

Zéphyr

Rapport de soutenance : 1^{re} soutenance

Mars 2019

Equipe Zelstrom :

Moustapha DIOP

Philippe HERNANDEZ - ALARCIA

Gabriel LION

Guillaume THOMAS



Table des matières

Introduction	3
1 Retour sur le cahier des charges	4
2 Retour sur le projet	6
2.1 Mécaniques de jeu	6
2.1.1 Les mécaniques du joueur	6
2.1.2 Les mécaniques de l'ennemi	7
2.1.3 La mécanique des armes	7
2.1.4 Mécanique générale	8
2.2 Interface	9
2.2.1 Menu principal	9
2.2.2 Menu de pause	10
2.2.3 En jeu	10
2.3 Multijoueur	11
2.4 Implémentation des ennemies	13
2.5 Graphismes des personnages	13
2.5.1 Le personnage	13
2.5.2 Les armes	14
2.6 Graphismes des cartes	14
2.7 Effets sonores	15
2.8 Site internet	16
3 La réalisation du projet et le ressenti de l'équipe	17
3.1 Moustapha	17
3.2 Philippe	18
3.3 Gabriel	19
3.4 Guillaume	20
Conclusion	22

Introduction

Ce premier rapport de soutenance de notre jeu Zéphyr permet de faire le point sur le projet tel que nous vous l'avons décrit dans notre cahier des charges.

Il détaille nos avancements mais aussi ce que l'on devra réaliser pour la soutenance intermédiaire. Ainsi, dans la première partie intitulée *Retour sur le cahier des charges* nous montrons les ajustements que nous avons opérés par rapport au projet initial ; ces ajustement concernent l'avancement des tâches et la distribution des rôles des membres de l'équipe de développement. La deuxième partie, intitulée *Retour sur le projet* permet à chacun d'expliquer, selon les différentes sections de conception du jeu, ce qu'il a réalisé, comment il l'a réalisé, mais aussi l'état d'avancement et les perspectives. La troisième partie, intitulée La réalisation du projet et le ressenti de l'équipe, présente une fiche de synthèse des impressions de chaque membre de ZELSTROM.

Chapitre 1

Retour sur le cahier des charges

Tableau de l'avancement issu du cahier des charges

Tâches	Soutenance 1	Soutenance 2	Soutenance 3
Mécanique de Gameplay	70%	95%	100%
Interface	25%	50%	100%
Multijoueur	70%	90%	100%
Intelligence Artificielle	60%	80%	100%
Graphisme des personnages	30%	60%	100%
Graphisme des cartes	40%	75%	100%
Effets sonores	20%	50%	100%
Site internet	50%	85%	100%

Dans son ensemble le cahier des charges a plutôt été respecté, mis à part la partie consacrée aux graphismes. Les pourcentages atteints sont assez proches, voire au-dessus de ce qui étaient attendus. Dans l'ensemble l'équipe de développement pense avoir respecté les délais, même si certaines priorités ont été accordées à d'autres éléments que ceux originellement annoncés dans le cahier des charges.

Tableau représentant l'avancement réel du projet

Tâches	Soutenance 1	Soutenance 2	Soutenance 3
Mécanique de Gameplay	70%	95%	100%
Interface	40%	60%	100%
Multijoueur	70%	90%	100%
Ennemis	40%	80%	100%
Graphisme des personnages	30%	60%	100%
Graphisme des cartes	30%	75%	100%
Effets sonores	25%	50%	100%
Site internet	30%	85%	100%

Tableau représentant la distribution des tâches décrite dans le cahier des charges

Tâches	Personnes			
	D. Moustapha	H. Philippe	L. Gabriel	T. Guillaume
Mécanique de Gameplay	S	-	R	-
Interface	-	R	-	S
Multijoueur	-	-	S	R
Ennemis	R	S	-	-
Graphisme des personnages	R	S	-	-
Graphisme des cartes	-	-	R	S
Effets sonores	-	R	S	-
Site internet	S	-	-	R

La distribution des tâches a été revue. Moustapha s'est beaucoup concentré sur la partie des mécaniques de jeu, au point d'en devenir responsable. Gabriel lui s'est penché sur de nombreuses parties, et commence à s'intéresser à la partie dédiées aux ennemis, c'est pourquoi il en devient le responsable.

