

Compte rendu Unity



Sommaire

1. Introduction.....	3
2. Conception du projet	3
3. Développement Graphique.....	3
4. Développement technique et fonctionnalités du jeu	3
4.1. Scripts :	3
4.3. Interaction et environnement :.....	4
4.4. Effets sonores et visuel :	5
6. Conclusion.....	5
Annexes :	6

1. Introduction

Dans le cadre de notre matière, un projet d'évaluation nous a été donné sur Unity. L'objectif de ce projet est de mettre en pratique les concepts fondamentaux abordés en cours de Programmation Multimédia, tout en explorant l'outil. Ce travail devait être réalisé en binôme.

2. Conception du projet

Pour l'idée du jeu de base, nous avons décidé de partir sur un aspect moderne en low-poly. Le scénario était inspiré des fameux cookies du Crous. Nous avons eu l'idée d'un personnage qui devait récupérer des cookies pour arriver au bâtiment final tout en affrontant des cafards du CROUS sur son chemin.

3. Développement Graphique

En ce qui concerne le développement graphique, nous avons employé de nombreux outils de Unity ou d'autres sites qui proposent des images en 3D ou des animations. Par exemple, le modèle du joueur a été pris sur le Unity Asset Store. Pour le rendu (shader), nous étions sous URP. En ce qui concerne l'organisation des fichiers, nous avons fait tout notre possible pour respecter l'architecture donnée en cours.

Pour le système de particules de pas, nous avons suivi un tutoriel sur YouTube de Bliz Studios. Le résultat est très subtil mais visible si on s'approche des pieds du joueur.

4. Développement technique et fonctionnalités du jeu

4.1. Scripts :

Nous utilisons 16 scripts au total qui contrôlent tous une partie :

- Attack : gestion de l'attaque du joueur envers les cafards.
- Chest : Gère l'ouverture du coffre seulement en cas de possession de clé, l'UI qui affiche la touche à appuyer pour ouvrir.
- damageEnemy : Gère la mort de l'ennemi une fois touché par l'arme.
- damagePlayer : Gère la prise de dégâts par le joueur, ainsi que sa barre de vie. Active l'animation de mort du joueur une fois que la vie est à 0 (à cause des cafards ou l'eau).
- enemyAI : Gère l'intelligence artificielle des cafards. Ils tournent en rond en attendant que le joueur, puis ils le poursuivent, mais si le joueur s'éloigne trop ils reviennent à leur emplacement de départ.
- GameManager : Gère le compte des Cookies (pièces). Mets à jour l'UI en fonction du nombre de pièces récupérées.

- HealthBar : Gère l'UI de la barre de vie du joueur. En interaction avec le script damagePlayer.
- KeyMove : Gère les déplacements du joueur en fonction de la caméra (pas tout à fait au point).
- MainMenu : Gère l'interaction avec les boutons sur le menu principal : charger la scène du jeu ou quitter celui-ci.
- objectSpin : Fonctionnement des objets qui tournent sur eux-mêmes.
- pickable : Gère la récupération des cookies et ajout au compte total, puis la disparition de celui-ci.
- pickupKey : Gère la récupération de la clé et l'apparition de celle-ci au-dessus de la tête du joueur. Mets le booléen de récupération de la clé à « Vrai » (utilisée par Chest)
- pickupSwat : Similaire à pickupKey, pour la tapette à mouche. Equipe l'arme dans la main du joueur.
- Teleporter : Gère la téléportation d'un point A vers un point B et inversement
- Win : Le script fait en sorte que dès que le joueur arrivé au trophée il gagne en faisant apparaître la scène WIN + le trophée qui tourne sur lui-même avec une rotation différente des pièces qui est géré par Object Spin.
- WinOrLoseMenu : Le script gère l'interaction sur les scènes de défaite (GameOver) ou de victoire (Win) pour quitter ou revenir au menu principal.

4.2. Scènes

Nous avons 4 scènes au total.

Menu : Le menu de départ. Affiche le logo du jeu (présent en page de garde), un bouton commencer et un bouton quitter. Un boutons Options était envisagé, mais il fut abandonné par la suite. Il pourrait également y avoir un bouton crédits.

Game : La scène du jeu principal. Contient tout ce qui fait le jeu : map, joueur, ennemis, etc...

GameOver : Ecran de défaite. Propose de recommencer une partie ou de retourner au menu principal.

Win : Ecran de Victoire. Propose de recommencer une partie ou de retourner au menu principal.

4.3. Interaction et environnement :

Le joueur peut se déplacer avec les flèches directionnelles ou les touches ZQSD (recommandé). Le joueur peut ramasser une clé en marchant dessus. A l'aide de cette clé, il peut ouvrir le coffre contenant l'arme en appuyant sur E. Une fois l'arme sortie du

coffre, il faut marcher dessus pour l'équiper, et ensuite appuyer sur F pour attaquer. Cela lui permettra d'éliminer les cafards.

