

GAFAM EATERS

RAPPORT DE SECONDE SOUTENANCE

Jeu vidéo - Senso No Michi



Antoine Mosimann — E1
Louis Rodet — E1

Mathis Galliano — C1
Abel Chartier — E1

PROMO 2027

10 Mars 2023 — 17 Avril 2023

Table des matières

1	Introduction	2
2	Le projet ZenTianJi	3
2.1	Le cahier des charges	3
2.2	Scénario et gameplay	3
2.3	La répartition des tâches	4
2.4	Avancement par soutenances	4
3	Réalisation technique	5
3.1	Les personnages	5
3.1.1	Menu choix de personnage	5
3.1.2	L'intégration des personnages	7
3.2	Le site internet	8
3.3	Multijoueur	10
3.3.1	Téléportation inter-scènes	10
3.3.2	Interactions entre joueurs et monstres	10
3.3.3	Synchronisation des vies	11
3.3.4	Détection d'objets	12
3.3.5	Connexion aux scènes	12
3.4	Intelligence artificielle	13
3.5	Levels Game Design	16
3.6	Audio	19
3.7	Gameplay	20
3.7.1	Bugs	20
3.7.2	Personnages et compétences	20
3.7.3	Amélioration des sorts	22
3.7.4	L'HUD	24
4	Poins négatifs	25

1 Introduction

Vous trouverez dans ce rapport de seconde soutenance les nouvelles créations, les changements ainsi que les améliorations techniques du jeu vidéo de l'équipe GAFAM EATERS.

Ce jeu vidéo appelé SENSO No MICHI est un jeu de rythme sous la forme d'un RPG en trois dimensions.

Nous vous exposerons son contenu, les tenants et aboutissants du projet ainsi que son organisation et sa répartition au sein du groupe.

2 Le projet ZenTianJi

2.1 Le cahier des charges

Un cahier des charges a été écrit en décembre dernier afin définir le scénario du jeu, la répartition des tâches et l'avancement par soutenance.

Il avait été remodifié entre son premier rendu et la première soutenance, mais il n'a pas changé entre la première soutenance et la seconde.

Nous avons pris la liberté de modifier quelques éléments de ce cahier des charges qui ne convenait pas à chacun mais ceux-ci n'impacteront pas les grandes lignes du projet initial.

2.2 Scénario et gameplay

Le scénario reste le même que défini, à une différence près : nous avons abandonné le côté "légende chinoise". Dans le scénario, des guerriers venant de trois grandes familles (La famille des mages, des guerrier et des assassins) ayant pour but de "s'unir et de suivre les pas de leurs ancêtres pour rétablir un monde de paix parmi la terreur semée par les envahisseurs".

Il est possible de jouer au jeu en solo, mais surtout en multijoueur ou la coopération est très importante car elle permet d'unir les forces des trois grandes familles.

Les joueurs jouent, attaquent, se déplacent en rythme avec la musique ce qui leur procurent des bonus pour être plus forts.

2.3 La répartition des tâches

Nous rappelons la répartition des tâches à laquelle nous nous sommes appliqués à suivre pour pouvoir travailler plus vite et mieux.

	Louis Rodet	Mathis Galliano	Antoine Mosimann	Abel Chartier
Chef de projet		Délégué	Suppléant	
Modelage 3D et animation		Délégué		Suppléant
Musique et sons		Suppléant		Délégué
Gameplay			Suppléant	Délégué
Levels Designs	Suppléant	Délégué		
Menus	Délégué	Suppléant		
Site Internet	Délégué			Suppléant
Intelligence Artificielle	Suppléant		Délégué	
Multijoueur	Suppléant		Délégué	

2.4 Avancement par soutenances

Voilà le détail de l'avancement par soutenance. Nous rappelons l'avancement auquel nous étions à la fin de la première soutenance, et précisons là où nous estimons être à la date de la seconde soutenance.

	Soutenance 1	Soutenance 2	Soutenance 3
Modelage 3D et animation	60%	90%	100%
Musique et sons	60%	80%	100%
Gameplay	65%	75%	100%
Levels Designs	60%	75%	100%
Menus	65%	90%	100%
Site Internet	45%	90%	100%
Intelligence Artificielle	30%	50%	100%
Multijoueur	70%	90%	100%

3 Réalisation technique

3.1 Les personnages

3.1.1 Menu choix de personnage

Nos menus précédents, ainsi que tout notre jeu n'intégraient pas la possibilité que le joueur puisse choisir entre plusieurs personnages.

