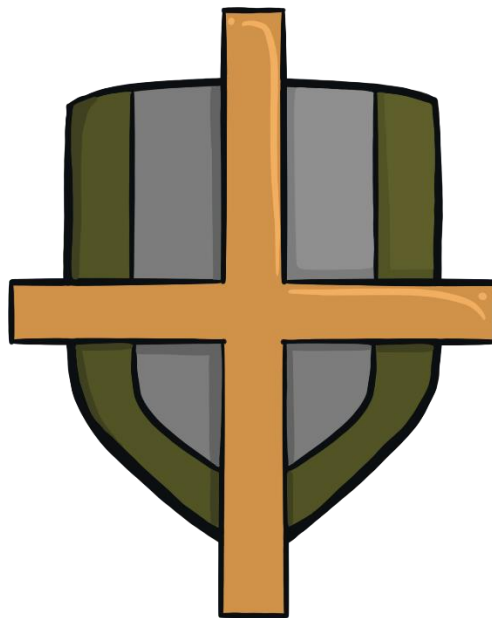


Rapport de Soutenance – 1.0

Unholy Crusades

Groupe 1 – Knight Unlimited

Mars 2024



BOURGUIN Axel
DARWICHEH Jana
DESMIER Katia (Chef d'équipe)
HALMOV Charles
LY Maxime

Table des matières

1.	Introduction	2
2.	Conception du Projet	3
2.1.	Répartition des tâches	3
2.2.	Personnages	3
2.3.	Niveaux	4
2.4.	Intelligence Artificielle	6
2.5.	Graphismes	6
2.5.1.	Design des niveaux	6
2.5.2.	Design des personnages	7
2.5.2.1.	Les héros	7
2.5.2.2.	Les monstres	8
2.5.3.	Design de l'environnement	9
2.5.4.	Design de l'interface	10
2.6.	Multijoueur	11
2.7.	Interface Utilisateur	12
2.8.	Site Web	16
2.9.	Son	17
2.10.	Jouabilité	18
3.	Avancement du Projet	21
3.1.	Comparaison avec le cahier des charges	21
3.2.	Personnages	21
3.3.	Niveaux	21
3.4.	Intelligence Artificielle	22
3.5.	Graphismes	22
3.6.	Multijoueur	22
3.7.	Interface Utilisateur	22
3.8.	Site Web	22
3.9.	Son	23
3.10.	Jouabilité	23
4.	Prévisions pour la Deuxième Soutenance	24
5.	Conclusion	24

1. Introduction

Unholy Crusades se positionne comme un jeu vidéo immersif et captivant qui plonge les joueurs au cœur d'un univers sombre et fantastique (d'où le nom du jeu). Mettant en œuvre les technologies C# et le moteur Godot Engine, ce jeu propose une expérience visuelle et interactive hors du commun. S'inspirant des jeux vidéo *dark fantasy* en 2D, nous avons décidé de développer un jeu du type Aventure/Thriller dans le but d'offrir une expérience riche en jouabilité et d'action. De plus, nous souhaitons cibler plusieurs plateformes pour notre jeu, dont Windows. Finalement, nous avons comme objectif de faire de notre jeu un monde ouvert avec des éléments de RPG, soit « jeu de rôle » en français.

Ce document a pour objectif de faire un compte-rendu de ce que nous avons réalisé à ce jour, présentant les avancements effectués de chaque membre du groupe Knight Unlimited sur le jeu Unholy Crusades.

2. Conception du Projet

2.1. Répartition des tâches

Nous avons divisé les tâches entre nos membres de la manière suivante :

	Axel	Jana	Katia	Charles	Maxime
Personnages	Assistant			Responsable	
Niveaux				Assistant	Responsable
IA	Assistant				Responsable
Graphismes		Assistant	Responsable		
Multijoueur	Responsable	Assistant			
IU		Responsable	Assistant		
Site Web	Assistant	Responsable			
Son			Responsable		Assistant
Jouabilité				Assistant	Responsable

FIGURE 1 – Tableau de répartition des tâches



2.2. Personnages

Pour ce qui est des personnages, lorsque le jeu commence, le joueur a le choix entre les deux principaux personnages en fonction du style de jeu qu'il préfère. Chaque personnage aura ses propres statistiques de combat. Les deux personnages sont conçus pour jouer en coopérations et doivent se compléter.

Chaque personnage est doté d'un kit de 2 sorts et d'un ultime. Ses capacités sont censées appuyer le style de jeu du personnage tout en étant amusant à utiliser.

Pour Eldric, en rapport à son histoire, joue un rôle de tank/bruiser et doit jouer en premier rang, il a donc des capacités en rapport avec son rôle, inspiré de Reinhard d'*Overwatch*, son rôle est de prendre de l'espace pour permettre à Matt de pouvoir faire des dégâts. Ses capacités lui permettent donc de prendre de l'espace en chargeant ou bien réduire les dégâts qu'il subit.

Pour ce qui est de Matt, étant donné que c'est un nain, l'inspiration de l'histoire des nains de *Warhammer* était obligatoire. Là où son changement d'arme entre une "forme" mêlée et à distance est inspiré de Jayce de *League of Legends*, sa bombe incendiaire et son cracheur de feu sont directement inspirés des unités de *drakes* de fer nain, utilisant l'effet de brûlure afin de maximiser leurs dégâts sur la durée. Son objectif comme dit précédemment est de maximiser les dégâts qu'il engendre grâce à ses capacités offensives tout en restant assez résistant grâce à son armure digne d'un vrai soldat nain.