Tableau représenant la nouvelle distribution des tâches

Tâches	Personnes			
	D. Moustapha	H. Philippe	L. Gabriel	T. Guillaume
Mécanique de Gameplay	R	-	S	-
Interface	-	R	S	S
Multijoueur	-	-	S	R
Ennemis	S	S	R	-
Graphisme des personnages	R	S	-	-
Graphisme des cartes	-	-	R	S
Effets sonores	-	R	S	-
Site internet	S	-	-	R

Chapitre 2

Retour sur le projet

2.1 Mécaniques de jeu

Il n'y a pas de retard dans cette partie qui repose essentiellement sur la création de scripts permettant le bon fonctionnement du jeu. Cependant, il nous reste encore à finaliser le positionnement exact des « game objects » et « components ».

2.1.1 Les mécaniques du joueur

Ce qui a été fait par Moustapha

Pour créer ces mécaniques nous avons utilisé l'éditeur de code Rider, nous nous sommes aidé de tutoriels [YouTube](#) faits par [Unity](#) ou [Brackeys](#). Cependant ces aides nous ne donnaient pas entière satisfaction et nous avons donc créé nos propres scripts.

Ainsi le joueur a quatre scripts qui lui sont directement liés.

- Le premier lui permet de se déplacer dans l'espace (avancer, reculer, aller à droite et à gauche) à l'aide des flèches ou des touches « w, s, a, d » et de pivoter sur lui-même grâce au curseur de la souris qu'il suit du regard.
- Le deuxième gère sa vie. Ainsi quand le joueur est attaqué il perd de la vie jusqu'à ce que celle-ci atteigne zéro. Il meurt et ses déplacements deviennent impossibles.
- Le troisième ne contient qu'une seule fonction qui permet simplement d'enlever des points de vie à l'ennemi passé en paramètre.
- Le dernier, un peu plus compliqué, permet au joueur de choisir son arme en utilisant la molette de la souris. L'arme choisie devient ainsi

opérationnelle alors que les autres sont désactivées. De plus le joueur ne peut qu'avoir au maximum trois armes.

Ce qu'il reste à faire

Il faut ajouter la possibilité aux joueurs de courir et de faire des roulades. De plus il serait intéressant de rajouter d'autres personnages allies pour avoir une plus grande diversité de jeu et avoir le choix.

2.1.2 Les mécaniques de l'ennemi

Ce qui a été fait par Moustapha

Nous avons utilisé les documentations présentes sur le site de UNITY pour comprendre comment fonctionnent certaines de leurs fonctions.

L'ennemi a un « component » special le « Nav Mesh Agent » qui lui permet de suivre un joueur et trois scripts qui lui sont liés.

- Le premier régit ses déplacements. Quand un joueur s'approche trop près de lui il se met automatiquement à le suivre jusqu'à ce que l'un des deux protagonistes meurt.
- Le deuxième concerne sa vie. Il fonctionne de la même façon que celui du joueur. Cependant lorsqu'il meurt, il s'arrête de se déplacer et est détruit au bout de deux secondes.
- Le troisième permet à l'ennemi d'attaquer. Ainsi quand il est suffisamment proche d'un joueur c'est à dire lorsque leur « collider » entre en contact il lui inflige des dégâts.

Ce qu'il reste à faire

De même il faut ajouter des nouveaux ennemis, plus puissants qui auront la capacité de tirer. Par ailleurs nous envisagerons que lorsqu'ils seront touchés des particules seront projetées.

2.1.3 La mécanique des armes

Ce qui a été fait par Moustapha

Ce sont elles qui permettent aux personnages de tirer, elles ont un rôle principal dans le jeu, et donc ont aussi besoin de scripts adaptés. Pour ce faire nous crée trois programmes principaux.

Chacune des armes détient un premier programme qui donne la possibilité soit de la ramasser en appuyant sur la touche « e » (quand un joueur est proche d'elle son collider le détecte et un message apparait « Press e to take »), soit de l'abandonner en appuyant sur la touche « f ». Cependant si le joueur possède déjà trois armes celle qu'il ramasse remplace celle qu'il a en main.

Au niveau du canon de chaque arme, un deuxième programme qui prend en paramètre le projectile a été implémenté. Ce programme permet, lorsqu'on appuie sur le clique gauche de la souris, le tir du projectile. Ce script gère aussi la cadence de tir, le nombre de balles tirées, leur direction et leur rapidité. Après sa projection et si elle n'a pas rencontré un obstacle la balle est automatiquement détruite au bout d'un moment qui varie selon les armes utilisées.

Pour finir chaque projectile possède un script qui contient les points de dégâts que la balle inflige lorsque cette dernière entre en contact avec le collider d'un ennemi. Elle appelle la fonction d'attaque du joueur associé avec comme paramètre l'ennemi touché. Par contre si la balle rencontre un élément du décor elle est détruite.

