

EPITA

RAPPORT DE PROJET

LES BARDES

GAMMA

Auteurs :

Laure MARCHERON
“Tombelune”

Louis LE GATT
“jetsPepito”

Romain HERMARY
“Kazoryah”

Gaël GAUTIER
“G.”

6 juin 2018

Table des matières

Introduction	3
1 Game Design & Gameplay	4
1.1 Scénario	5
1.2 Collectables	6
1.3 Timer	7
1.4 Score	8
1.5 Gameplay	8
1.5.1 Plateformes	10
1.5.2 Boutons, Portails	10
1.6 Intelligence Artificielle	11
2 Level Design & Build	12
2.1 Niveau 1	13
2.2 Niveau 2	15
2.3 Niveau 3	16
2.4 Niveau 4	17
2.5 Travail sur les lumières & textures	18
3 Multijoueur	21
4 Niveau Musical	24
4.1 Les mélodies	24
4.2 Création du niveau	26
5 Menu	29
6 Modélisation 3D & Animation	31
6.1 Modélisation 3D	31
6.2 Animation	35

7	Site Internet	37
8	Récits personnels de la réalisation	39
8.1	Romain	39
8.2	Louis	40
8.3	Laure	41
8.4	Gaël	42
	Conclusion	45
	Annexes & Sources	46
8.5	Sources	46
8.6	Captures d'écran	47

Introduction

Nous sommes le groupe des Bardes, composé de Laure MARCHERON, Louis LE GATT, Gaël GAUTIER et Romain HERMARY. Nous avons travaillé durant ce deuxième semestre sur un projet de jeu vidéo intitulé GAMMA. Celui-ci est un jeu de plateformes musical en 3D, axé sur la coopération entre deux joueurs et donc purement multijoueur. Il s'articulera en deux modes de jeu : une possibilité de jouer sur un même clavier, donc sur le même ordinateur, et une possibilité de jouer sur deux ordinateurs différents, ce qui introduit évidemment la nécessité d'un serveur. Ces deux modes de jeu aboutiront pourtant au même objectif : une entraide entre deux joueurs, qui leur permettra d'évoluer à travers les quatre niveaux proposés de manière à terminer le jeu.

Pour mener à bien ce projet, nous nous sommes inspirés de plusieurs types de jeu tels que les jeux de plateformes classiques (2D et 3D) et les jeux musicaux, de leur univers ainsi que de leurs codes pour proposer un jeu cohérent et original car mélangeant plusieurs genres. Nous avons repris quelques mécaniques de jeu qui font l'essence des jeux de la célèbre licence *Mario* par Nintendo, c'est-à-dire une avancée plus ou moins linéaire dans plusieurs niveaux dans lequel le joueur doit faire preuve de réflexion et d'habileté à contrôler le personnage qu'il incarner. Nous avons également repris l'aspect de réflexion, qui est une notion centrale au sein des jeux *Portal*, ainsi que l'idée des portails de téléportation, qui permettent une nouvelle manière de concevoir les niveaux et leur résolution. Mais GAMMA n'est pas seulement un platformer en 3D : il contient également une partie musicale, qui permet de valider la fin de chaque niveau et d'ouvrir la voie aux niveaux suivants. Pour cette partie, nous nous sommes inspirés de jeux mobile tels que *Deemo* ou *Voez*, qui eux-mêmes ont été inspirés par des jeux du type *Guitar Hero*.

Notre volonté concernant ce jeu était de le rendre accessible au plus

grand nombre. Par conséquent, il fallait y introduire les éléments de gameplay de manière progressive, pour permettre aux joueurs de prendre le jeu en main de façon optimale, et de développer leur coopération à leur rythme. Chaque niveau est d'une difficulté croissante et nécessite absolument la collaboration entre deux joueurs, étant donné que chacun des personnages possède une capacité qui lui est propre et qui permet de terminer les niveaux : le personnage féminin peut traverser des portails de téléportation pour découvrir de nouveaux chemins, bonus et interrupteurs, tandis que le personnage masculin peut pousser des blocs de manière à débloquer certains chemins. Ensemble, ils devront traverser les épreuves des différents niveaux : labyrinthe, ennemis prêts à tuer qui-conque s'aventure dans leur champ de vision, plateformes farceuses, murs bloquant le passage, épreuve musicale. Le tout réparti au sein de quatre niveaux qui demanderont de plus en plus de dextérité, de réflexion et de réflexes.

Au travers de ce rapport de projet, vous pourrez en apprendre plus sur le travail que nous avons fourni pour mener notre jeu vidéo à bien. Nous tâcherons de relater de manière chronologique et par personne les différents aspects du jeu que nous avons évoqués ci-dessus, en fonction de ce dont chacun s'est occupé.

1 Game Design & Gameplay

La partie Game Design et Gameplay pour GAMMA concerne principalement l'élaboration des différentes mécaniques de jeu qui sont présentes dans le jeu, ainsi que la caméra, les personnages et leurs contrôles. Autrement dit, cette partie doit proposer des éléments de gameplay diversifiés et également respecter la "règle des trois C" : "camera", "character", et "controls". En effet, il nous a fallu choisir une vue spécifique

pour la caméra du jeu, que nous nous sommes accordés à définir en troisième personne : ainsi, on suit les mouvement du personnage contrôlé comme un spectateur positionné dans son dos. Concernant les personnages d'ailleurs, nous avons décidé d'en introduire deux, puisque le but de GAMMA est de finir les niveau en coopération entre deux joueurs. Nous avons donc créé un personnage masculin et un personnage féminin, qui ont chacun des caractéristiques spécifiques : le personnage féminin est capable de passer des portails téléporteurs, tandis que le personnage masculin a la capacité de pousser des blocs pour dégager le chemin ou donner accès à une plateforme trop haute. Pour finir avec les contrôles, ceux-ci sont différents selon le mode de jeu. En effet, nous avons parlé durant l'élaboration du cahier des charges de la possibilité de jouer sur un même ordinateur ou bien sur deux ordinateurs différents. Si le multijoueur est local, dans ce cas l'un des personnages est contrôlé par les touches Z, Q, S, D sur clavier français tandis que l'autre se dirige avec les flèches directionnelles. Si le multijoueur est entre deux ordinateurs différents, chaque personnage est contrôlé par les flèches directionnelles du clavier.

Les éléments de gameplay quant à eux sont détaillés ci-dessous. Le scénario de GAMMA est également rappelé.

1.1 Scénario

Dans le royaume de Gamma, vaste péninsule prospère où la musique a une importance capitale, la puissance des rois se mesure à leur maîtrise prodigieuse des Sept artefacts renfermant l'essence même de la musique. Ceux-ci, détenus depuis des temps immémoriaux dans le palais royal de Gamma, étaient cependant convoités par les contrées voisines, envieuses du pouvoir bienveillant qu'offraient les artefacts au peuple Gammarois.