Pour la future mise à jour du jeu, nous aimerions implémenter un système d'arbre de compétence avec de nouvelles capacités et augmentation de statistique déblocable au fur et à mesure que l'on avance dans le jeu. Cela permettra de personnaliser encore plus son style de jeu tout en restant en rapport avec l'histoire du personnage. Notre but est de rendre les personnages jouables les plus amusants possible tout en ne les rendant pas trop puissants afin qu'il y soit un réel intérêt à améliorer son équipement.

2.3. Niveaux

Les niveaux ont été créés en utilisant des tuiles, soit des carrés ayant des paternes permettant de représenter les éléments du décor, comme les sols, qui sont répétés et combinés pour ensuite créer un monde cohérent.

Ces tuiles sont assemblées d'une manière à respecter plusieurs objectifs : il faut que l'architecture ait une importance en termes de décor, c'est un objectif qu'il faut savoir parfois ignorer, considérant que le temps que nous avons pour la création des assets graphiques est bien trop limité, et aussi parce qu'il faut prioriser l'importance en termes de

jouabilité. Il faut donc réfléchir à l'avance à ces défis pour pouvoir ensuite produire les environnements propices.

Par exemple, prenons cette partie du niveau. En sachant qu'elle apparaît au tout début du jeu, on ne souhaite pas créer une situation trop complexe, en termes de demande au joueur ou même en termes d'architecture. Nous avons donc opté pour quelque chose de simple, qui demande au joueur de comprendre les méthodes de déplacement, pour qu'on puisse ensuite pousser un peu plus la difficulté, par exemple, dans cette autre partie du niveau, située à un endroit que le joueur devrait explorer un peu plus tard. Cette zone peut paraître intimidante, dû au fait que les plateformes sont très étroites. Cependant, les contrôles du jeu sont plutôt simples, et il n'y a aucune menace autre que le vide. Le joueur peut donc pleinement se concentrer sur l'atteinte des plateformes. De plus, la présence d'un double saut dans les mécaniques du jeu confère aux joueurs une marge d'erreurs extrêmement large.