Ce qu'il reste à faire

Il faut aussi ajouter de nouvelles armes avec des capacités différentes. Il y aura peut être des modifications de la taille et de la forme des balles. Il faudra permettre au joueur de ramasser des munitions pour qu'il ne se retrouve pas à court de balles pendant la partie.

2.1.4 Mécanique générale

Ce qui a été fait par Moustapha

Ce sont des scripts qui ont un impact général sur le jeu. Pour cela nous sommes aidé de tutoriels faits par Unity. Il nous ont permis de créer d'une part un programme qui permet à la camera de suivre le joueur, et d'autre part de créer un script qui fait apparaitre des ennemis à des positions précises et à certain moment de la partie.

Ce qu'il reste à faire

Nous voudrions créer des coffres pour gagner par exemple d'autres armes ou des munitions pendant la partie.

2.2 Interface

L'interface évolue plus rapidement que ce qui était prévu. Le menu principal ainsi que le menu de pause sont proches de leur finalisation. L'équipe de développement constate une avance en comparaison de ce qui était prévu pour cette date.

2.2.1 Menu principal

Ce qui a été fait

Guillaume a créé le menu multijoueur permettant ainsi aux joueurs de créer un lobby du nom de leur choix ainsi que de se connecter à ceux de leurs amis. Cela a nécessité d'implémenter les boutons pour se connecter et créer un lobby ainsi que les scripts correspondants. Un menu de déconnexion a également été implémenté dans le cas où le joueur perd sa connection ou s'il y a un problème sur les serveurs. Une fois le problème résolu, il pourra alors facilement se reconnecter à son lobby. Cette implémentation a été relativement facile, aucun bug en particulier.

Le reste du menu principal a été réalisé par Philippe. Il est aussi à l'origine de l'esthétique et de la disposition du menu principal. Ainsi, en plus de la partie traitée par Guillaume, il y a, un bouton permettant de quitter le logiciel "Quit", un autre, d'accéder à un écran d'options avec de nouvelles interactions possibles "Options", et enfin un dernier permettant de démarrer le jeu en mode 1 joueur "Play Solo". Le bouton "Options" va changer l'interface et le joueur ne verra plus les boutons mentionnés précédemment. Il verra à la place de nouveaux éléments d'interface permettant les actions suivantes : la configuration de la fenêtre de l'application en plein écran, la gestion du volume, le choix de la qualité graphique souhaitée et enfin un bouton permettant de revenir à l'écran précédent et de sortir du menu des options "Back". La gestion du volume se fait à l'aide d'un curseur permettant de changer le volume sonore en lui affectant la valeur du curseur. L'affichage en plein écran se fait grâce à une forme d'interrupteur, il suffit de cliquer pour mettre le jeu en plein écran, et de cliquer de nouveau pour le mettre en mode fenêtré. Enfin, un menu déroulant a été mis en place pour permettre le choix de la qualité graphique souhaitée.

Ce qu'il reste à faire

Il est envisagé de retravailler le fond du menu principal, l'équipe trouve que ce dernier manque de "personnalité", au sens où il est trop générique, des

discussions sur la direction à prendre concernant l'esthétique de ce dernier sont en cours. Il s'agit du premier contact qu'à l'utilisateur avec le travail réalisé, l'équipe souhaite donc que l'esthétique soit à son goût et reflète davantage l'univers du jeu. On souhaite changer le fond et ajouter des effets de particules sur le menu. Des ajouts pourraient encore voir le jour au niveau des options, dont le choix de la résolution. D'autres changements pourraient également s'ajouter au cours du projet.

2.2.2 Menu de pause

Ce qui a été fait

Philippe s'est chargé de l'intégralité du menu de pause. Lorsque le joueur appuie sur la touche "échap" de son clavier, le jeu se met en pause, les couleurs deviennent plus sombres, et des boutons apparaissent. Ces derniers permettent actuellement de revenir au menu principal, de quitter le jeu, ou de reprendre le jeu, action qui est également possible en réappuyant sur "échap" une fois le jeu en pause.

Ce qu'il reste à faire

Il faut encore donner accès aux options depuis ce menu, pour le moment elles ne sont présentes que sur le menu principal. Mis à part cela, il n'est pas envisagé de revenir sur l'écran de pause.