Nous avons donc commencé par implémenter un menu de choix de personnages.

Une des possibilités que nous avions était d'offrir au joueur la possibilité de choisir tous les éléments de son personnage : son sexe, sa couleur de peau, sa coupe de cheveux, son armure, son arme, etc...

Les assets que nous avions nous le permettaient, mais après avoir compris comment fonctionnaient les *assets*, nous nous sommes rendu compte qu'il aurait été bien trop long de faire cela. En effet, ce n'était pas très compliqué de faire cela mais très répétitif. Nous aurions dû ajouter tous les éléments des personnages (bras, têtes, torses, ...) ainsi que leur *materials* pour pouvoir recréer les personnages, le tout en plaçant correctement les différents éléments du corps. Nous avons compté, il y aurait eu plus de 600 combinaisons de personnages, donc nous ne pouvions pas créer un *prefab* par combinaison ce qui aurait rendu la tâche bien plus simple.

Nous avons donc décidé d'ajouter un menu dans lequel on voit les 4 uniques personnages possibles, 2 féminins et 2 masculins. Ils sont chacun dans une salle entourée de torches, et sont dans leur animation par défaut. La caméra qui les filme tourne autour d'eux entre un angle de -45° et un angle de $+45^\circ$ pour rendre l'image moins statique. Lorsque l'on clique sur un personnage, les torches qui entourent le personnage émettent plein d'étincelles, et le personnage se retrouve encadré par sa couleur.



Figure 1 — Aperçu Menu Choix de personnages

Les positions et rotations des caméras sont calculées en temps réel par trigonométrie.

Nous pouvons noter que pour simplifier les calculs, nous avons implémenté des listes de GameObject plutôt que de noter les variables une par une, ce qui permet d'y voir plus clair, voyez plutôt ci-dessous.

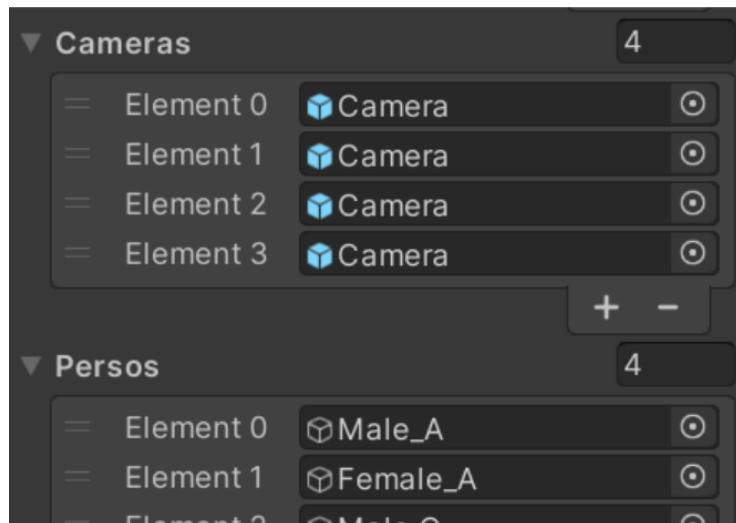


Figure 2 — Aperçu des listes sur Unity

3.1.2 L'intégration des personnages

Dans le mode solo nous avons une scène pour choisir les personnages, pour instancier les personnages sur la carte nous avons tout d'abord voulu instancier directement le joueur avec une Prefab. Mais cette méthode posait certains problèmes et des erreurs.

Pour régler cela, tous les joueurs sont tous déjà instanciés sur la carte (ce qui ne pose pas un problème car nous ne sommes pas en multijoueur) et sont supprimés par rapport au choix que nous avons fait dans l'ancien menu. (Les informations sont transmises de scène en scène avec des instances).



Figure 3 — Intégration des personnages

Ce menu est donc parfaitement fonctionnel, il suffira donc de rajouter les apparences des joueurs pour la prochaine soutenance.

3.2 Le site internet

Lors de la première soutenance, Louis Rodet avait réfléchi et implémenté le design du site web, fait le menu (qui avait du mal à s'afficher pour la version mobile), et il s'était arrêté là.