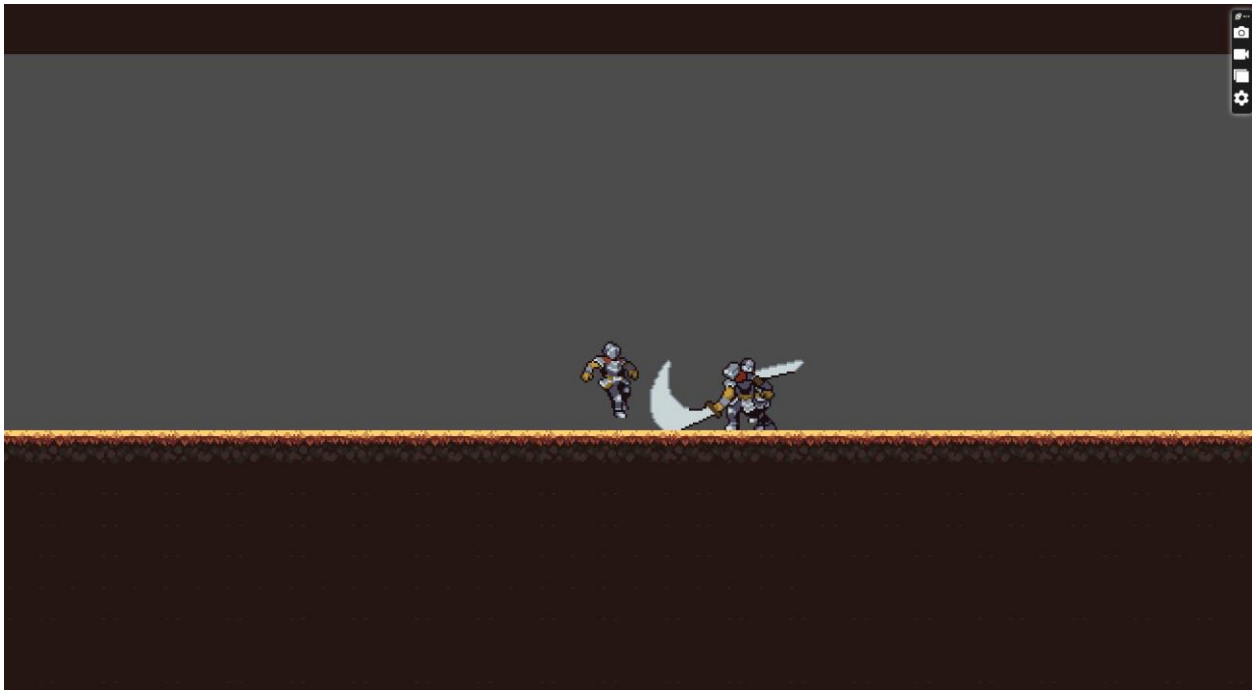


FIGURE 2 – Premier niveau créé

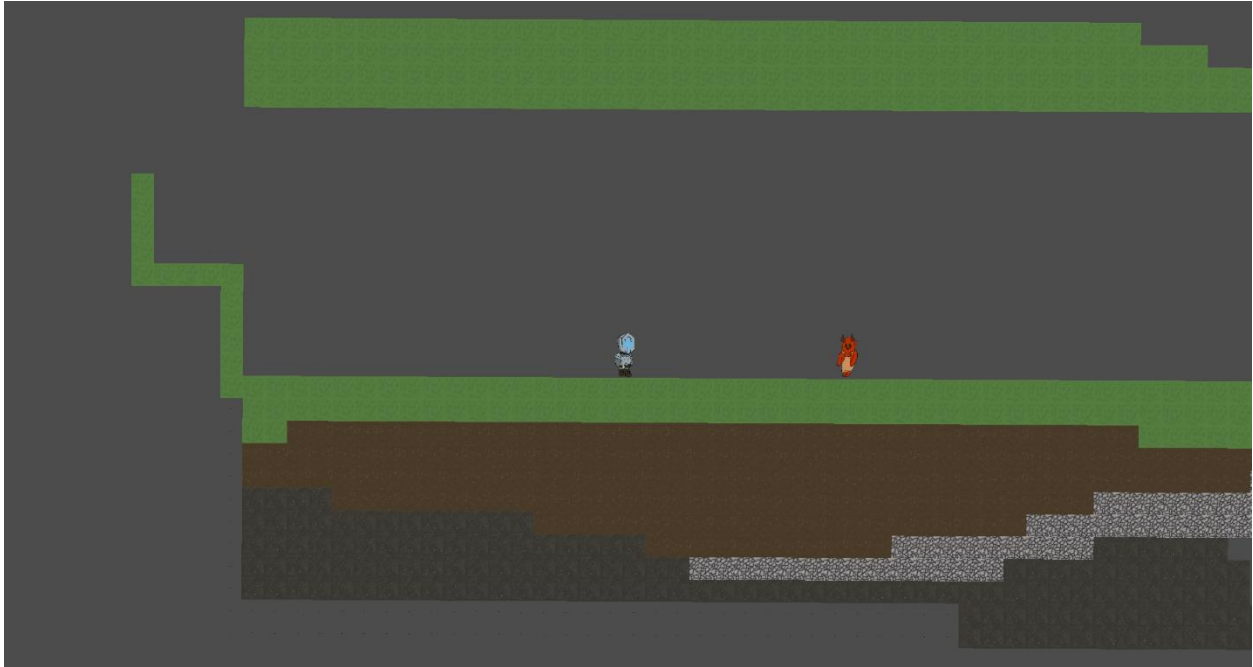


FIGURE 3 – Premier niveau avec les personnages et les tuiles conçus

2.4. Intelligence Artificielle

L'intelligence artificielle est en développement. Pour le moment, les ennemis sont capables de détecter et éviter les murs et les vides, à l'aide de nœuds RayCast 2D. Ils peuvent aussi détecter les joueurs, et les attaquer, tout en ignorant les autres ennemis.

2.5. Graphismes

2.5.1. Design des niveaux

Nous avons décidé d'utiliser une animation en 2D et l'usage d'une *tilemap* (notre carte sera composée de tuiles, et chaque tuile a une texture pour délimiter le parcours que le joueur devra emprunter). Pour le moment, il y a quatre types de tuiles, pour tout simplement délimiter les différentes parties de nos niveaux. L'un de notre objectif à court

terme est donc de terminer la création de l'ensemble des tuiles pour une immersion plus complète.



FIGURE 4 – Tuile « herbe »

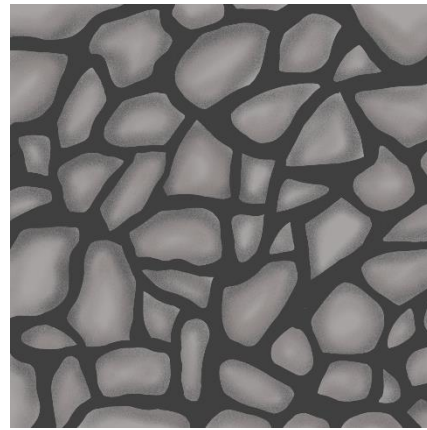


FIGURE 5 – Tuile « gravier »

2.5.2. Design des personnages

2.5.2.1. Les héros

Pour le design de Matt et d'Eldric nous avons opté pour un design simple à cause du nombre d'animations nécessaires aux bons fonctionnements des personnages. Ils garderont les mêmes habits et armes durant l'entièreté de cette aventure, pour une question de simplicité.



FIGURE 6 – Matt Lanvergue

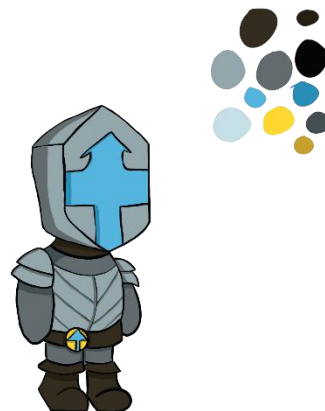


FIGURE 7 – Eldric Fierester

2.5.2.2. Les montres

Nous avons choisi de créer un design encore plus simpliste pour les ennemis pour garder en lumière les personnages principaux. On a donc opté pour une représentation simple mais efficace pour les adversaires. Leurs animations sont simples, tant dans les déplacements que pour les attaques. Ici, le démon courte portée se contente simplement de donner des coups de bâtons.



FIGURE 8 – Démon courte portée

Globalement le même design que pour les démons courtes portée mais avec un changement de nuances pour les différencier plus facilement. Leurs animations d'attaques varient aussi, le démon longue portée lance un caillou sur son adversaire.