En ce temps régnait le roi Octave, un roi bon envers ses sujets, mais orgueilleux et fier auprès de quiconque osait le sous-estimer ou le défier.

Malheureusement, son caractère impulsif et suffisant le mena bientôt à sa perte : Bémol, roi envieux d'une contrée voisine, projeta en effet de provoquer Octave dans le but d'obtenir un duel et, peut-être, de récupérer les artefacts de musique. Évidemment, Octave, blessé dans son égo, accepta le duel, qui prit place dans l'immense cour du palais royal de Gamma sous l'œil attentif des Gammarois. Quelle ne fut pas leur horreur, lorsqu'au terme d'un combat acharné entre les deux rois, Bémol brandit un poing victorieux vers le ciel alors qu'Octave gisait au sol, blessé et démuné du pouvoir qui lui avait été transmis...

Octave fut contraint de tenir parole. Malgré le vent de protestation au sein du peuple Gammarois, Bémol avait gagné. A sa guise à présent d'emporter les artefacts avec lui et de jouir pleinement de leur pouvoir. Effondré par cette perte inestimable et hué par son peuple, Octave se vit obligé de se retirer dans son palais et devint sourd aux revendications de ses sujets. Dans son désespoir, il fit appel à ses deux conseillers, ceux-là même qui avaient essayé de le décourager à accepter le duel. Après de nombreux jours à supplier, Octave obtint enfin ce qu'il voulait : ses conseillers, habiles et rusés, iraient subtiliser les artefacts à Bémol.

1.2 Collectables

Louis

Notre jeu étant un platformer, il lui fallait tout d'abord absolument un système de score, mais aussi d'éléments qui puissent augmenter ce score, c'est-à-dire des bonus à l'image des étoiles utilisées dans la célèbre licence Mario. Ainsi, j'ai décidé de travailler sur ces objets avec Laure, qui a modélisé une note de musique en 3D. Cela correspond parfaitement à l'univers musical du jeu et est très vite repérable comme étant un bonus important dans le jeu. Après avoir récupéré la note modélisée, je l'ai implémentée dans une scène puis j'ai ensuite travaillé sur celle-ci. Les différentes choses importantes à réaliser pour obtenir un élément

collectable étaient les suivantes : créer un parallélépipède autour de la note afin d'avoir une zone dans laquelle le joueur récupère la note, augmenter le score actuel du joueur et supprimer la note lorsque celle-ci a été récupérée par un des deux joueurs. J'ai ajouté un collider afin de détecter lorsque le joueur entre en collision avec la note. Ceci m'a permis d'ajouter un script à la note qui permet de détecter la collision avec le joueur. Lorsque cette collision est détectée, l'objet disparaît et le score est actualisé. Le bonus était alors créé. Il a ensuite suffi de les placer aux différents endroits décidés par Gaël et Laure lors de la conception des niveaux. Ces objets sont donc au final un moyen d'augmenter son score durant la partie mais aussi un moyen pour le joueur d'avoir un défi en plus lors de la réalisation du niveau. En effet, le joueur n'a plus comme seul objectif de finir le niveau, il peut aussi décider de récupérer la totalité de ces bonus, qui sont parfois un challenge à récupérer car demandant plus de concentration et de dextérité.

1.3 Timer

Laure

GAMMA comportera des niveaux qui seront munis d'un timer. Cela impliquera plusieurs choses concernant chaque niveau : tout d'abord, ce timer indiquera aux joueurs combien de temps ils ont, au maximum, pour terminer le niveau. Nous ne voulons pas que cela conduise à du speedrun obligatoire, mais nous pensons que cela peut constituer un petit bonus satisfaction aux joueurs qui auront vite résolu les niveaux. Si jamais le timer tombe à zéro, la sentence est irrévocable : c'est un Game Over pour les joueurs, qui perdent immédiatement tous leurs points de vie et qui doivent tout recommencer du début. De plus, le timer aura une fonction de bonus score à la fin de chaque niveau, ce qui rejoint la satisfaction du joueur d'avoir fini le niveau rapidement, car il sera récompensé pour ses efforts.

L'implémentation du timer n'a pas été si compliquée que ça. J'ai eu de l'appréhension concernant le code en lui-même, car je mets en général du temps à comprendre ce que fait une fonction, et j'avais un peu de mal à comprendre les principes de classes, attributs etc. lors des TPs de C# en cours. Cependant, avec l'aide d'un tutoriel disponible sur YouTube qui expliquait pas à pas ce que devait faire le timer, et avec une explication brève de ce que faisait une fonction "OnGUI", j'ai réussi à créer un timer qui diminue d'une unité à chaque seconde, qui s'affiche en haut de l'écran à gauche dans un rectangle.

1.4 Score

Laure

Je parlais de bonus score concernant le timer, mais encore faut-il avoir implémenté un score en tant que tel. Celui-ci dépendra des bonus récupérés par les joueurs pendant le niveau, comme les notes de musique, ainsi que du timer comme dit plus haut. L'implémentation du score n'a pas été beaucoup plus difficile que celle du timer, à ceci près qu'elle requerrait l'appel à un autre script écrit par Louis pour faire disparaître la note de musique une fois le joueur passé dessus et comptabiliser la valeur du bonus au score total. Grâce à ce script, j'ai pu comme pour le timer afficher le score dans un rectangle sur la partie haute de l'écran, qui a la capacité de se mettre à jour à chaque passage sur un bonus. Le rectangle obtenu dans ma fonction "OnGUI" rendait mieux visuellement que le canvas initialement prévu par Louis, c'est pourquoi nous avons préféré supprimer le canvas.

1.5 Gameplay

Romain

Pour éviter au joueur de tomber indéfiniment dans le niveau s'il loupaît une plateforme, et nous faire perdre du temps lors de la première

soutenance, j'ai décidé d'implémenter un système de réapparition quand le personnage tombe. J'ai rajouté des variables représentant les points de vie ainsi que les dégâts infligés aux joueurs lors de la chute. Quand le joueur n'a plus de vie, un message "GAME OVER" s'affiche au milieu de l'écran. Dans la même optique, et en modifiant le script du Timer de Laure, j'ai aussi implémenté le même message quand le temps est écoulé. Je n'avais pas réussi à trouver comment désactiver les commandes des joueurs, sachant qu'on n'utilise pas le "Player Controller" de Unity, mais ce n'était en fait qu'une condition à rajouter dans le code. La rotation du joueur est elle aussi remise à zéro.

Concernant les déplacements du joueur, nous avons décidé de ne pas reprendre le script de l'asset, que nous avons jugé trop compliqué. Le saut posait encore quelques soucis, comme le saut infini ou des sauts à des hauteurs différentes selon le temps de pression de la barre espace.

