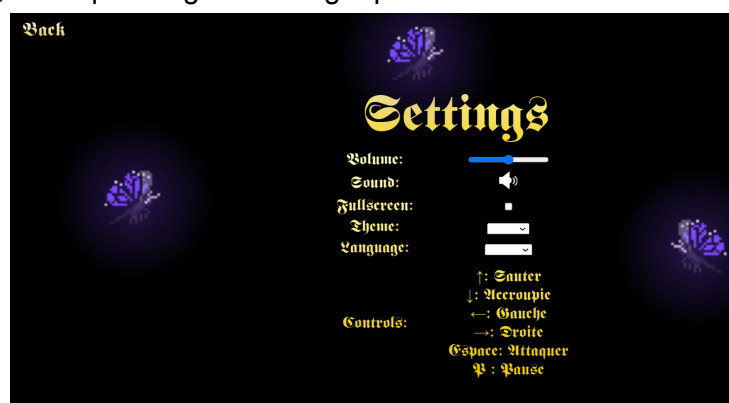
Dans cette section destinée à décrire les composants du projet, nous n'entrerons pas dans les détails puisque le jeu sera fourni accompagné de ce rapport. Tout d'abord, lorsque l'utilisateur lance le jeu, il sera accueilli par un écran de chargement portant le nom du jeu :

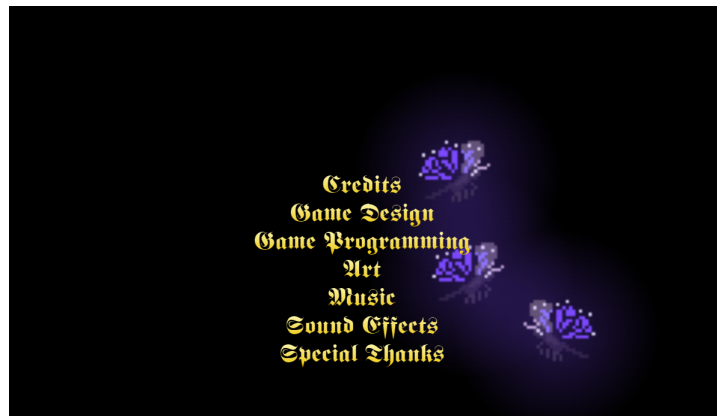


Le joueur accédera au menu principal, lui offrant plusieurs fonctionnalités telles que la possibilité d'agrandir l'écran en appuyant sur l'icône située en haut à droite, ou encore de naviguer entre les différents menus en cliquant sur les différents boutons.

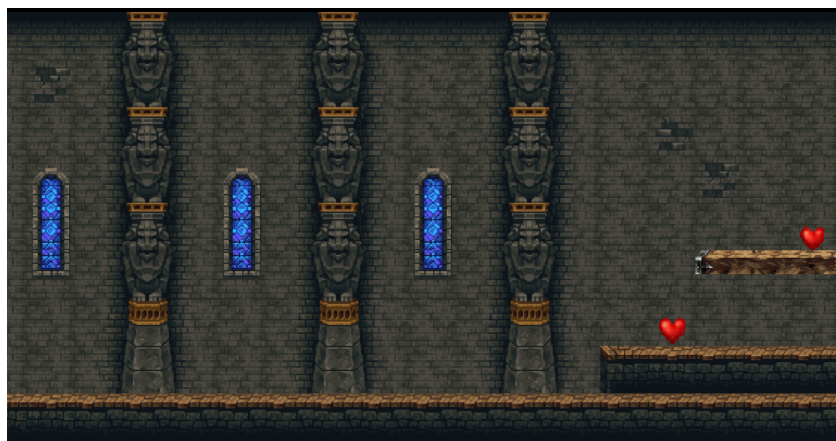


Dans les menus disponibles, il y a l'option "Crédits" qui affichera les informations relatives aux contributeurs, ainsi que l'onglet "Settings" permettant d'effectuer divers ajustements.

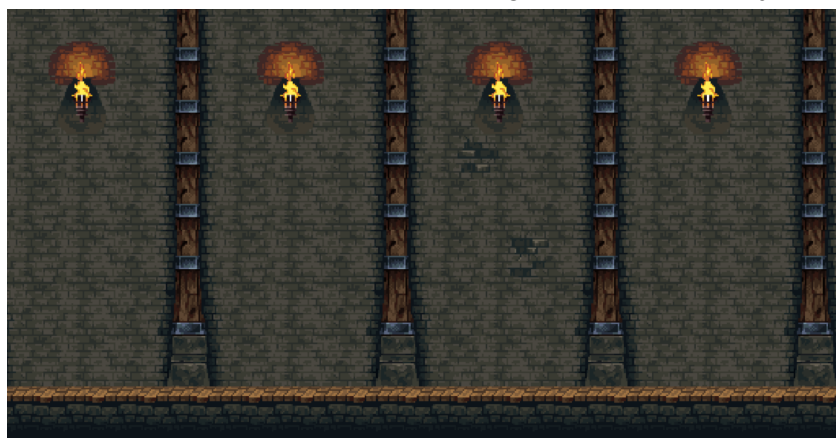




Mais revenons à notre jeu. Pour le lancer, le joueur doit appuyer sur "Jouer", ce qui déclenchera le premier niveau avec un fondu enchaîné. Une fois dans le niveau, représenté ici par une image générée à partir du tilesheet pour une meilleure visibilité (sans les ennemis), le joueur peut se déplacer en utilisant les touches directionnelles ("Gauche", "Haut", "Droite" et "Bas"), et enfin utiliser la touche "ESPACE" pour attaquer.