FIGURE 9 – Démon longue portée

Nous avons choisi comme inspiration des chauves-souris que nous avons 'démonifier' pour potentiellement les réutiliser plus tard dans le niveau, une fois être sorti de la grotte. Son animation n'est pas encore terminée mais nous envisageons une animation simple pour l'attaque ou il crache sur son adversaire, et des déplacements de vols simples.



FIGURE 10 – Démon volant

En ce qui concerne le démon « boss », il n'est pour l'instant que sous état de croquis. Il a pour objectif d'apparaître beaucoup plus imposant que les démons normaux, tout en gardant une esthétique similaire. Nous envisageons toutefois des animations bien plus complexes que pour ses sbires, tant dans sa démarche que dans ses attaques.

2.5.3. Design de l'environnement

Pour débiter le développement de notre jeu, nous avons opté pour des tuiles simples (herbes, roche, etc.) et des éléments simples (arbres, boites, etc.), pour visualiser l'ambiance de notre jeu.



FIGURE 11 – Roche



FIGURE 12 – Arbre

Nous avons aussi mis en point des boules d'énergie que le joueur doit récupérer, son utilité va plus loin que simplement redonner de l'énergie au joueur, ses traces d'énergie seront disposées sur la carte afin de guider le joueur lors de son expérience.



FIGURE 13 – Boule d'énergie

2.5.4. Design de l'interface

Elle n'est pour l'instant pas implémentée mais elle a déjà été conçue. Nous souhaitons que notre interface reste simple pour que le joueur puisse se concentrer sur la jouabilité et de son aventure.

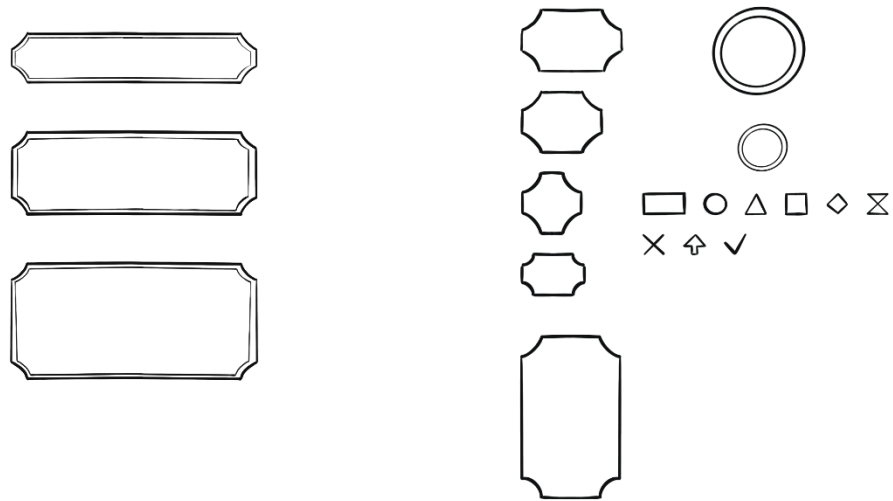


FIGURE 14 – Boutons et icônes

2.6. Multijoueur

La mise en place d'un mode multijoueur dans notre jeu n'est pas considérée comme notre principal problème à ce stade. Bien que le développement d'un mode multijoueur présente des défis uniques, nous avons choisi de concentrer notre attention sur d'autres aspects du jeu qui sont actuellement plus critiques pour offrir une expérience immersive et satisfaisante pour le joueur.

Nous avons mis l'accent sur des éléments tels que la conception des niveaux, l'équilibrage du gameplay et cela afin de garantir que le jeu fonctionne pour les joueurs. Ces aspects sont essentiels pour établir une base solide sur laquelle nous pourrions ensuite construire un mode multijoueur robuste et engageant.

Bien que le développement du mode multijoueur soit important pour nous, nous croyons qu'en consolidant d'abord les fondations du jeu en mode solo, nous serons en mesure d'aborder plus efficacement les défis

liés au mode multijoueur plus tard dans le processus de développement. Une fois que nous aurons atteint un niveau de satisfaction élevé avec le jeu en mode solo, nous nous concentrerons pleinement sur l'ajout d'une expérience multijoueur enrichissante pour nos joueurs.

2.7. Interface Utilisateur

Une IU de jeu est une interface qui permettra à l'utilisateur d'interagir avec elle pendant un jeu. Nous avons donc opté pour des éléments interactifs qui vont servir à l'utilisateur de personnaliser son jeu.

Nous avons mis en place le menu de départ qui permettra au joueur de commencer le jeu. Il est composé d'un titre intitulé Main Menu (« Menu Principal ») et de trois boutons : Play (« Jouer »), Options (« Réglages ») et Exit (« Quitter »).