Aujourd'hui, tout le site dans ses moindres détails est implémenté, en HTML5 et CSS3. Pour commencer, le menu est responsive, trois barres s'affichent quand la largeur de l'écran est inférieure à 768 pixels.



Figure 4 — A gauche la page web version ordinateur et à droite la page web version mobile avec le menu déroulant

Sur cette barre de navigation il y a six différents onglets :

- Le premier est le logo de la société “GAFAM Eaters” et le nom du projet “ZenTianJi”. En cliquant dessus, il nous ramène à la page d'accueil.
- Le second appelé “Accueil” ramène à la page d'accueil qui sert de présentation globale du jeu.
- Le troisième appelé “Personnages” ramène à la page personnages qui sert de présentation des personnages du jeu.
- Le quatrième appelé “Univers” sert de présentation du décor, de l'histoire et tout l'univers du jeu dans ses moindres détails.
- Le cinquième appelé “À propos” permet de présenter l'équipe, les origines du projet, et surtout permet de préciser que ce projet de jeu vidéo a été fait à l'EPITA, et que l'école peut communiquer sur le jeu si elle le souhaite (voir figure 5 ci-dessous).
- Pour finir, le sixième onglet est simplement un bouton qui permet de télécharger le jeu en format zip.

Nous estimons que le site web est fini à 80% : il reste encore à écrire quelques détails que nous rajouterons dans les pages personnages et univers, et à mettre en ligne le site sur <http://senzo.no.michi.free.fr>.



Figure 5 — Crédits auprès de l'EPITA

Nous avons déjà réservé le nom de domaine, il ne reste plus qu'à ajouter le site web dessus.

3.3 Multijoueur

3.3.1 Téléportation inter-scènes

A la dernière soutenance nous avions un problème de téléportation de scène des joueurs, quand nous rentrions dans le portail, le client maître n'avait aucun problème, mais les prochains clients avaient une duplication du personnage.

Pour pallier cela nous avons utilisés une méthode souvent utilisée qui comporte beaucoup d'avantages. Il s'agit de simplement téléporter le joueur dans la même scène à une grande distance. La carte est donc déjà chargée, et il n'y a pas de temps de chargement. Aussi toutes les spécifications du joueur sont sauvegardées, au lieu de devoir les recharger dans la scène suivante.

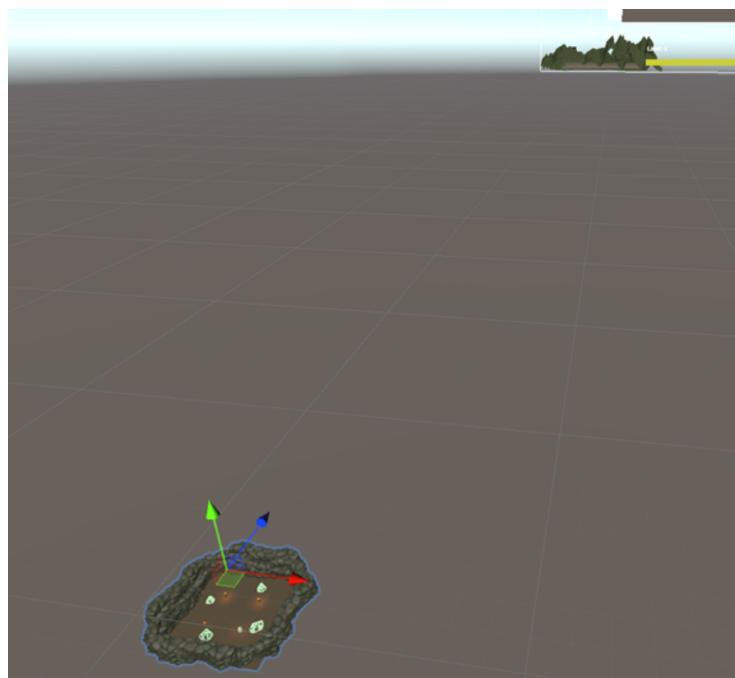


Figure 6 — Téléportation inter-scènes

3.3.2 Interactions entre joueurs et monstres

Pour la première soutenance nous avions des problèmes d'interaction en multijoueur entre le joueur et les intelligences artificielles en ligne. Par exemple seul le premier joueur connecté pouvait prendre des dégâts.