Après avoir terminé, il pourra fermer l'écran et se diriger vers la suite du jeu:



RÔLES ET RESPONSABILITE:

Dans le cadre de ce projet, nous avons réparti les rôles en fonction des fonctionnalités que nous souhaitions implémenter pour notre jeu. Nous avons choisi de maintenir une organisation flexible afin de ne pas rompre la collaboration et l'entraide entre les membres, notamment pour résoudre les bugs et optimiser l'ensemble de manière cohérente.

Ainsi nous n'avons pas réellement de tâche attribuée même si à un moment donnée les membres se sont focalisés sur une spécificité du jeu ou fonction principales:

- Damien s'est axé sur l'aspect visuel perçu par l'utilisateur, notamment sur les éléments graphiques tels que les tiles. Son travail inclut la recherche, la découpe, l'arrangement et le positionnement de ces éléments. Il a également pris en charge la sélection et la modification des musiques, ainsi que la découpe et l'arrangement des sprites des personnages et des ennemis. De plus, il a été désigné pour la rédaction du rapport au nom du groupe.
- Viet lui était chargé du contenu interactif, il a travaillé sur l'interactivité des menus, la création d'effets graphiques, la mise en place des systèmes de niveaux, la correction de nombreux bugs, et a aussi collaboré avec Damien sur l'aspect graphique pour l'animation des personnages et des tiles.
- Wilfried lui a eu un rôle polyvalent, il a contribué à divers aspects du projet. Il a corrigé des bugs, notamment ceux liés aux différents menus, et a amélioré l'interactivité. De plus, il a joué un rôle dans la création de nouvelles collisions et dans l'implémentation des ennemis avec leurs intelligences artificielles, pendant que Damien et Viet perfectionnaient le visuel du jeu.
- Jonny s'est spécialisé dans la mise en place des paramètres, notamment du menu et de la fonction de pause, Jonny a également apporté des correction pour les nombreux bug afin d'améliorer le jeu globalement

Comme vous l'aurez compris, cette répartition semi-structurée des rôles a favorisé une collaboration harmonieuse au sein de l'équipe. Chaque membre a apporté sa contribution spécifique pour compléter le projet, tout en restant disponible pour venir en aide aux autres membres. Cette approche a encouragé un échange constant d'idées et d'expertise, renforçant ainsi la qualité globale de notre réalisation.

DÉFI ET SOLUTIONS:

Pour ce genre de projet plusieurs défis se posent, donc présentons les de manière simple et chronologique tout d'abord, il était impératif de définir clairement ce que nous souhaitions réaliser et comment nous envisagions de le faire. Chacun d'entre nous avait des idées divergentes, nécessitant ainsi des réunions et concertations pour comprendre les limites de notre sujet et déterminer quels éléments nous souhaitions conserver pour le projet inspiré de Castlevania.

Concernant les problèmes spécifiques rencontrés, Damien a investi beaucoup de temps dans la recherche en ligne de ressources, passant des heures considérables à les découper une par une. Il estime avoir traité près de 300 images ! Pour les sons, trois versions ont été nécessaires, ajoutant une complexité qui par moments a limité sa disponibilité pour assister les autres membres sur d'autres aspects du projet.

Viet a été confronté à de nombreux bugs et a dû gérer la gestion globale des fichiers car certains dépôts ne fonctionnant pas de manière cohérente entre eux, ce qui entraînait des plantages du programme.

Wilfried a également fait face à un défi de taille en mettant du temps à la mise en place d'une intelligence artificielle pour les ennemis, les rendant capables de réagir aux obstacles en sautant, par exemple.

Un défi conséquent que nous avons relevé a été la création du menu avec tous ses éléments. Ce dernier a posé problème au niveau de la transmission des informations entre les différentes pages, demandant une attention particulière pour garantir son bon fonctionnement.

Globalement, nous avons réussi à surmonter ces défis, bien que l'un des obstacles externes que nous n'avions pas anticipé ait été le facteur temps. En effet, avec le projet coïncidant avec la période de rendu, certaines idées ont dû être mises de côté ou repensées, entraînant une perte de temps et de qualité sur le projet final. Malgré cela, nous sommes parvenus à livrer un projet fonctionnel.

CONCLUSION :

En conclusion, la programmation de notre jeu inspiré de Castlevania en utilisant JavaScript a constitué une expérience très enrichissante. Tout au long de la création du jeu, nous avons été confronté à plusieurs défis qui ont exigé de nous une réflexion approfondie pour parvenir à un résultat fonctionnel. La mise en place de la logique de déplacement, la gestion des différentes configurations du niveau, ainsi que l'implémentation des intelligence ennemis ont constitué des aspects particulièrement stimulants.

Malgré les difficultés rencontrées, nous avons su les surmonter grâce à une compréhension approfondie de notre code et une préparation minutieuse. Nous sommes donc fiers de voir le projet aboutir et cela a renforcé nos compétences en programmation, tout en nous permettant de mettre en pratique les concepts théoriques acquis au fil de mes cours.

Nous nous sommes développés personnellement car nous avons appris l'importance de la persévérance et de l'adaptabilité face aux défis du développement de jeux. mais aussi sur le plan de la gestion de projet car nous y avons vu l'importance de la planification, mais nous sommes ravis..