2.2.3 En jeu

Ce qui a été fait

Gabriel est à l'origine de la barre de vie du joueur et de celles des ennemis. Deux styles différents ont été adoptés. Pour le joueur un cercle vide disposé aux pieds et entourant le personnage a été choisi. La création de cette interface a été permis grace a un tutoriel de Unity sur la gestion des sliders. Lorsque le personnage du joueur a toute sa vie, le cercle est complet et vert. Lorsque le personnage perd de la vie le cercle se vide dans le sens horaire, et plus il perd de la vie, plus la couleur du cercle tend vers le rouge. Cela a été réalisé a l'aide d'un slider et du script de la vie du personnage. Celui-ci a été modifié afin d'influer sur l'interface graphique de la vie du personnage. L'interface de la vie des ennemis est, quant à lui, représenté par une barre en 2 dimensions se situant au-dessus de sa tête. Cette implémentation a été effectuée grace à un tutoriel sur youtube, ainsi que les connaissances précédentes. Lorsqu'il a toute sa vie, la barre est verte et lorsqu'il perd de la vie, la

barre verte diminue à droite et à gauche et laisse apparaître une barre rouge sous celle-ci. Ainsi lorsque l'ennemi n'a plus de vie sa barre de vie est totalement rouge. La modification du script de la vie l'ennemi permet de remplir ou diminuer la vie de l'ennemi. Un autre script a été implémenté permettant de garder la barre face à la caméra.

Une petite carte (minimap) a également été réalisée par Philippe. Pour cela, une caméra a été placée en haut du personnage, et montre en vue de dessus l'environnement du personnage. Enfin pour réaliser la carte, un carré a été ajouté à l'écran de jeu, et sur ce dernier a été superposé le rendu de ce que capturerait la caméra.

Ce qu'il reste à faire

Il reste encore de nombreux éléments à rajouter dans cette partie, mais ces derniers viendront au fur et à mesure que nos idées se feront de plus en plus précises sur le rendu final de notre jeu. Ainsi, pour le moment, il est prévu de rajouter des informations sur le nombre de munitions restantes dans les armes, cela fait partie des priorités de ce qu'il faut encore rajouter dans l'interface, puisque sans ça, le joueur manque d'informations qui lui sont très importantes pour jouer. Les différents collectables acquis devront également pouvoir être vus par le joueur, la aussi il s'agit d'information qui devraient être facilement accessibles par le joueur.

2.3 Multijoueur

L'implémentaion du mode multijoueur est dans les temps. Le mode multijoueur est développé en même temps que le reste du gameplay. Il faut donc logiquement implémenter régulièrement les éléments nouveaux. Ainsi les scripts d'interaction avec les ennemis ne sont pas encore totalement à jour.

Ce qui a été fait

Après que les lobbies soient créés, le joueur a besoin d'avoir son personnage qui apparaît sur sa carte. Même si cela paraît simple, cela est loin d'être le cas. En effet, les premiers scripts que nous avons réalisé considéraient que le personnage était déjà sur la carte au début de la partie. Or dans le mode multijoueur, il ne peut en être ainsi. Le personnage a besoin d'être ajouté après que la carte soit chargée. Cela permet d'une part, d'éviter tout problème

de synchronisation avec les autres joueurs et d'autre part d'avoir plusieurs personnages, ce qui est pratique pour un mode multijoueur...

Mais avant que le joueur soit ajouté, des scripts appelaient ce dernier. Ce qui provoquait logiquement des erreurs... Ainsi il a fallu trouver une solution face à ce problème. Celle retenue fut de changer la conception de la plupart des scripts. Ainsi il fallait à présent que ces scripts commencent par une fonction `Start()` au lieu d'une fonction `Awake()`. Ce qui laisse donc le temps au personnage d'apparaître.

Après cela, un problème subsistait. Les scripts ne trouvaient plus le personnage. En effet, ce dernier n'est pas mis lui-même sur la carte. C'est une copie du personnage qui est ajouté. Ainsi on ne peut donc faire référence au personnage directement. Il faut alors le chercher dans la scène pour après pouvoir le manipuler.

Après cela, certes le mode multijoueur a son personnage. Mais le mode solo n'en a plus. En effet, le personnage n'est plus sur la scène d'origine car il était nécessaire de l'ajouter au début de la partie. Il fallait donc également le faire apparaître dans ce mode au début de la partie.

Simple me direz-vous ?

Toujours pas... Pour ajouter notre personnage dans le mode solo, un script sur le menu principale restait de scène en scène et ajoutait le personnage lorsque la scène de jeu était chargée. Ce qui avait pour effet d'égaleme nt ajouté un personnage dans le mode multijoueur. Donc 2 personnages contrôlés par un seul joueur. Pour y remédier, le script pour le personnage dans le mode multijoueur et dans le mode solo ont été unifiés. Et une séparation plus claire entre le multi et solo a été faite.