Pour régler ce problème j'ai donc tout d'abord dû utiliser une fonction pour récupérer tous les objets ayant un certain attribut (Les statistiques du joueur) puis nous avons dû utiliser les « Appel de procédure à distance » de photon qui permettent d'appeler la fonction de prise de dégâts du joueur, en faisant une requête au serveur. Ainsi chaque client peut mettre à jour sa scène en conséquence.

3.3.3 Synchronisation des vies

Ces appels serveurs de photons peuvent aussi par exemple permettre de synchroniser la vie ou les dégâts entre tous les clients. Ainsi avec cette méthode nous avons pu afficher la vie de tous les joueurs en ligne :



Figure 7 — Synchronisation des vies

Nous avons eu certains problèmes, par exemple les 4 barres de vies s'affichaient même si seulement deux joueurs étaient connectés sur les 4.

La solution a été de désactiver les barres en fonction du nombre de joueur, ainsi que de désactiver la barre du joueur actuel.

Un autre problème était que les barres de tous les joueurs se superposaient, car lorsque nous instancions un joueur sur la carte, tous ses attributs sont envoyés au serveur et à tous les clients, ce qui faisait bugger l'affichage. Nous avons donc désactivé les HUD des autres joueurs.

3.3.4 Détection d'objets

Quand l'on n'est pas sur serveur il est simple de savoir sur quels objets vont interagir le joueur, et donc pouvoir mettre des liens directs ou détecter des "tags". Sauf qu'en multijoueur il peut y avoir un nombre indéterminé de joueurs (nous en avons quatre, mais il est mieux de prévoir pour en avoir plus). Ainsi il est plus compliqué d'avoir des liens.

Une méthode que nous avons utilisée, est de chercher les objets par composants. Et avec ces composants nous pouvons récupérer certaines données, comme la position, l'orientation, la vie...

3.3.5 Connexion aux scènes

Pour la première soutenance, nous avions juste un bouton pour se connecter en multijoueur et nous ne pouvions pas choisir avec qui nous nous connections. A présent nous avons un menu pour créer sa propre partie et se connecter à n'importe quelle partie en cours.



Figure 8 — Connexion aux scènes

Il reste encore quelques problèmes dans ce menu, par exemple lorsqu'un client se connecte et se déconnecte, la scène est affichée en double :

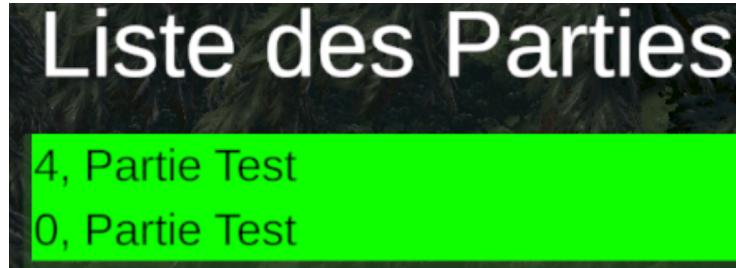


Figure 9 — Problème d'affichage de la scène

Nous avons néanmoins rencontré divers problèmes. Quand un joueur se connecte au client maître, la barre de vie reste à 0 tant que le joueur ne prend pas de dégâts. Probablement car il n'y a pas de requête lors de la connexion.

Ces mises à jour utilisant les appels de procédure à distance de photon utilisent beaucoup de ressources, et certains lags peuvent apparaître.

Un autre problème que nous avons eu, est la différence de fonctionnement entre l'éditeur de Unity et les versions build du projet. Certaines fonctionnalités fonctionnaient très bien sur l'éditeur et non sur les versions build sans réelles logiques.

Avant d'expliquer l'un des problèmes majeurs de l'éditeur pour le développement du jeu, une Room pour photon est une scène en ligne, ayant un identifiant unique. C'est le client maître (le premier client) à se connecter qui la crée.

Mais le plus gros problème a été lorsque l'éditeur créait une Room même s'il y en avait déjà une en ligne. C'était donc impossible de tester notre projet sans build notre jeu et le copier pour l'ouvrir deux fois.

3.4 Intelligence artificielle

Durant la première soutenance nous avions mis en avant un certain retard du côté du développement de l'intelligence artificielle, nous utilisions une *IA* basique qui, suivait le joueur en tout point de la carte et l'attaquait quand il était assez proche.