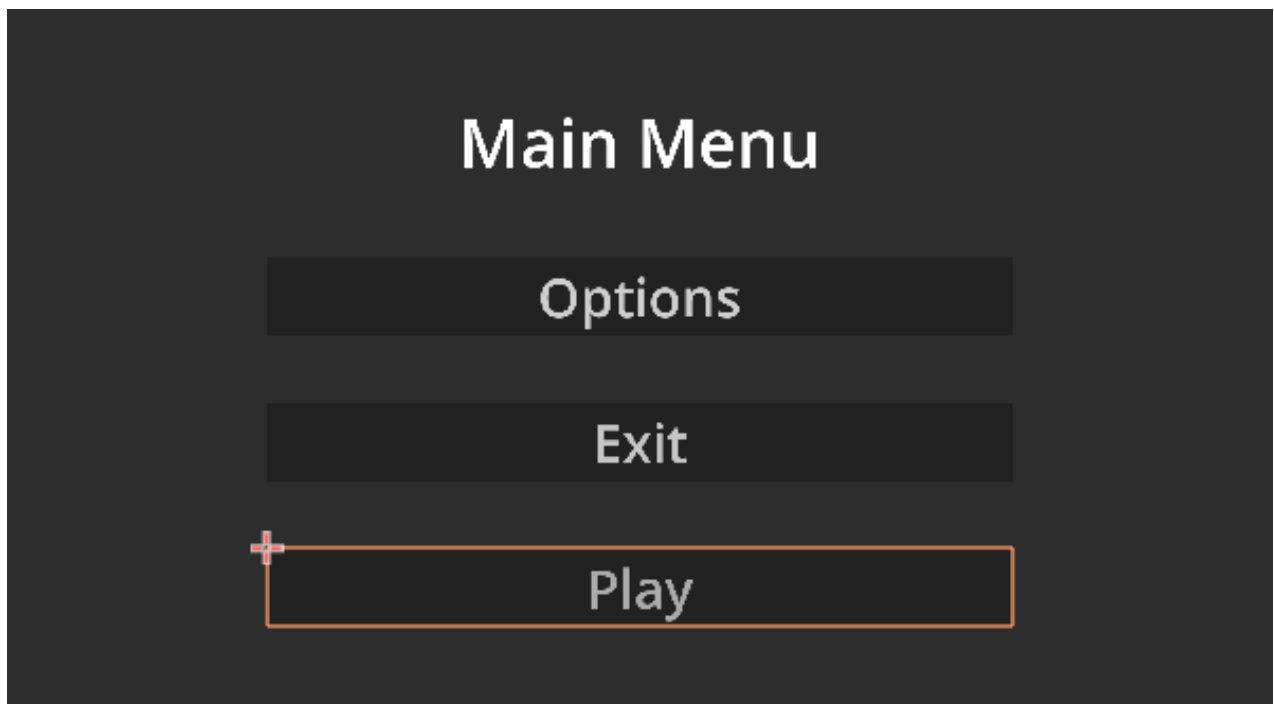


FIGURE 15 – Menu de départ

Le bouton Play permettra à l'utilisateur de commencer une partie et le bouton Exit permettra à ce dernier de quitter le jeu. Le bouton Options, quant à lui, ouvre un menu secondaire composé de réglages que le joueur pourra modifier à son goût.

Le menu secondaire est composé d'un titre intitulé Options (« Réglages ») et d'un bouton Back (« Retour »). De plus, on a un CheckButton (« Bouton Check »).

Le bouton Back permettra au joueur de retourner à l'arrière, au menu principal. Le CheckButton, quant à lui, permettra au joueur d'éteindre ou non le son.