Ainsi de nombreux scripts ont dû être modifiés et adaptés pour être compatible avec le mode multijoueur.

Ce qu'il reste à faire

La position des joueurs est donc parfaitement synchronisée. Mais ce n'est pas encore le cas de leur orientation. Ainsi si un joueur tourne sur lui-même, les autres joueurs ne le voient pas.

Le ramassage des armes et les tirs ne sont pas non plus implémentés dans le mode multijoueur. Ainsi une même arme peut être ramassée par plusieurs joueurs. Et si un joueur veut tirer, alors tous les personnages se mettent à tirer.

De plus, comme indiqué plus haut, il y a un certain nombre de problèmes avec la synchronisation des ennemis, notamment au niveau de la cible qu'ils suivent. Ce qui implique que leur position n'est pas la même d'un joueur à l'autre.

Il reste donc ces quatre éléments à implémenter dans le mode multijoueur, plus les éventuels ajouts d'éléments futures.

2.4 Implémentation des ennemies

Nous avons pris un peu de retard sur ce que nous pensions faire, car la première création d'un objet et de ses caractéristiques associées demande un temps de conception important qui ensuite diminue rapidement pour les créations suivantes de même ordre.

Ce qui a été fait par Moustapha

Pour le moment nous n'avons qu'un seul ennemi.

Puisque nous ne savons pas modéliser des personnages en trois dimensions nous avons décidé de récupérer l'ennemi (le personnage zombie) dans l'asset store de Unity pour avoir un meilleur rendu. Cet ennemi est animé lors de ses déplacements et de sa position au repos. C'est un individu très basique, quand il repère un joueur il se dirige vers lui et l'attaque à main nue s'il est assez proche. Il sera utile pour faire les premiers niveaux du jeu.

Ce qu'il reste à faire

Nous voulons avoir plusieurs autres ennemis avec des compétences différentes (tirs, vitesse de déplacement, cachettes, guet-apens...) et de nouvelles animations.

2.5 Graphismes des personnages

2.5.1 Le personnage

Ce qui a été fait par Moustapha

Comme pour l'ennemi nous avons récupéré ce protagoniste sur l'asset store de Unity puisqu'il présente des avantages pour être animé. Ce personnage nous servira de référence pour la création des prochains. Il peut tirer, avoir plusieurs armes et en changer.

Ce qu'il reste à faire

Nous voudrions que notre personnage puisse courir et faire des roulades. On devra ajouter des animations (déplacements, roulades, position de repos)

pour rendre la chose plus réelle et immersive.

2.5.2 Les armes

Ce qui a été fait par Moustapha

Nous avons trois armes opérationnelles (un fusil a pompe, un fusil d’assaut et un pistolet). Ces trois armes ont des capacités différentes (nombre de balles tirées a la fois, vitesse de la balle, cadence de tir).

Ce qu’il reste à faire

Nous voudrions rajouter au moins deux armes et peut être en modifier certaines.

2.6 Graphismes des cartes

La création des carte s’est avérée plus difficile que prévue, étant donné le peu de ressources que nous avons trouvés, la quantité de code a fournir ainsi que sa complexité. Ces facteurs ont quelque peu entravé l’avancement des cartes.

Ce qui a été fait par Gabriel

Tout d’abord, nous voulions générer nous mêmes nos cartes de façon procédural à partir de salles que nous faisons nous mêmes. Nous avons commencé la création de ce dernier. Or la complexité du script est élevé. Le sujet étant peu renseigné il était difficile de s’informer afin de procéder nous même a cette génération. En effet il faut vérifier que les portes de chaque bâtiment soient bien l’une en face de l’autre, sinon il faut faire tourner la salle dans le bon sens pour qu’elles soient en face, ainsi que les salles et les couloirs ne se superposent pas.

Nous n’avons pas réussi a créer ce système permettant de générer lui même ses propres cartes de façon aléatoire a partir d’un jeu de salles déjà créé. Mais nous avons trouvé un Asset de Unity permettant de générer des cartes. Cela nous a paru être une alternative intéressante. En effet cet asset peut générer des cartes avec des salles de tailles aléatoires sur une surface et un nombre de salle définie. La modification des scripts de l’asset a permis de l’utiliser pour notre projet. Sa modification nous a permis de créer des cartes de tailles proportionnées par rapport à nos personnages et ennemis,

avoir des colliders sur les murs ainsi que nos balles ne passent pas a travers ces derniers.