Par la suite, j'ai réglé les problèmes liés aux sauts : en effet, jusqu'à la première soutenance, il était possible d'effectuer de multiples sauts. J'ai réglé ce problème en calculant la distance entre le joueur et le sol, ce qui me permet d'interdire le saut quand le personnage est déjà en train de sauter. J'ai également amélioré le script de réapparition qui ne pose plus de problème au niveau de l'élan que le joueur avait parfois et qui n'avait pas lieu d'être. Quelques améliorations ont été effectuées sur les prefabs au niveau des colliders et des rigidbodies, et effectuées de nouveau sur les personnages modélisés et animés en plus de bien régler leur échelle. De plus, afin de rendre l'expérience de jeu et l'esthétique meilleures, j'ai perfectionné le script de déplacement du joueur afin que sa vitesse soit réduite quand il saute, et d'augmenter sa vitesse de redescente qui était beaucoup trop faible jusque là et n'était liée qu'aux lois physiques de base de Unity. Les animations finies et implémentées, le jeu a pris beaucoup de valeur et est d'autant plus agréable à regarder.

1.5.1 Plateformes

Ensuite, j'ai écrit un script pour des boutons qui activent des plateformes mobiles se déplaçant de haut en bas et de droite à gauche. J'ai ainsi réussi à faire deux types de boutons : l'un qui reste enclenché dès lors que le joueur est passé dessus pour l'activer une fois, l'autre qui reste enclenché jusqu'à temps que le joueur soit sorti de la zone du bouton et revienne l'activer. À la demande de Laure, qui s'occupait de distribuer les scripts aux différents éléments de gameplay sur Unity, j'ai modifié mon script et créé des classes pour les plateformes, pour qu'elles puissent bouger dans toutes les directions, mais aussi pour qu'un unique bouton puisse activer plusieurs plateformes, n'activer qu'une seule direction, avoir des plateformes qui chutent et se détruisent au bout d'un certain temps, et des plateformes qui bougent de manière indépendante.

1.5.2 Boutons, Portails

Dans les éléments de gameplay qui sont à la base de notre jeu, il y a les interactions entre l'environnement et le joueur uniquement possibles pour l'un des deux personnages. Indispensable pour que notre jeu est du sens, j'ai donc rajouté, modifié des scripts pour que l'on puisse choisir si les deux joueurs, ou bien un seul en particulier peuvent prendre tel ou tel portail ou activer tel ou tel bouton.

D'ailleurs, j'ai remarqué qu'un GameObject pouvait avoir plusieurs colliders différents, avec des Caractéristiques (notamment "Trigger") différentes. Partant de ce principe, j'ai pu modifier, voire au final simplifier les conditions d'activation des boutons et de ce fait améliorer la précision du jeu : il n'est plus nécessaire de sauter sur le bouton pour l'activer (car l'ancien script fonctionnait avec des collisions), passer dessus est suffisant.

1.6 Intelligence Artificielle

Romain

Pour l'IA, j'ai décidé de ne pas faire quelque chose de trop compliqué au départ. J'ai repris des prefabs que j'avais modélisées lors d'un autre tutoriel, en attendant d'avoir d'autres modèles et par la même occasion faire une animation basique avec ce que j'avais. J'avais réussi à coder un script qui calcule la distance entre l'ennemi et le joueur : à une certaine distance, l'ennemi réagit. Pour la première soutenance, sa tête devenait rouge et il se tournait en direction du joueur. Cependant, je n'avais pas encore réussi à faire actualiser la distance entre l'ennemi et le joueur au fil du temps, ce qui faisait que la seule distance prise en compte était la distance de départ entre le point d'apparition du joueur et la position de l'ennemi. Ensuite, j'ai commencé à coder des déplacements automatiques pour l'ennemi, une sorte de chemin de patrouille, qui marchait presque lors de la première soutenance : l'IA ne retournait pas au premier point de départ et ses rotations étaient quelques fois hasardeuses. Mon objectif était donc de réussir à faire synchroniser les distances avec Photon, peaufiner le code des chemins suivis par les IA, ce qui me permettrait de rajouter à ma guise et celles de mes coéquipiers des éléments de gameplay, d'interaction entre le joueur et les PNJ, et faire tirer des projectiles à l'IA.

Pour la deuxième soutenance, j'ai trouvé l'erreur dans mon code qui ne donnait pas une visualisation correcte entre le joueur et l'IA. Il n'y avait plus de problème de détection de la distance à laquelle l'IA prend en compte la présence du joueur (elle prenait en compte la position d'apparition du joueur dans le niveau et non pas sa position actuelle, ce qui était problématique). J'ai pu donc modifier cette erreur pour rendre l'IA de détection fonctionnelle et utilisable pour l'implémentation dans les niveaux. Par la suite, j'ai utilisé certains scripts d'un ancien tutoriel Unity effectué au tout début du projet pour prendre en main le logiciel. Je les ai

évidemment améliorés pour qu'ils s'adaptent à Photon et deviennent les scripts non pas d'un joueur mais d'une IA : notre jeu présente donc des mobs qui patrouillent sur un chemin prédéfini, détectent les joueurs et leur tirent dessus de manière à les obliger à les contourner, ou à trouver un autre élément de gameplay qui puisse détruire ou bouger l'IA, étant donné que les joueurs n'ont pas les outils pour leur infliger des dégâts. Ce sont encore simplement des cylindres qui tirent des boules, mais le fonctionnement interne étant là, seul leur aspect a été à changer. Pour cela, nous avons utilisé un modèle 3D téléchargeable gratuitement, car il était impossible de modéliser nous-même un mob dans le laps de temps qu'il nous restait.

2 Level Design & Build

Lors de l'élaboration du cahier des charges, nous avons défini le level design comme la partie qui englobait la conception des niveaux, la réflexion quant aux mécaniques de gameplay à intégrer dans chaque niveau. Nous voulions recréer, grâce à ces mécaniques de jeu, un mix entre plusieurs jeux célèbres : *Super Mario Odyssey*, le géant de la licence Nintendo, mais également *Portal* et les mécaniques que l'on retrouve dans les jeux musicaux de type *Guitar Hero* ou *Deemo*. Pour cette tâche, nous avons deux cerveaux créateurs : Laure et Gaël, qui se sont occupés de la conception puis de la modélisation des quatre niveaux qui composent notre jeu. Nous avions pour idée au départ de le découper en cinq niveaux, mais cela aurait été trop long à concevoir ; nous avons donc pris le parti de ne faire que quatre niveaux, sachant que le dernier est assez long et que le niveau de difficulté est assez élevé.