Problème de collisions connus avec l'arme : lorsque l'on frappe un cafard, cela a pour effet de l'enfoncer dans le sol et d'enchaîner des dégâts sur le joueur, ce qui est un effet indésirable. Par manque de temps ceci n'est pas corrigé.

4.4. Effets sonores et visuel :

Pour les effets sonores, nous avons pris des sons connus. Par exemple, à la mort du joueur, le « Oh No » de Mario peut être entendu. Pour ce qui est des cookies (pièces) nous avons utilisé un son disponible dans le pack « BUT Project » fourni par Mme Vessiot. Lorsqu'on élimine un cafard, un bruit d'écrasement peut être entendu. Quand on tombe dans l'eau (zone mortelle du jeu), un bruit d'éclaboussement est joué.

Pour ce qui est des effets visuels, nous avons ajouté un système de particules lorsque le joueur marche en suivant un tutoriel YouTube (voir Annexes).

5. Fonctionnalités non implémentées

Contrôle manette : N'ayant pas de manette pour tester cette fonctionnalité, nous n'avons pas pu implémenter celle-ci. Elle devrait fonctionner pour les déplacements puisque le jeu utilise le système « Input Manager », mais pour ce qui est de l'interaction avec le coffre et l'arme, la compatibilité n'est pas assurée.

Bruits de pas du joueur : Dans le souci de vouloir faire fonctionner les particules visuelles pour les pas du joueur, nous avons omis de faire les bruits de pas. Ceux-ci sont normalement simples à implémenter.

Icône pour le score : nous n'avons pas vu cette contrainte.

La structure du projet est également assez désordonnée, par manque de temps nous n'avons pas fait le ménage.

6. Conclusion

Nous avons particulièrement apprécié ce projet, qui nous a permis de mettre en pratique les connaissances théoriques acquises en cours, tout en explorant des outils spécifiques comme Unity et des ressources annexes. La création d'un jeu vidéo, bien qu'exigeante, s'est avérée très enrichissante pour comprendre les différentes étapes de développement, que ce soit la conception, l'implémentation technique, ou encore l'aspect graphique et sonore.

Ce projet nous a également offert l'opportunité d'approfondir des domaines que nous n'avons que peu explorés auparavant, tels que l'animation et la 3D. Les difficultés rencontrées, comme la gestion des collisions ou l'adaptation au pipeline URP, nous ont

poussés à rechercher des solutions techniques innovantes, renforçant ainsi notre autonomie et notre capacité à résoudre des problèmes complexes.

En somme, ce projet nous a permis d'avoir un aperçu concret et pratique des multiples facettes du développement de jeux vidéo. Bien que certains éléments n'aient pu être implémentés faute de temps, nous avons acquis des compétences transférables et une meilleure compréhension de la collaboration au sein d'un projet numérique. Cela ouvre des perspectives intéressantes, que ce soit pour des projets futurs ou pour approfondir certains aspects techniques.

Annexes :

Animation pour le personnage :

<https://www.mixamo.com/>

Menu tuto YouTube:

https://www.youtube.com/watch?v=zc8ac_qUXQY

Canvas images :

<https://www.youtube.com/watch?v=poaPCgYdZKc>

Récupération des pièces :

https://www.youtube.com/watch?v=6iSJ_jh6Rdo

Asset 3D Trophée :

<https://assetstore.unity.com/packages/3d/props/3d-low-poly-trophy-and-cup-pack-created-with-fastmesh-asset-293357>

Asset 3D arme tapette à mouche :

<https://free3d.com/3d-model/fireswatter-v1--168317.html>

Vidéo tuto Arme :

https://www.youtube.com/watch?v=IYI_bP_Wjpg

Asset 3D coffre + animation :

<https://assetstore.unity.com/packages/3d/props/low-poly-chest-animated-247127>

Model 3D clef:

<https://www.turbosquid.com/3d-models/key-3d-2287880>

Particule Pas :

https://www.youtube.com/watch?v=mDr_HMOHmLs

Conversion URP :

<https://www.youtube.com/watch?v=E0Qnq1y1S20>

Player 3D model :

<https://assetstore.unity.com/packages/3d/characters/humanoids/fantasy/free-low-poly-human-rpg-character-219979>

Asset 3D Végétation :

<https://assetstore.unity.com/packages/3d/environments/landscapes/free-low-poly-nature-forest-205742>

Eau URP :

<https://assetstore.unity.com/packages/2d/textures-materials/water/simple-water-shader-urp-191449>