Ce qu'il reste à faire

Si l'on garde l'asset pour générer des cartes il faudra encore le modifier afin de rajouter des portes qui s'ouvrent et se ferment, ainsi que le fait que nos personnages et les monstres soient visibles derrière les murs lorsqu'ils sont cachés par ces derniers. Il faudra par la suite rajouter des textures.

2.7 Effets sonores

La partie dédiée aux effets sonores avance très vite. Les fonctions permettant d'activer la musique risque d'être retouchées, mais peu de changements sont attendus sur le plan de la programmation pour ce qui est du son.

Ce qui a été fait

La partie effets sonores a été traité exclusivement par Philippe. La musique implémentée au jeu est présente à la fois dans le menu principal ainsi que dans le jeu. Cette dernière est gérée de manière assez simple. Une classe "Sound" a été créée répertoriant les différentes musiques et effets sonores incorporés dans le jeu. Chacun des éléments de cette classe vient avec différents attributs, tel qu'un nom ou une hauteur (la hauteur de la musique) notamment. Un script a ensuite été créé, ce dernier fait appel à une liste d'éléments issus de la classe "Sound" réunis dans une liste dont on se sert à de nombreuses reprises. Ainsi, dès que le script est appelé, dépendant de la scène dans laquelle se trouve actuellement le joueur, c'est à dire dans l'état actuel du jeu, s'il se trouve dans le menu principal, ou en jeu, il appellera la musique du menu principal ou bien celle du jeu, et ce grâce à une fonction permettant de lancer une piste audio.

En plus de la musique, des effets sonores ont été introduits dans l'application. Le but de ces derniers est d'immerger le joueur dans le jeu. Pour le moment, seul un effet sonore a été ajouté dans le jeu. Il s'agit de la piste audio jouée lorsque le joueur clique sur un bouton. Il fait appel au même script que pour la musique. Quand on appuie sur un bouton, une fonction du script s'exécute et joue le morceau souhaité. Cet effet sonore a été réalisé à l'aide d'un logiciel appelé BOSCA CEOIL permettant de créer des morceaux. Il est très probable que d'autres effets sonores soient encore réalisés à l'aide de ce logiciel.

Ce qu'il reste à faire

Comme dit précédemment, il ne devrait plus y avoir beaucoup de changements par rapport aux fonctions qui s'occupent de gérer le son. Ainsi, il devrait surtout s'agir d'ajouts de nouvelles pistes audio, que ce soient des musiques ou des effets sonores. Comme exemples d'effets sonores à ajouter, on a le son que fait l'arme lorsqu'elle tire un projectile ou des effets qui s'activent lorsque le joueur prend des dégâts.

2.8 Site internet

L'avancement du site web a pris un peu de retard. Notre objectif d'avoir fini 50% de sa réalisation pour la première soutenance s'est avérée un peu trop ambitieux.

Ce qui a été fait

L'hébergement du site web et l'achat du nom de domaine pour ce dernier sont tout de même déjà réalisés. Sur la page d'accueil de Zelstrom.me, vous pouvez retrouver deux liens : un qui redirigent vers la page française et l'autre vers la page anglaise. Ces dernières ne sont pour l'instant pas très fournies... Toutefois le contenu qui ira sur ses pages est déjà partiellement créé. Ainsi les pages décrivant le gameplay, le scénario, notre histoire et qui nous sommes sont déjà écrites.

Ce qu'il reste à faire

Logiquement, il reste donc ces éléments à ajouter sur le site. Et la création d'une page avec tous les documents concernant le projet est également à réaliser.

Chapitre 3

La réalisation du projet et le ressenti de l'équipe

3.1 Moustapha

Depuis l'écriture du cahier des charges et la division des tâches avec mes coéquipiers, je me suis lancé dans la réalisation de mon travail qui porte sur la mécanique du jeu, le graphisme des personnages et les ennemis. Ce travail a été pour moi une véritable découverte, car je n'imaginais pas tout ce qui était nécessaire à concevoir pour faire vivre un jeu.

Globalement les nombreuses heures passées à la mise en place des éléments nécessaires au fonctionnement des étapes et processus nécessaires à la bonne marche de Zéphyr, ont été pour moi des moments passionnants et les dernières semaines qui restent avant la réalisation finale seront pour moi captivantes, car l'enjeu d'arriver à présenter une production convenable me passionne. Cependant quelques doutes subsistent et je ne sais encore comment les atténuer.