2.1 Niveau 1

Gaël

Le premier niveau a été pensé par Laure et moi. Nous avons pour objectif que ce niveau soit explicite et pose les bases des mécaniques de jeu. Pour cela, nous avons réfléchi lors de la conception à implémenter des mécaniques de jeu dites "classiques" (soit celles qui reviendront le plus souvent) afin que les deux joueurs les rencontrent et les comprennent dès le début. Nous avons choisi d'y implémenter les portails téléporteurs ainsi que les boutons qui permettent de faire bouger les plateformes. Nous avons ensuite conceptualisé le niveau. Ce dernier ne devait pas être trop long, pas plus de cinq minutes, ni trop difficile. Ainsi, afin d'éviter trop de difficulté technique pour le joueur mais pour y ajouter un peu de challenge, nous avons eu l'idée de finir le niveau avec un labyrinthe (dans lequel nous n'avons pas mis de murs au début), obligeant ainsi les deux joueurs à coopérer pour trouver le bon chemin. Après avoir posé ces bases, nous avons imaginé le design du niveau, d'abord sous forme d'un croquis papier. L'élaboration de ce dernier a nécessité 2 bonnes heures de travail, car c'était la première fois que nous nous exercions à la tâche. Une fois cela fait, je me suis occupé de la modélisation. Au départ, j'ai donc utilisé les objets basiques de Unity, notamment le cube pour modéliser le premier niveau. J'ai très vite rencontré une difficulté technique, qui a été l'adaptation du croquis papier en 2D à un modèle 3D. J'ai donc dû repenser certaines parties du premier niveau afin d'éviter que les joueurs puissent suivre un autre chemin que celui auquel nous avions pensé, autrement dit pour éviter qu'ils trichent. J'ai par la suite ajouté des bonus et des ennemis sur le niveau. Ce premier travail de modélisation a permis de poser les bases de l'utilisation de Unity pour moi, c'est d'ailleurs cette première expérience d'utilisation Unity qui me servira dans la suite du projet. En effet, les erreurs que j'ai pu commettre tout au long de cette première modélisation m'ont permis de voir comment ne pas les faire et surtout comment travailler de façon efficace sur Unity.

Cette première modélisation du niveau (voir annexes FIGURE 1) a connu une mise à jour la première soutenance. En effet, nous étions tous peu satisfaits du rendu esthétique des niveaux, pauvre. Nous avons donc recherché des assets gratuits et libres de droit à télécharger sur le Unity store, afin de donner une véritable ambiance de château aux niveaux. L'asset que nous avons choisi comprend un set complet de plateformes avec chemin, ainsi que des blocs de pierre standards et des éléments tels que des portes, arches, colonnes, orbes lumineuses... Nous avons donc eu un choix large d'éléments, ce qui nous a permis de customiser au maximum l'esthétique de nos niveaux. Suite au choix de tous les éléments, j'ai dû refaire intégralement le premier niveau. J'en ai profité pour mettre à jour certains passages du jeu et ajouter des murs au labyrinthe pour rendre sa résolution plus difficile et pour que cela s'accorde mieux la nouvelle esthétique du niveau. Ce travail de seconde modélisation du niveau 1 s'est révélé long et intéressant mais fastidieux. En effet, l'asset que l'on utilise ne proposait pas tout ce que l'on souhaitait : par exemple, les murs du labyrinthe ont été problématiques car les blocs de l'asset ne sont pas entièrement remplis, il manque une face. A cause de cela, il a fallu coller deux blocs ensemble à chaque fois, ce qui les rendait plus épais qu'ils auraient dû. J'ai donc parfois dû placer un à un les différents murs du labyrinthe, ce qui a été long. Une fois le travail de modélisation terminé, j'ai dû harmoniser la taille de mes plateformes avec celles qui existaient déjà dans le niveau 2 que Laure avait pris soin de modéliser rapidement avec l'asset. Ce travail a nécessité quelques modifications du niveau bien entendu, mais rien de long, seulement quelques manipulations.

Finalement, le travail total effectué sur le niveau 1 s'est avéré plus long que prévu, mais aussi moins difficile techniquement. En effet, la plus grosse partie du travail a été la conception du niveau et le choix des mécaniques de jeu, et non pas la modélisation comme j'ai pu le croire au début. Néanmoins, la modélisation a tout de même représenté le travail

le plus chronophage.

2.2 Niveau 2

Laure

Le deuxième niveau a également été pensé par Gaël et moi-même, étant donné que nous étions les deux membres chargés de la conception des niveaux. Le niveau 2 devait être un niveau de transition entre le premier, qui avait des allures de tutoriel, et le troisième niveau qui lui serait plus vaste. Ainsi, nous avons rapidement décidé de diversifier les éléments de gameplay, de manière à ce que les joueurs en découvrent plus et s'habituent au travail en coopération. Pour cela, il fallait diversifier le gameplay, et introduire réellement la nécessité de coopération pour avancer. En effet, dès le départ, les deux personnages doivent se séparer pour débloquer le chemin de l'autre : le personnage féminin passe par un portail qui lui permet d'activer un bouton ouvrant la voie au personnage masculin, qui lui pousse un bloc pour pouvoir poursuivre son chemin. Dans cette optique, nous avons conçu la suite du niveau en introduisant également une plateforme sensible au poids de chaque personnage, et qui par conséquent penche lorsque ceux-ci sont dessus : il faut donc faire attention au placement sur la plateforme et éviter de tomber, sinon on perd de la vie et on réapparaît au début du niveau. De même, plus loin, chaque personnage a besoin de l'autre pour parvenir aux dernières plateformes du niveau, car certaines plateformes sont dirigeables par des boutons que l'un des deux joueurs est obligé de presser pour que l'autre puisse avancer.

Pour la première soutenance, j'avais modélisé le niveau 2 avec les blocs de base disponibles sur Unity, c'est-à-dire des cubes sans texture qu'il a fallu déformer pour leur donner une allure de plateforme. Précisément, j'avais repris les mêmes blocs que ceux que Gaël avait utilisés

lors de la première modélisation du niveau 1, de manière à garder la même échelle pour tous les niveaux. Malheureusement, nous avons eu un problème avec le logiciel de mise en commun du projet que nous utilisons, GitHub, et l'échelle du niveau que je venais de modéliser avait été dérégulée : les coordonnées, que j'avais fait exprès de régler sur des données entières, étaient désormais décimales et inutilisables simplement si jamais il fallait effectuer des changements. Cependant, cela n'a pas été un problème pour la suite : en effet, après le passage en première soutenance, nous avons rapidement compris que l'on ne pourrait pas laisser nos niveaux avec un aspect esthétique aussi pauvre. Grâce à l'asset que nous avons trouvé sur le Unity store, j'ai pu me mettre rapidement à la re-modélisation du niveau 2 (voir annexes FIGURE 2), balayant par la même occasion le problème d'échelle. J'ai travaillé avec deux fenêtres Unity pour modéliser de nouveau le niveau 2 : l'ancienne scène du niveau 2 avec les blocs blancs, et celle que je m'apprêtais à modéliser avec l'asset. Cela m'a permis d'être plus rapide car je pouvais voir exactement quelle plateforme mettre à quel endroit. Ainsi, j'ai fini rapidement la modélisation de mon niveau, ce qui m'a permis de passer à la conception et la modélisation du niveau 3.