Nous avons maintenant une intelligence artificielle qui repère le joueur dans une certaine zone, le poursuit jusqu'à pouvoir l'attaquer, si le joueur sort de la zone de détection, alors l'*IA* revient à sa position initiale. Cela permet évidemment au joueur de pouvoir s'enfuir sans pour autant avoir une horde d'ennemis à sa

poursuite, mais cela permettra aussi dans un second temps de libérer l'espace pour charger moins d'ennemis en même temps.

Évidemment nous avons rencontré quelques soucis, notamment au niveau des animations, en effet durant l'attaque, l'ennemi avance tout de même vers le joueur.

Un second problème a été l'organisation du script pour prendre en compte tous les personnages auxquels nous voulions attribuer cette *IA*. Ce problème c'est donc manifester sur les *Animators*. Pour ceux-ci nous utilisons comme à la première soutenance, des booléens pour lancer les transitions. Ces booléens doivent donc être identiques sur chaque *Animator* de chaque personnage pour que le script soit compatible avec tous les ennemis. Le souci a donc été en fonction des animations des différents *assets* puisque certains ont des animations supplémentaires comparés à d'autres. Un troisième problème rejoint le second, un des *assets* utilisés est à l'envers ce qui fait qu'un ennemi nous poursuit et nous attaque à l'envers.



Figure 10 — IA du gobelin et de l'araignée



Figure 11 — Personnage et IA de l'araignée

Pour ce qui est de l'intelligence artificielle des ennemis de fin de zone, elle n'est pas encore terminée, mais se base sur un système d'actions aléatoire, reprend les bases de l'intelligence plus simple présent sur les ennemis de base de début de jeux.

Toutefois, celle-ci ne comprend pas une zone de repérage, dès que le joueur entre dans la zone du boss, celui-ci ne peut perdre le joueur de vue, et commence donc à combattre. Plusieurs situations sont possibles et choisies en fonction de la position du joueur par rapport à celle du boss.

Dans un premier temps, si le joueur est assez proche, une action est choisie aléatoirement entre attaquer celui-ci ou se protéger d'une potentielle attaque pendant un court laps de temps. Si le joueur n'est pas assez proche pour être attaqué, alors le but est que le boss se déplace jusqu'au joueur, mais pour cela, ce n'est pas forcément le chemin le plus court qui est emprunter. En effet, dans cette situation le boss a pour objectif le joueur mais peut aléatoirement aller à droite, à gauche, ou tout droit.

Si le joueur est à une certaine distance, alors le boss court jusqu'à une distance raisonnable pour décider d'une action. A cela s'ajoute une situation aléatoire où une action est effectuée comme reculer, aller à droite, à gauche ou encore ne rien faire du tout.

Un dernier problème que nous avons rencontré et à cause duquel l'évolution d'une intelligence artificielle sur la carte principale de jeu est impossible pour le moment est le *NavMesh*.

En effet, malgré de nombreuses recherches nous ne trouvons toujours pas le moyen de faire suivre les routes à notre IA. Aucune solution trouvée ne nous satisfait réellement puisqu'elle implique soit de dupliquer la carte du jeu pour recréer un terrain, le superposer au terrain existant. Cela aurait un impact négatif sur les performances de notre jeu et nous ne l'envisageons pas encore.

Une seconde solution que nous avons trouvée mais que nous n'envisageons pas du tout est de refaire entièrement la carte du jeu sans le faire via un terrain.

Une dernière solution qui semble plus envisageable qui resterait plutôt coûteuse en terme de performances serait de créer des barrières de collision sur la carte au bord des routes.

3.5 Levels Game Design

Lors de la dernière soutenance nous avions présenté un début de carte de jeu avec un village de départ dans une zone forestière. Aujourd'hui nous avons fait quelques campements de plus ainsi que quelques emplacements de futurs campements, ceux-ci serviront notamment pour soigner le joueur, sauvegarder la partie et servir de point de téléportation.



Figure 12 — Point de campement



Figure 13 — Point de campement

La zone forestière a été finie et une seconde zone désertique a été faite à moitié. Une salle pour un combat d'ennemis de fin de zone avait aussi été présentée, une seconde a été préparée, nous avons maintenant deux salles de boss, lesquelles sont décorées en fonction de la zone à laquelle elles correspondent.