L'intérêt pour ce travail vient d'abord du fait que lorsque je mets en place les différents scripts, je suis complètement immergé dans cette activité et je prends beaucoup de plaisir à réfléchir comment trouver une méthode pour arriver à ce que je veux produire. Par exemple pour le script qui permet de récupérer, jeter, ramasser les armes, j'avais créé au début un programme qui utilisait des listes. Je pensais mettre les armes dans une liste et à partir de cette liste les changer avec la molette et ensuite, le script devait chercher dans la liste la bonne arme pour la rendre opérationnelle. Aussi, quand on voulait jeter l'arme, le script devait récupérer l'arme de la liste et la supprimer de cette liste. J'ai expérimenté plusieurs fois ce programme, mais je me suis

rendu compte que marginalement ce programme ne fonctionnait pas à 100%, peut être était ce un problème d'index, mais je ne sais pas trop. J'ai du alors repenser entièrement le programme en accrochant les armes à un game object enfant du joueur.

L'intérêt vient aussi du fait que j'ai beaucoup apprécié de faire des recherches sur Internet pour trouver des solutions aux problèmes rencontrés. Ainsi, j'ai pu connaître et comprendre comment fonctionnent certains composants comme le collider, certaines fonctions comme awake, start, update, OnTriggerEnter, ou certains scripts comme ceux concernant le déplacement des personnages.

Quand on réussit à faire ce que l'on veut, on a un véritable sentiment de fierté d'autant plus que l'on a trouvé par nous même la solution sans aucune aide extérieure si ce n'est les cours d'algorithme et les différents entraînements de programmation réalisés depuis le début de cette année à EPITA.

Malgré tout, des doutes et des incertitudes subsistent.

D'abord, la réalisation des graphismes me pose quelques soucis, car réaliser un personnage en 3D est pour moi difficile. Je ne suis pas un expert en dessin, mais par contre je ne désire pas que les personnages prennent des formes géométriques, comme des cubes ou parallélépipèdes... J'aimerais donner à chacun d'entre eux un visage et une corpulence différente. Cet objectif me pose des problèmes de réalisation technique, mais aussi de création, car cela nécessite de dessiner de très nombreux détails, d'utiliser de très nombreuses nuances de couleurs et de formes. Ce travail paraît nécessiter une dépense considérable de temps et d'énergie pour un résultat qui ne sera pas forcément à la hauteur de mes espoirs vu mes compétences limitées en dessin. Mais je ne perds pas courage et vais trouver une solution peut être en demandant l'avis de mes camarades.

3.2 Philippe

Le projet a jusqu'à maintenant été divertissant et enrichissant. Divertissant car une certaine fierté pouvait se dégager dès qu'un élément fonctionnait comme il était voulu après de nombreux efforts, et même dans la compréhension globale du logiciel qu'est Unity. Par exemple, en travaillant sur la petite carte (minimap), un obstacle a été de surmonter une erreur réalisée très tôt. En effet, cette partie avait été réalisée pendant le début du projet et la carte au lieu de rester statique sur l'écran comme il était convenu suivait les déplacements de la souris. Cela était dû au fait que cette dernière était liée au joueur alors qu'elle n'aurait pas dû l'être, et il a fallu attendre la

semaine précédant la première soutenance pour que ce problème soit résolu. Il ne s'agissait là que d'une légère erreur, mais comme cette carte avait été faite uniquement en suivant un tutoriel fait par la chaîne Youtube Brackeys, il était alors difficile de la comprendre. C'est donc en s'investissant dans le projet que la compréhension du logiciel et de ce qui est attendu s'améliore.

Concernant la partie du son, une véritable déception a vu le jour avec le logiciel BOSCA CEOIL, ce logiciel permettant de faire de la musique avait été téléchargé dans un premier temps pour essayer de réaliser les musiques du jeu. Malgré le temps investi sur le logiciel, les résultats obtenus restaient insatisfaisants. C'est pour cette raison que les musiques choisies pour le projet sont des musiques libres de droits offrant un bien meilleur rendu que ce qui a été obtenu en essayant de manipuler le logiciel. Le logiciel reste tout de même très intuitif et simple d'utilisation, c'est pourquoi il est encore utilisé dans la réalisation de certains effets sonores.

Il est à noter que Brackeys a été une source très importante dans le travail réalisé grâce à ses nombreux tutoriels.

3.3 Gabriel

Tout d'abord, je pense que la réalisation de ce projet en première année est une très bonne chose car cela nous permet d'avoir un premier contact entre les études et le monde réel du travail. C'est pourquoi je trouve ce projet très enrichissant.

Une variable souvent oubliée, est le temps. En effet nous savions tous qu'il fallait s'y mettre au plus vite pour ne pas se laisser déborder comme l'ont dit d'autres groupes l'année passée.