2.3 Niveau 3

Laure

Pour le niveau 3, Gaël et moi avons appelé Louis en renfort concernant la conception. En effet, après le temps gaspillé à modéliser deux fois les niveaux 1 et 2, il fallait que nous soyions efficaces pour rester dans les temps. Nous avons réfléchi à un niveau type "infiltration", c'est-à-dire qu'il fallait y implanter des IA qui empêchent une résolution trop linéaire du niveau en détectant la présence des joueurs et en leur mettant des dégâts. La conception s'est comme d'habitude faite dans un premier temps en 2D à l'aide d'un croquis. Ensuite, je me suis occupée de modéliser en 3D (voir annexes FIGURE 3) sur Unity, ce qui s'est une fois de plus révélé

casse-tête. En effet, nous avons encore eu ce problème de triche possible dû à la possibilité de passer d'une plateforme à l'autre alors que cela ne devrait pas être possible, il a donc fallu augmenter les distances et modifier l'idée originale. De même, l'idée de départ que nous avions reportée sur papier ne semblait plus convenir : les deux personnages n'évoluaient pas en même temps dans le niveau. C'est-à-dire que lorsque l'un s'occupait d'activer un bouton pour permettre d'avancer, l'autre n'avait rien à faire à part attendre. Ensuite, les rôles s'inversaient. Le problème était donc de rendre la résolution du niveau active pour chaque personnage à chaque instant. Pour solutionner cela, j'ai choisi de séparer la carte en deux : le personnage masculin irait sur la gauche, et le personnage féminin sur la droite. Malgré l'apparente distance et le chemin différent que les deux personnages prennent, ils contribuent en fait tous les deux à la résolution du niveau : le garçon en actionnant un bouton permettant de dégager le passage pour la suite du niveau, et la fille permettant de faire disparaître l'IA qui bloque le passage. Ils se retrouvent ensuite pour poursuivre jusqu'à la fin du niveau, où la partie musicale se déclenche.

2.4 Niveau 4

Gaël

Le niveau 4 a constitué un travail théorique lui aussi important, mais un travail pratique en revanche moins technique. En effet, tous les scripts ayant déjà été réalisés, et la modélisation étant devenu un travail que je maîtrisais de mieux en mieux, le travail concret fut donc moins technique que celui du niveau 1. En revanche, la conception du niveau quatre s'est avérée quant à elle être plus technique qu'auparavant.

Ce dernier niveau devait être la conclusion de notre jeu, il devait donc réunir toutes les mécaniques de jeu, être difficile, et aussi novateur. Nous avons donc imaginé un niveau qui évoluerait en spirale, et où le garçon et la fille verraient leur niveau évoluer de façon symétrique. Ainsi,

les deux spirales sont reliées par des mécaniques de Gameplay, mais pas physiquement, car le but ici est de pousser le joueur à observer le niveau. En effet, chacun des deux joueurs doit être aidé de l'autre pour progresser dans son niveau, et c'est cette idée qui a été en quelque sorte notre ligne de pensée durant la création de ce niveau. Tout au long du design du niveau, j'ai donc appliqué ces idées et fait évoluer les éléments de Gameplay afin d'offrir aux joueurs une expérience nouvelle et plus difficile que celles qu'ils ont rencontrées précédemment. En effet, en plus des éléments de Gameplay du présents dans les autres niveaux, une notion d'esquive plus évidente a été mise en place. Ce nouveau concept impose au joueurs d'avoir à esquiver les projectiles des ennemis ainsi que les ennemis eux mêmes lors de son cheminement dans le niveau. Bien sûr, certains éléments tels que des murs ou des plateformes permettent aux joueurs d'esquiver l'ennemi. La modélisation du niveau quant à elle a été plus facile que pour le premier niveau, mais a été plus longue car le niveau 4 est beaucoup plus grand que le premier.

Au final, le travail effectué sur le niveau 4 s'est avéré être plus amusant que celui du niveau 1, car il nécessitait moins de technique (au sens où les techniques utilisées avaient déjà été maîtrisées auparavant grâce à la conception des autres niveaux) et plus de création. Ce travail a donc été instructif car il m'a appris à faire évoluer le gameplay sans en modifier foncièrement les éléments.

2.5 Travail sur les lumières & textures

Louis

Le travail sur les lumières est arrivé assez naturellement entre la première et la deuxième soutenance. En effet, le jeu devenait de plus en plus joli avec l'ajout des différentes textures et des modèles 3D en jeu. Ainsi, j'ai décidé de travailler sur ces lumières. Nous souhaitons dès le début du projet avoir une ambiance assez sombre pour notre jeu. J'ai

donc décidé de travailler sur une scène sur laquelle j'ai fait le maximum de tests de couleurs et d'intensité des lumières sur nos éléments de décor. De ces nombreux tests sont ressorties 2 couleurs principales : le bleu et le rouge. J'ai ensuite cherché une manière de les implémenter dans le jeu. J'ai fait quelques recherches afin de voir comment sont implémentées les lumières dans les jeux vidéo aujourd'hui, et surtout dans les jeux de type platformer. J'ai alors décidé de créer des petites pyramides sur lesquelles j'ai déposé les orbes de couleur, disponibles grâce à l'asset téléchargé, qui émettent la lumière souhaitée. De plus, j'ai décidé de rajouter des lumières beaucoup plus importantes pour les téléporteurs. En effet, ce sont des éléments très importants qui doivent être visibles le plus rapidement possible par le joueur afin de comprendre et d'analyser le niveau. J'ai alors décidé de colorer le portail de départ en bleu et le portail d'arrivée en rouge, afin que le joueur comprenne plus facilement ses déplacements sur le niveau. J'ai ensuite eu à décider s'il fallait "Baker" les lumières ou de les mettre en "Realtime". J'ai donc testé les deux, j'en ai discuté avec les membres du groupe et nous avons convenu de les laisser en RealTime. Le rendu se rapprochait plus de ce que l'on souhaitait pour notre jeu et cela ne posait pas de problème de performance en jeu.