De nombreuses zones plus larges sont présentes sur la carte, elles seront utilisées comme zone de combat. En effet en fonction des zones de nombreux et différents ennemis pourront apparaître sur ces zones et permettre au joueur de ne pas tourner en rond de village en village.

En ce qui concerne les différents combats de boss de zone qui seront disponibles à chaque zone, la téléportation vers l'antre du boss se fera via un portail que nous avions présenté à la première soutenance, ces portails seront différents en fonction du boss et de la zone. Une fois dans l'antre du boss, le combat commence !

Plus le joueur avancera dans le jeu plus le boss sera difficile à vaincre que ce soit en terme de force ou de vie le boss surclassera le joueur et chacun des boss aura une *IA* différente en fonction de sa difficulté ce qui permettra au joueur une expérience différente à chaque zone.



Figure 14 — Salle de boss



Figure 15 — Portail de téléportation

De plus certains chemins abritent des coffres cachés qui pourront donner plus d'expérience au personnage lui permettant alors d'évoluer plus vite.



Figure 16 — Coffre d'expérience

3.6 Audio

Comme nous avons un jeu basé sur le son nous avons ajouté dans les paramètres un curseur pour changer le volume du jeu. Le volume du son est géré sur toutes les scènes à partir de ce menu.



Figure 17 — Menu avec l'onglet volume

3.7 Gameplay

3.7.1 Bugs

L'objectif numéro un à l'approche de cette seconde soutenance était la correction des bugs rencontré lors des débuts du projet et reportés lors de la nouvelle soutenance.

Parmi ces bugs ont pouvais en observer certain ruinant complètement l'expérience de jeu tel qu'une erreur dans le système de détection d'ennemi qui résultait qui stoppait toutes les fonctionnalités des scripts liés au combat dès qu'un ennemi était tué.

D'autres problèmes liés au système de rythme et aux sorts déjà implémentés lors de la 1ere soutenance ont aussi été corrigé.

3.7.2 Personnages et compétences

Afin d'apporter plus de diversité au gameplay nous avons décidé d'implémenter différents personnages en se basant sur les archétypes classiques des héros de fantaisie médiéval classique.

Nous avons donc opté 4 personnages dont trois sont déjà jouables dans les grandes lignes :

Un Mage/Support disposant d'un sort de soin et d'un sort de zone ;



Figure 18 — Sorts

Un héros centré sur les dégâts permettant d'atteindre les plus gros chiffres du jeu avec un sort d'augmentation de dégâts et un sort ciblé au corps à corps ;



Figure 19 — Sort augmentation de dégâts

Un assassin jouant avec la mobilité afin de permettre un gameplay *hit-and-run* ;



Figure 20 — *Hit-and-run*



Figure 21 — *Hit-and-run*

Pour le dernier personnage nous avons décidé d'implémenter un tank ayant pour capacité d'attirer les ennemis vers lui afin de protéger ses coéquipiers.

L'objectif de rendre tous ces personnages capable d'interagir entre eux afin de rendre des synergies possibles pour rendre le multijoueur intéressant et innovant.

3.7.3 Amélioration des sorts

Pour toujours pousser plus loin la liberté du joueur nous lui permettons d'améliorer ses sorts suivants un arbre de talent.

Pour illustrer ce concept nous l'avons déjà implémenté sur le support qui peut transformer son sort de soin en un sort de zone plus adapté au multijoueur