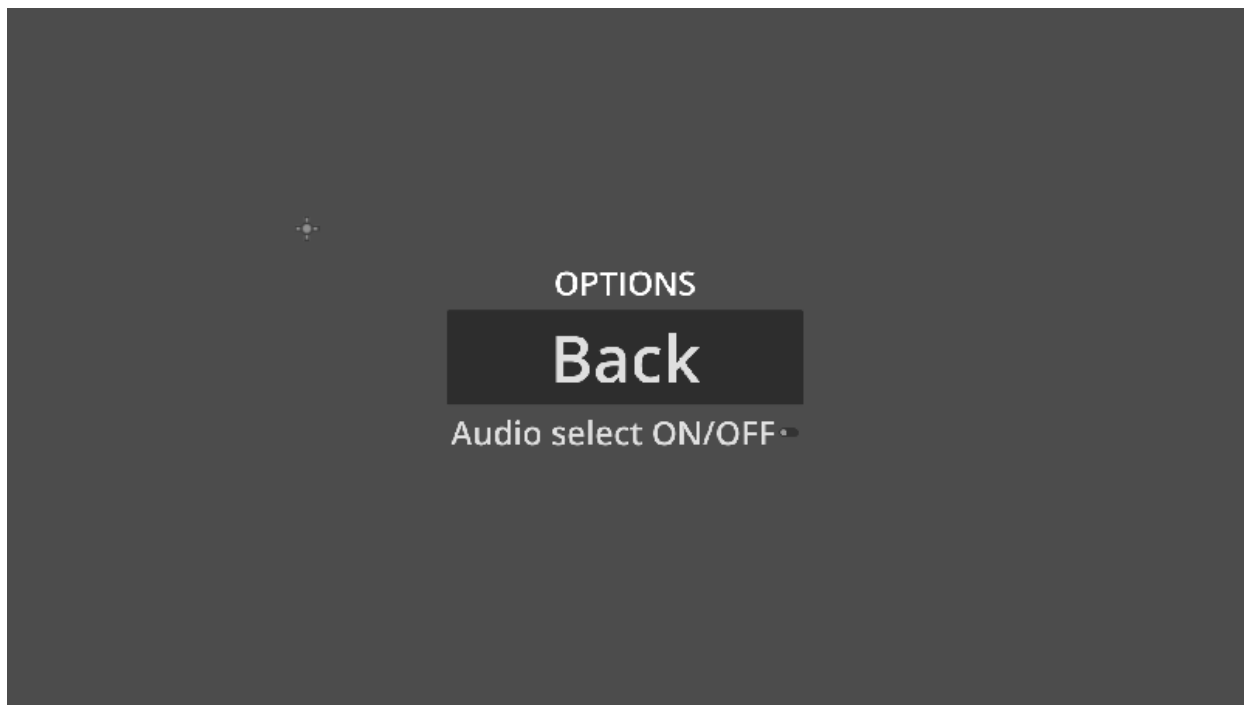


FIGURE 16 – Menu secondaire

Nous avons mis en place un code, dans le fichier « menu », qui permet aux différents paramètres de fonctionner et de se connecter entre eux.

```

using Godot;
using System;

public partial class menu : Control
{
    public override void _Ready()
    {

    }

    public override void _Process(double delta)
    {

    }

    private void _on_options_pressed()
    {
        GD.Print("Option button");
        GetTree().ChangeSceneToFile("res://options.tscn");
    }

    private void _on_exit_pressed()
    {
        GD.Print("Exit button");
        GetTree().Quit();
    }

    private void _on_play_pressed()
    {
        GD.Print("Play button");
        GetTree().ChangeSceneToFile("res://Scenes/Map/world1.tscn");
    }
    private void _on_texture_button_pressed()
    {
        GD.Print("Chat scene");
        GetTree().ChangeSceneToFile("res://chat.tscn");
    }

    private void _on_back_pressed()
    {
        GD.Print("Back button");
        GetTree().ChangeSceneToFile("res://menu.tscn");
    }

    private void _on_check_button_toggled(bool button_pressed)
    {

        var streamPlayer = GetNode<AudioStreamPlayer2D>("AudioStreamPlayer2D");
        streamPlayer.Stream = GD.Load<AudioStream>("res://ship-radar.wav");
        if(button_pressed)
        {
            GD.Print("CheckButton sound : ", button_pressed);
            streamPlayer.Play();
        }
        else
        {
            GD.Print("CheckButton sound : ", button_pressed);
            streamPlayer.Stop();
        }
    }
}

```

FIGURE 17 – Code C# du fichier « menu »

De plus, lorsque vous lancez le jeu, vous avez le choix entre les différents personnages depuis le menu de lancement. Puis lorsque vous mourrez, un menu se lancera pour vous laisser le choix entre relancer le jeu, ou bien quitter le jeu.

Pour de futures mises à jour du jeu, nous voudrions implémenter le menu Pause (« Pause ») directement dans la scène où le joueur se trouve. Quant au menu Game Over (« Fin du jeu »), nous aimerions que lorsque le joueur meurt, il soit indiqué de quelle façon le joueur est mort. Puis quand le jeu sera lancé pour deux personnes, il faudra que les menus soient eux aussi synchronisés pour les deux joueurs.

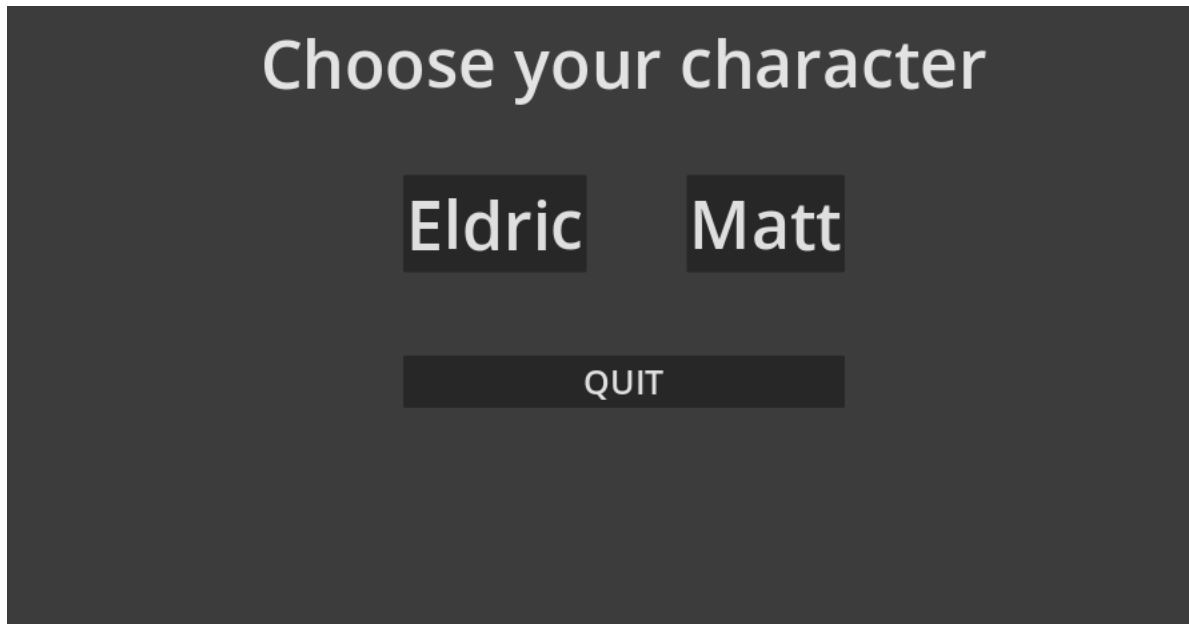


FIGURE 18 – Menu Choose Your Character (« Choisissez votre personnage »)

2.8. Site Web

Un site web doit être créé afin de présenter le projet ainsi que le jeu. Celui-ci contient une page d'accueil permettant d'accéder aux différents éléments de notre travail.