Lors de mes tests sur ma scène composée de seulement quelques éléments de notre jeu, je me suis rapidement rendu compte que les calculs à réaliser par l'ordinateur afin de retranscrire les lumières étaient très nombreux. La question de l'optimisation et de la qualité s'est alors posée. J'ai donc baissé la qualité des lumières sur ma scène jusqu'à ce que je ne sois satisfait du rendu. J'ai alors divisé par 4 le nombre de calculs, tout en gardant un résultat très satisfaisant. Il était maintenant temps de placer tout ces éléments dans les niveaux du jeu. J'ai alors commencé par le niveau 2 qui était le plus avancé à ce stade, puis le niveau 1 et enfin le 3 puis le 4. Le niveau 2, qui a été le premier sur lequel j'ai ajouté la lumière, m'a pris un certain temps à réaliser. En effet, malgré les tests réalisés avant de travailler sur le niveau en lui-même, il a fallu faire de

nombreux ajustements. Tout d'abord, la puissance d'émission des lumières : en effet, sur la scène de test, je n'avais que très peu d'objets et je travaillais sur un espace restreint. Sur un niveau entier, il fallait plus de puissance et donc augmenter l'émission des lumières. Cependant, si j'augmentais l'émission des lumières, leur couleur devenait plus claire, et finalement trop claire. Il a donc fallu retravailler les différentes couleurs pour que celles-ci ne soient pas trop claires tout en restant suffisamment puissantes pour illuminer le niveau. Une fois le niveau 2 entièrement "mis en lumière", le niveau 1 fut plus simple et plus rapide à réaliser. Les niveaux étant tous à la même échelle, je n'ai eu qu'à reproduire ce que j'avais déjà réalisé dans le niveau 2. J'ai ensuite travaillé sur le niveau 3 qui m'a pris beaucoup plus de temps car il était le premier à devenir réellement complexe dans son architecture. Il a donc demandé un peu plus de réflexion pour le placement des lumières. Par exemple, je ne souhaitais pas placer de lumières sur les plateformes mouvantes de peur de créer des problématiques lorsqu'elles bougent. Une fois les lumières réglées pour le niveau 3, j'ai simplement refait la même chose pour le niveau 4. Afin d'obtenir la teinte de nuit que nous souhaitions, Laure et moi avons décidé d'enfermer les niveaux dans une boîte sur laquelle nous avons appliqué une texture réalisée par Laure. Ainsi, le niveau se retrouvait entièrement dans l'obscurité, avec pour seules sources de lumière les éléments que j'avais implémentés. Cependant, ces éléments font principalement partie du décor et ne suffisent pas à illuminer le niveau. La luminosité dans le niveau était devenue un problème. En effet, pour un jeu que nous souhaitons fortement tourné vers la communication, il fallait que les joueurs puissent se voir entre eux tout au long du niveau. Nous avons alors décidé d'ajouter une lumière ambiante en haut de notre boîte qui contient le niveau. Nous avons donné une teinte bleue très sombre à cette lumière, toujours dans l'optique d'obtenir une ambiance nocturne. Nous avons réussi à obtenir le résultat souhaité, tout en optimisant un maximum le rendu. Cela facilite même la vision du joueur dans le niveau. Cela lui permet de comprendre rapidement comment le niveau est

construit et où sont les éléments importants qu'il ne faut pas rater sous peine de ne pas pouvoir finir le niveau.

Laure

Pour terminer concernant cette partie consacrée au Level Design, nous nous intéresserons aux textures qui constituent la toile de fond des niveaux. En effet, les scènes sur Unity ont été créées dans l'environnement basique, c'est à dire un gris uniforme. Or, dans le même souci de rendu graphique que lors du travail sur les lumières, il nous fallait un beau fond pour l'environnement du niveau. Nous avons donc eu l'idée de créer nous mêmes une texture pour l'environnement des niveaux. Habitée à manipuler des outils pour créer des textures, ou bien modifier des textures disponibles gratuitement sur la plateforme DeviantART pour le besoin de mes dessins, je me suis penchée sur le sujet. Pour rester dans les tons de couleurs que nous avons convenu, je me suis appuyée sur deux textures libres de droit de l'artiste Sirius-sdz sur DeviantART, que j'ai modifiées concernant la balance des couleurs et l'opacité de manière à les superposer. Par dessus, à l'aide d'abord de logiciel de dessin Paint Tool SAI et par la suite de Photoshop, j'ai ajouté quelques effets grâce à différentes formes de brosses sur les logiciels. Une fois les textures terminées, je les ai introduites sur les scènes Unity grâce au 3D object "Quad" en les plaçant en carré autour des plateformes, au dessus et en dessous pour avoir un bloc.

3 Multijoueur

Louis

Très vite dans le démarrage du projet, j'ai compris l'importance majeure de l'implémentation du multijoueur dès le début de la création du

jeu. En effet, il s'agissait de la première brique sur laquelle toutes les autres allaient devoir être empilées. C'était une partie essentielle et indispensable à notre jeu qu'il fallait comprendre et maîtriser afin de ne pas bloquer sur la suite de la création. Ainsi, dans les premières semaines qui ont suivi la validation du cahier des charges, j'ai tout d'abord appris à manier Unity puis j'ai commencé à travailler sur le multijoueur. J'ai décidé de ne pas utiliser d'assets pour comprendre comment fonctionne le multijoueur sur Unity. J'ai commencé par mettre des personnages en réseau et à les synchroniser entre eux. Cela m'a permis d'apprendre à manier Unity et de comprendre les différentes parties importantes du multijoueur. J'ai suivi de nombreux tutoriels qui m'ont appris différentes facettes de Unity, que ce soit la création d'un menu, la gestion des dossiers, les scripts de déplacements et bien d'autres. Cette prise en main de Unity m'a semblé plutôt simple, ce qui m'a énormément rassuré à ce stade pour la suite du projet. Cependant il nous fallait un outil plus puissant pour gérer le multijoueur. Nous avons recherché différents assets pour simplifier et pour optimiser le multijoueur sur notre jeu. De tous ces différents assets, nous avons retenu Photon. Depuis, nous travaillons exclusivement sur Photon, qui est largement suffisant pour nous. Romain a ensuite pris le relais sur la création du multijoueur en utilisant Photon.

Romain

Pour la première soutenance, je me suis occupé des scripts Photon qui permettent de se connecter à un serveur Photon depuis n'importe quel ordinateur, et de faire apparaître un personnage contrôlable par le joueur et visible par les autres personnes connectées au serveur.

Avec Photon, j'ai réussi à faire apparaître et bouger un personnage issu d'un asset (ses commandes et animations aussi), ce qui nous a été utile jusqu'à ce que le modèle de Laure et ses commandes soient terminés. J'ai réalisé toutes les modifications nécessaires sur les prefabs des joueurs que l'on voulait faire apparaître pour que le système Photon dé-

marre bien et fasse ce que l'on voulait. Le plus compliqué a été de faire en sorte que tout se synchronise correctement par la suite.