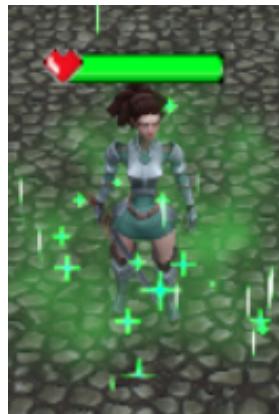


Figure 22 — Amélioration de sorts

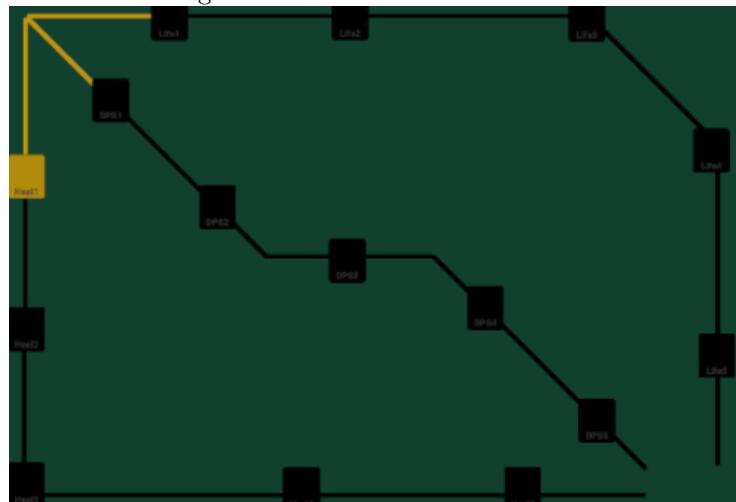


Figure 23 — Arbre des compétences



Figure 24 — Sort de soin

Tous les personnages pourront à terme améliorer leurs deux sorts sur deux branches distinctes de l'arbre (*cooldown*, effet, puissance) et une branche augmentant les statistiques de base du personnage.

3.7.4 L'HUD

Afin de rendre les différentes informations déjà implémenté à la dernière soutenance visible, nous avons implémenté un HUD donnant des informations tel que le combo l'expérience et le niveau.



Figure 25 — L'HUD

4 Points négatifs

Durant la réalisation du projet de second semestre, l'équipe GAFAM EATERS peut rencontrer de nombreux problèmes internes alors que nous découvrions le travail d'équipe.

La création de l'équipe n'était pas évidente, car Louis et Antoine se connaissaient très bien, Abel qui était dans leur classe a ensuite été intégré à l'équipe, et Mathis a été ajouté car il était une connaissance d'Antoine sur *Discord*. Il peut donc être difficile d'exiger du travail de la part des autres quand on ne les connaît pas très bien. De plus le fait que toute l'équipe ne soit pas dans la même classe rendait la communication au sein de l'équipe encore plus difficile car il était très rare que toute l'équipe se voit en même temps en physique.

Pour remédier à ces problèmes, nous avons commencé par programmer des réunions en physique de manière hebdomadaire, mais nous avons mis du temps trop de temps à réellement créer le projet. En effet, Mathis qui était chargé créer toute la partie graphique du jeu (poser des textures, des arbres, des chemins, etc.) a été pour le reste de l'équipe une occasion de ne rien faire, donc nous avons commencé toute la partie programmation qu'à partir de février.

La partie programmation n'ayant pas besoin de la partie graphique pour être développée, nous aurions dû commencer à partir de décembre afin d'éviter le rush d'avant soutenance.

Heureusement, aujourd'hui nous connaissons mieux, ce qui nous permet de nous répartir plus facilement le travail sans avoir à forcément nous réunir tout le temps

Pour finir, nous sentons que l'équipe commence à ralentir pour des raisons de perte de motivation. En effet, nous avions de grands rêves pour ce projet de jeu vidéo, et nous rendons peu à peu compte qu'il ne se réalisent pas. Créer des jeux vidéos est un vrai métier, surtout quand on veut conquérir un public, ce qui est forcément compliqué pour un projet étudiant. Mais nous ne comptons pas nous arrêter là, et nous allons pousser pour que le jeu soit le meilleur possible à la fin de la dernière soutenance !

Ce sont sur tous ces points que l'équipe aimeraient s'améliorer ou s'est déjà améliorée, pour n'avoir qu'à se concentrer sur le projet, et non sur des problèmes d'organisation !

Conclusion

Pour conclure, ce projet nous fait vivre une très grande aventure : du code en langage C#, aux relations humaines en passant par des *merge conflicts*, nous sommes parés pour l'avenir !

Nous sommes une équipe bien unie et toujours positive qui saura toujours résoudre tout type de problèmes !

Pour la prochaine soutenance, comptons finir avec succès notre jeu, en finissant d'implémenter nos menus, nos boss et de nouvelles mécaniques de jeu qui permettront à notre jeu d'avoir des mécaniques à la fois complexes et intéressantes.

C'est ainsi que s'achève la seconde partie du projet de notre semestre deux à l'EPITA. Nous vous tiendrons au courant des dernières évolutions du projet et de son équipe lors de notre dernier rapport de soutenance.