Pour la page d'accueil, nous nous sommes inspirés de l'univers de notre jeu pour établir ce fond d'écran ainsi que la police d'écriture. Nous avons aussi défini les différentes pages qu'on souhaite mettre dans notre site :

- **Accueil**, où il y a le nom du jeu ainsi qu'une bande sonore ;
- **Plus**, où il y aura toutes les informations sur le projet (l'historique, la chronologie de réalisation, les problèmes rencontrés, les solutions envisagées) ;
- **Dev**, où il y a la présentation des membres de l'équipe ;
- **Installation**, où il y aura un téléchargement du rapport et du jeu ;
- **Wiki**, où il y aura des annexes tels que des logiciels, images, sons, bibliothèques, applets utilisés pour le développement du jeu.



FIGURE 19 – Page Accueil

En ce qui concerne la page des membres d'équipe, nous avons opté pour un effet « overlay » pour montrer les brèves présentations de chaque membre d'équipe. Il s'agit de la superposition d'une couche sur une autre. Ici, nous avons la couche de présentation sur la couche de l'avatar du membre en question.

Nous avons choisi de faire du pixel art pour la présentation de nos avatars. Cela est due à la nature 2D du jeu.



FIGURE 20 – Page Dev avec effet « overlay » sur un des avatars

2.9. Son

La partie sonore de notre jeu n'a pas encore été pleinement développée principalement en raison de notre focalisation sur d'autres aspects prioritaires du jeu. Au cours du processus de développement, nous avons décidé de concentrer nos ressources et notre énergie sur des éléments tels que la mécanique de jeu, les graphismes.

Le son est certainement crucial pour l'expérience globale du jeu, mais nous avons jugé qu'il était plus important dans un premier temps de nous assurer que les mécaniques de jeu étaient solides et que les

graphismes étaient attrayants et reflètent pleinement le style graphique que nous voulions donner à notre jeu.

De plus, le développement sonore peut être une tâche complexe et nécessite une attention particulière pour créer des musiques et des effets sonores qui s'intègrent parfaitement à l'atmosphère du jeu. Nous voulons nous assurer de consacrer le temps nécessaire à cette tâche afin de produire un résultat de qualité qui enrichira véritablement l'expérience du joueur.

Bien que la partie sonore n'ait pas encore été pleinement développée, elle reste une priorité pour nous. Nous prévoyons de nous y consacrer une fois que nous aurons atteint un niveau de satisfaction élevé avec les autres aspects du jeu, afin de garantir une expérience de jeu complète et immersive pour nos joueurs.

2.10. Jouabilité

Pour la jouabilité, nous avons opté de nous orienter vers un style *metroidvania*, c'est-à-dire un genre de jeu qui combine des éléments de jeux de plateformes 2D et d'action « jeux de rôle ». Cela implique plusieurs choses : Comme tout jeu de plateforme, le jeu est en vue de côté 2D classique, et non pas en vue de dessus comme les vieux *final fantasy* et autres jeux de rôle classiques. Ensuite, les combats se jouent en temps réel, à l'opposé des jeux de rôle classiques qui se jouent au tour par tour. Les joueurs doivent donc gérer une notion d'espace dynamique pour pouvoir combattre. Cependant, la dimension stratégique des jeux de rôle est toujours présente dans une certaine mesure : Chaque entité "vivante" possède des statistiques de jeu de rôle, divisées pour l'instant en 4 : l'attaque, l'armure, les points de vie ainsi que la vitesse. Enfin, chaque personnage jouable possède plusieurs attaques : une attaque basique, qui consiste en un simple coup d'arme, ainsi que 2 attaques spéciales et une attaque ultime, qui sont des capacités puissantes qui définissent le style de jeu et l'individualité des personnages.

On va maintenant observer comment ces aspects de jouabilité ont été implémentés. Premièrement, on a l'aspect jeu de plateforme et action, cette partie n'abordant pas la création du niveau lui-même. Le joueur se déplace en utilisant des touches qu'il pourra définir lui-même. Ces touches servent à se déplacer de droite à gauche ou à sauter. Ceci est implémenté en réécrivant la fonction `__PhysicsProcess`, qui est une fonction qui va être appelée la plupart du temps 60 fois par secondes, ça dépend de la machine, et en lui faisant appeler la fonction `Input.GetVector()`, qui prend les noms des différents inputs et les assigne aux déplacements. Il faut savoir que les inputs pour se déplacer sur l'axe `y` ne sont pas utilisés. Lorsque cette fonction renvoie une direction qui n'est pas neutre, on applique au personnage une translation égale la direction multipliée par la vitesse du personnage. Sinon, on fait s'arrêter le personnage. Le personnage peut aussi sauter. On récupère son input en utilisant la fonction `Input.IsActionJustPressed` qui récupère la touche uniquement au moment où elle est appuyée, et au lieu de continuellement, et qui ajoute la valeur de saut à l'ordonnée du joueur. On gère ensuite la gravité à l'aide d'une constante qui sera continuellement ajoutée à l'ordonnée du vecteur pour faire retomber le joueur, car l'axe des ordonnées est inversé sur Godot. Il y a aussi la possibilité de faire un double saut, qui est gérée à l'aide d'un booléen stockant si oui ou non ce deuxième saut a été effectué, et si non, alors la valeur de l'ordonnée du vecteur devient la valeur de hauteur du deuxième saut. On ne se contente pas d'ajouter la valeur au vecteur car si le joueur était en chute libre, l'ordonnée de son vecteur serait positive et entrerait en conflit avec la valeur du deuxième saut, alors qu'on souhaite qu'un saut se produise dans toutes les situations.