Le multijoueur est devenu de plus en plus complexe avant la deuxième soutenance, avec des erreurs qui ne devaient tout simplement pas exister, mais qui étaient bel et bien présentes. J'ai commencé par essayer de trouver un moyen de faire apparaître les joueurs lors du chargement de la scène du niveau, car lors de la première soutenance, la scène qui succédait à celle du menu chargeait mais les joueurs n'apparaissaient pas. Le problème venait du fait que quand la scène chargeait, les joueurs étaient déjà connectés au serveur, et donc aucun de mes scripts ne disait qu'il fallait faire réapparaître les joueurs après le changement de scène. J'ai donc modifié mes scripts en conséquence, mais un deuxième problème est apparu : comment garder les choix des joueurs effectués dans le menu (choix des personnages), faire apparaître dans le niveau les bons prefabs pour chaque joueur ? Une solution m'a parue faisable : garder, lors des changements de scène, le `GameObject` qui gère les connexions au serveur Photon, et l'apparition des joueurs. Pour résumer, il fallait garder l'objet auquel j'ai attaché les scripts fondamentaux et le modifier pour qu'il contienne toutes les informations nécessaires au fonctionnement du jeu. Ainsi, j'ai créé une classe qui sera utilisée aussi par d'autres scripts pour avoir une base de données statique tout au long du jeu.

Cependant, la commande proposée par Photon permettant de synchroniser les scènes ne fonctionnait pas correctement : les deux joueurs chargeaient la scène, mais seul le premier joueur voyait le second (c'est-à-dire était en synchronisation avec), le deuxième étant quant à lui - de ce que j'ai pu en déduire - dans la même "Room", mais dans une scène différente (le deuxième joueur n'était pas seulement invisible pour le premier, il n'était tout simplement pas là, comme s'il était en partie solo). C'est donc après beaucoup d'heures de recherches infructueuses et pénibles sur le problème que j'ai décidé de ne pas détruire les prefabs des

joueurs et de reprendre plutôt mes scripts de réseau en entier. Cela voulait également dire en abandonner certains. Au final, j'ai mieux compris le fonctionnement interne de Photon au cours de mes recherches, même si cela ne m'a pas été utile pour résoudre les problèmes. J'ai appris à "surcharger" les fonctions de Unity pour ajouter mes propres exécutions lors des changements de scène par exemple, réorganiser proprement autant mes scripts réseau que mon script de mouvement du joueur. Le choix des personnages, et donc les changements de prefab lors du chargement du niveau, marchaient pour la deuxième soutenance mais seulement dans certains cas, des problèmes devaient encore être résolus.

Pour notre dernière soutenance et donc pour terminer le jeu, j'ai dû me replonger dans ces scripts de réseau. J'ai fini par trouver comment résoudre les soucis de changement de prefab en modifiant, entre autres, le moment où étaient détruits les personnages. Afin d'augmenter la convivialité de notre jeu, j'ai implémenté un mode de jeu où les deux personnes jouent sur le même ordinateur, le même clavier. L'écran est scindé en deux quand ce mode de jeu est activé.

4 Niveau Musical

4.1 Les mélodies

Gaël

Dans le niveau musical, mon travail a consisté en composer les différentes mélodies des niveaux musicaux. Ce travail a été initié d'abord par une volonté commune du groupe de mélanger jeu de plateformes et jeux musicaux motivée par notre passion commune pour la musique. D'un commun accord, nous avons imaginé que les niveaux musicaux se trouveraient en fin des niveaux de plateformes, et qui seraient en quelque sorte l'aboutissement final du parcours effectué par le joueur, un peu

comme l'ultime ennemi de chaque niveau. Nous avons choisi que l'instrument que nous utiliserons pour jouer ces mélodies serait le piano, et cela pour deux raisons : d'abord par amour commun pour cet instrument, mais aussi pour la facilité que l'on peut avoir à enregistrer un morceau grâce au clavier MIDI.

Tandis que Louis s'occupait de créer le niveau musical, je me suis occupé de la composition des mélodies de ce dernier. L'idée était d'en composer une par niveau, il a donc fallu que je réfléchisse à un style musical pour les mélodies afin qu'il y ait une continuité entre les différentes musiques. Afin d'isoler un style musical qui me semble en adéquation avec l'histoire du jeu et son évolution, j'ai écouté de nombreuses musiques différentes, venant de nombreux styles différents. En effet, j'ai écouté la *Sarabande* de Haendel à *For a Better Day* d'Avicii, en passant par *The Orange Tree* de Philip Glass (la liste des morceaux que j'ai écoutés est disponible en annexe). Ce long travail d'écoute a représenté plus d'une douzaine d'heures et s'est avéré très instructif, car même si je connaissais une bonne partie des morceaux que j'écoutais, certains étaient des découvertes pour moi, souvent très agréables. J'ai donc isolé plusieurs morceaux qui m'inspiraient et semblaient en adéquation avec ce que je souhaitais comme style musical pour les mélodies. Tout d'abord, j'ai écrit une première mélodie inspirée du morceau de Kid Cudi, *Dat New*, mais suite à une concertation avec Louis nous avons décidé de ne pas utiliser cette dernière mélodie, car son tempo était trop rapide. J'ai réajusté mon idée du style musical que je souhaitais proposer pour le niveau musical. Pour la deuxième mélodie, j'ai décidé de m'inspirer du morceau de Elvis Presley, *Why I Can't Help Falling in Love With You*. Ce second morceau terminé, je l'ai envoyé au groupe. On a donc décidé de le mettre pour le premier niveau. La troisième musique que j'ai composée s'est cette fois-ci inspirée d'une composition de Geoff Zanelli et utilisée par Kid Cudi dans *Dat New*. La quatrième composition est inspirée de la *Tempête* de Beethoven ainsi que du morceau *Beetwen men and Trees*

de Ezio Bosso (ce morceau est aussi connu pour avoir été samplé par David Guetta dans le morceau Dangerous). La cinquième mélodie quant à elle est inspirée du thème de la série The Leftovers, composé par Max Richter. Quand je dis m'être inspiré d'un morceau, cela peut être par sa construction rythmique (comme les 3 temps avec une valse), par les accords ou bien encore par les thèmes qu'ont pu m'inspirer ces dernières. Prenons par exemple la quatrième mélodie, je me suis inspiré de l'écart des notes dans le thème de Beethoven mais en l'adaptant à une autre construction rythmique qui est celle du morceau de Bosso. Même si je l'ai faite avec ma vision et ma touche personnelle, ce travail de composition, comme beaucoup d'autres d'ailleurs, a des influences et inspirations qui sont traçables si l'on y prête attention.

Au final ce travail de composition a été très instructif pour moi. Bien qu'il ne m'apporte rien directement dans ma formation d'ingénieur en informatique, il m'a appris beaucoup indirectement. Ça a été un travail long et fastidieux, demandant beaucoup d'essais et de patience mais qui fut aussi instructif et plaisant de part son côté créatif et sa malléabilité.

4.2 Création du niveau

Louis

Notre jeu est composé de 2 grandes parties : une partie platformer, et une partie musicale dont j'ai décidé de m'occuper. Cette partie musicale avait pour but de ressembler aux différents jeux très connus qui utilisent comme mécaniques celle de Guitar Hero ou même de Osu!. Pour rester dans le thème du jeu, il fallait que cette partie reste accessible au plus grand nombre et puisse être agréable à jouer. Pour cela, il fallait des mécaniques simples à comprendre, un design simpliste et épuré et des mélodies agréables à écouter.