C'est pourquoi je me suis intéressé à beaucoup d'éléments de Unity qui est un monde très vaste. J'ai suivi quelques gros tutoriels en entier traitant de parties importantes du développement d'un jeu vidéo. Cela m'a permis d'acquérir une maîtrise dans le logiciel et sur des éléments très variés, allant des animations des particules, à l'interface de vie des personnages et des ennemis ainsi qu'à leur contrôle en passant par la génération aléatoire de cartes.

Je suis très satisfait de la cohésion du groupe, nous avons pu nous rapprocher d'avantage tout en faisant l'acquisition de nouvelles connaissances. En premier, je me suis penché sur la partie de la Mécanique De Gameplay à l'aide de tutoriels Unity, or voyant que Moustapha avançait à bonne allure

et plus vite que moi, je décidais alors de lui donner mon rôle de Responsable. C'est alors que je me suis orienté vers Les Cartes et L'interface.

J'ai trouvé très enrichissant le fait de chercher des informations ainsi que d'essayer de trouver par soi-même des solutions, c'est pourquoi j'ai beaucoup aimé faire l'interface des vies du joueur et des ennemis, que j'ai essayé de rendre assez original.

Puis est venu le problème des cartes, la complexité de l'algorithme afin de générer nos propres cartes procéduralement à partir de salles préexistantes fut très compliqué, mais à force de chercher j'ai trouvé des solutions que j'ai pu adapter à notre jeu. J'ai repris un générateur de carte que j'ai modifié par la suite. Tout n'est pas au point mais c'est très encourageant !

Tout cela m'a permis d'être assez polyvalent et j'ai pu aider un peu tous les membres du groupe sur diverses tâches. Ainsi je me suis beaucoup impliqué dans le projet et en plus des connaissances que j'ai acquises, j'ai développé une méthode de travail et d'apprentissage qui est indispensable à tout bon ingénieur. J'ai trouvé cet exercice très enrichissant et je suis motivé à continuer sur ma lancée dans le développement de Zéphyr, afin d'arriver à un résultat proche de nos attentes et qui satisfait toutes l'équipe.

3.4 Guillaume

Au tout début du projet l'excitation de ce que nous allions réaliser me motivait beaucoup. Alors que jusqu'ici nos travaux de programmation n'étaient pas très palpitant. En effet, ceux-ci sont enrichissant sur le plan technique mais le rendu final n'était jamais impressionnant visuellement parlant (à part le jeu de la vie qui m'a beaucoup plus à coder, soit-disant passant). Alors qu'ici, il s'agit de la création d'un jeu vidéo de bout en bout, avec un vrai challenge de programmation et un résultat final qui nous corresponde. Je trouvais ça incroyable (et je n'étais pas le seul).

Mais cette motivation est rapidement partie...

En effet, ne maîtrisant pas le logiciel, le travail se limitait à suivre des tutos et appliquer aveuglément ce qui était dit... Pas très amusant d'une part et pas très enrichissant non plus... (A part pour développer la persévérance peut-être...) C'était plutôt très ennuyant. Et dans cette période, je commençais à sérieusement remettre en question l'intérêt de ce projet. Mais il fallait continuer avancer de toute manière d'une part pour ne pas abandonner mon groupe et d'autre part parce que ce projet comptait dans la moyenne (ce qui est non négligeable).

Mais au bout d'un mois environ, cela avait changé du tout au tout.

Après un certain temps, on est obligé d'adapter ce qu'on a vu dans les

tutoriels à notre projet, car ceux-ci ne correspondent jamais à 100% avec ce que l'on souhaite réaliser. Et c'est là que quelques choses de totalement nouveau c'est passé dans ce projet, j'ai pu réfléchir ! Ca peut paraître stupide dit comme ça mais ça a vraiment tout changé. Quand je travaillais sur Zéphyr ce n'était plus uniquement dans le but d'obtenir une bonne note mais réellement parce que ça m'amusait tout simplement. Je passais ainsi jusqu'à plusieurs heures dessus certains jours et j'avais vraiment l'impression d'avancer et par dessus tout d'avoir le contrôle sur ce que je faisais.

Conclusion

L'ensemble de l'équipe Zelstrom continue de se fixer des objectifs réguliers. Le logiciel Unity a été pris en main, et maintenant l'avancée du projet devrait être plus rapide. Cette première soutenance nous a permis de nous rendre compte que la date butoire approche à grand pas, et l'avancée du projet est désormais une véritable priorité. Pour la prochaine soutenance un planning a été mis en place montrant l'avancement théorique du projet à ce moment (cf chapitre 1), l'organisation autour de cet objectif va être rapidement mis en place.