Deuxièmement, on a l'aspect dynamique des combats. En effet, les ennemis et vous-mêmes seront constamment en déplacement dans cet environnement. Il faut donc pouvoir gérer la zone qui définit le corps des entités, c'est-à-dire les *hitbox*, ou masque de collision en français. Ces masques de collision dépendent en majorité de l'animation et de l'emplacement de l'entité. Il faut donc qu'ils suivent l'entité à laquelle ils sont assignés, ce qui se fait facilement en utilisant les nœuds `CollisionShape2D`, que l'on assigne en tant qu'enfant du nœud scène.

Chaque *frame* de chaque animation possédant une forme différente, on crée un masque par *frame*.

Troisièmement, nous allons parler de l'implémentation des capacités. L'implémentation des capacités est encore en ses débuts. Pour l'instant, nous avons créé des classes abstraites tout comme pour les équipements. Enfin, l'implémentation des statistiques de jeux de rôle se fait à l'aide de « composantes d'entité », qui possèdent des Dictionnaires de *stat* avec pour clé une énumération ayant le nom de chacune des *stat* et en valeur un objet héritier de la classe abstraite *stat* étant l'une des 4 statistiques que l'on a implémenté, ainsi que 2 autres Dictionnaires, l'un stockant les statistiques bonus issues des équipements et des capacités, et le dernier stockant la somme des valeurs des statistiques des deux dictionnaires.

3. Avancement du Projet

3.1. Comparaison avec le cahier des charges

Nous reprenons ci-dessous les prévisions faites lors du cahier des charges et on les compare avec l'avancement actuel.

	Prévisions	Avancement
Personnages	80%	60%
Niveaux	50%	50%
IA	20%	30%
Graphismes	0%	65%
Multijoueur	20%	0%
IU	30%	40%
Site Web	70%	70%
Son	30%	0%
Jouabilité	0%	50%

FIGURE 21 – Avancement du projet



3.2. Personnages

Nous souhaiterions développer d'avantage le côté « jeu de rôle » des personnages avec un vrai système de compétences déblocables et personnalisables afin que chaque joueur puisse avoir un kit de sort qu'il le correspond et qu'il préfère.

3.3. Niveaux

Le niveau devra être amélioré et contiendra des coffres et des autres objets à ramasser. Il devra être agrandi et des PNJ seront placés à certains endroits.

3.4. IA

Il y aura au moins deux nouvelles IA : Celle d'un monstre volant suivant le joueur, et celle d'un boss dont la conception est encore en réflexion.

3.5. Graphismes

Nos prochains objectifs quant aux graphismes sont : de finir la totalité des animations nécessaire au déroulement du jeu, et la mise en place d'une image de fond. De plus, nous souhaitons posséder une plus grande variété de tuiles pour la carte et avoir des environnement plus immersifs et divers.

3.6. Multijoueur

Un de nos membres de l'équipe, Axel, a pris la décision de se retirer de notre groupe. Cette décision a entraîné un impact significatif sur notre progression, notamment sur la partie concernant le développement du mode multijoueur. A cause de la précipitation de cette décision, nous n'avons malheureusement pas pu récupérer le code qu'Axel devait fournir.

3.7. IU

Nous allons concevoir un design pour l'interface utilisateur. De plus, nous allons travailler sur les paramètres présents sur le menu secondaire, afin qu'elles soient fonctionnelles (Audio).

3.8. Site Web

Nous allons mettre en place une version anglaise du site web ainsi que tout élément manquant du site (page Installation et page Wiki).

3.9. Son

Nous prévoyons d'avoir le minimum d'audio pour une expérience joueur agréable, mais cela reste une priorité mineure.

3.10. Jouabilité

Les capacités devront être implémentées. Les *hitbox* devront être retravaillées en « composantes d'entité » pour les rendre plus faciles à manipuler. Les *stats* seront aussi ajustées et de nouveaux équipements seront créés.

4. Prévisions pour la Deuxième Soutenance

Pour la deuxième soutenance, voici ce que nous avons prévu avec l'avancement qu'on a actuellement, avec le cahier des charges ainsi que les prévisions qu'on fait aujourd'hui.

	Etat actuel	Cahier des charges	Prévisions
Personnages	60%	80%	100%
Niveaux	50%	50%	100%
IA	30%	20%	100%
Graphismes	65%	0%	100%
Multijoueur	0%	20%	100%
IU	40%	30%	100%
Site Web	70%	70%	100%
Son	0%	30%	100%
Jouabilité	50%	0%	100%

FIGURE 22 – Prévisions pour la deuxième soutenance

5. Conclusion

Pour conclure, malgré quelques retards et le départ de l'un de nos membres, nous sommes satisfaits de l'avancement du projet. Nous avons pu atteindre, voire dépasser certains objectifs fixés lors de la répartition des tâches et du cahier des charges. La base du jeu est maintenant fixée, et nous pouvons nous consacrer pleinement au développement même de celui-ci. Nous allons maintenant redoubler d'efforts pour compléter le jeu au maximum.