J'ai réfléchi à la manière d'implémenter le niveau musical et surtout au moyen de le faire. Tout d'abord, cette partie devait être en 2D. Ensuite, il me fallait une mélodie. J'ai alors travaillé avec Gaël qui est pianiste et qui a pu me fournir une mélodie assez rapidement. Ensuite, il fallait que je construise le niveau, que je place les différents éléments. Tout d'abord, ce que j'ai appelé des "activateurs". Ce sont les éléments placés à la base de l'écran du joueur. Il s'agit de cercles alignés, au nombre de 7 comme les 7 notes d'une gamme (do, ré, mi, fa, sol, la, si). J'ai ensuite ajouté à ma scène quelques cercles au-dessus des activateurs qui seront mes notes de test. Il fallait maintenant que j'ajoute des scripts à ces différents éléments afin qu'ils fassent ce qu'ils sont censés faire. J'ai commencé par me dire qu'il me suffisait de rendre les notes sensibles à la gravité, qu'elles allaient ainsi tomber naturellement dans le bon sens sans avoir besoin de script. J'ai donc ajouté un rigidBody à mes notes de test, que j'ai ensuite testées sur la scène. Cela fonctionnait, cependant, je voulais pouvoir gérer facilement la vitesse de chute de mes éléments sans changer la gravité. J'ai ainsi décidé de ne pas utiliser la gravité pour les notes, mais d'utiliser un script qui déplace ma note vers le bas et sur lequel je pouvais choisir la vitesse de chute en changeant simplement une valeur sur Unity. Je possédais maintenant des notes fonctionnelles sur lesquelles je pouvais choisir leur vitesse de chute. Je devais ensuite m'occuper des activateurs et de leur script. J'ai donc écrit une liste des choses que devait faire mon activateur : récupérer la note et la supprimer si celle-ci avait été gagnée, être associée à une touche du clavier, détecter la note et ajouter des points au score du joueur. J'ai donc décidé tout d'abord d'associer chacun de mes 7 activateurs à une touche du clavier. Le plus naturel m'a semblé d'utiliser les touches "azertyu". Pour vérifier que chaque touche était associée au bon activateur, j'ai décidé d'ajouter quelque chose : lorsque l'on appuie sur la touche associée à l'activateur, cet activateur passe de sa couleur initiale au gris durant 0,2 secondes. Cela permet aussi au joueur de savoir sur quelle touche il a appuyé et quel

activateur il a activé. Ensuite, il fallait savoir si le joueur avait appuyé au bon moment et sur la bonne touche. Ainsi, si ces conditions étaient vérifiées, le joueur gagnait en score et en "notes réussies". De plus, la note est détruite car elle a été gagnée par le joueur. Si ces conditions ne sont pas vérifiées, le joueur ne gagne pas de score, son multiplicateur est réinitialisé, et la note n'est pas détruite. C'est ainsi qu'un nouveau problème s'est présenté à moi. Que faire des notes manquées ? J'ai alors décidé de tout simplement les détruire peu de temps après leur passage sur l'activateur.

J'ai ensuite ajouté au niveau des éléments plutôt décoratifs ; j'ai ajouté des couleurs aux activateurs et un canvas sur lequel on peut lire le score du joueur en temps réel, mais aussi son multiplicateur. De plus, j'ai remarqué que l'emplacement de la note n'était pas très lisible et il était ainsi difficile de savoir sur quelle touche appuyer au bon moment. Pour pallier à ce problème, j'ai ajouté des bandes noires un activateur sur deux. Cela augmente énormément la lisibilité des notes pour le joueur. Mon niveau était ainsi prêt à être associé à la musique que Gaël m'avait fournie. Cependant, comment faire pour créer autant de notes au bon tempo sans perdre une semaine entière ? J'ai alors décidé d'inverser le niveau. En effet, à la place de détruire les notes, j'allais les créer en appuyant sur les activateurs. J'ai alors complété le script des activateurs afin d'y ajouter un mode de création. Une fois ce mode activé, il me suffisait d'ajouter la musique à la scène et d'appuyer sur les touches de mon clavier au bon moment. Ainsi, je possédais toutes mes notes en dessous de mes activateurs. Je n'avais qu'à les copier puis les coller au-dessus de mes activateurs. Il m'a ensuite suffi de rectifier quelques notes, les déplacer pour rendre le niveau plus ou moins difficile en fonction des besoins. J'ai ensuite répété cette opération plusieurs fois pour les différents niveaux musicaux présents sur le jeu.

A présent, la question était la suivante : A quel moment convenir que le joueur gagne ou non la partie ? Nous en avons discuté avec les membres du groupe. Plusieurs possibilités s'offraient à nous. La première était d'utiliser le score atteint à la fin du niveau. Cependant, cette mécanique posait un problème : pour avoir un score moyen et donc un score à atteindre, il fallait prendre en considération la possibilité d'obtenir des multiplicateurs de score. Un joueur qui réussirait à ne toucher que 2 notes sur 3, et donc n'aurait pas la possibilité d'obtenir un multiplicateur, aurait beaucoup plus de mal que quelqu'un qui réussirait 10 notes à la suite. Une autre possibilité était de créer un système de vies que l'on perd à chaque note ratée. C'était une bonne idée et un concept très utilisé dans les jeux de ce type, cependant nous trouvions cette mécanique trop punitive pour une partie du jeu qui n'a pas pour vocation de l'être. La dernière solution présente sur le jeu : le joueur gagne la partie s'il réussit un nombre de notes précis durant le niveau. J'ai donc travaillé, répété plusieurs fois le niveau et même fait essayer les différents niveaux à des personnes extérieures afin de déterminer un nombre de notes ni trop simple, ni trop difficile à atteindre pour n'importe quel joueur. J'ai réussi à trouver un juste milieu pour chacun des niveaux. Les niveaux musicaux avaient ainsi une fin. Et les joueurs un objectif à atteindre pour gagner la partie. Il ne restait plus qu'à les intégrer dans le jeu à la fin de chaque niveau.

5 Menu

Louis

Peu après le lancement du projet et la validation du cahier des charges, j'ai commencé à travailler sur le menu. Cela m'a permis de continuer à prendre en main Unity et surtout la manipulation des Canvas que j'ai ensuite eus à utiliser, par exemple, dans le niveau musical.

J'ai appris à créer des boutons fonctionnels en suivant différents tutoriels et en essayant plusieurs choses car je savais que j'allais en avoir besoin dans ce que j'avais à réaliser plus tard dans le projet. J'ai donc conçu un premier menu simple, avec un bouton "Quitter" et un bouton "Jouer". Nous en avons ensuite discuté avec les autres membres du projet, afin de décider de la manière dont nous voulions intégrer le menu dans le jeu, mais aussi des différents éléments qui seront présents dans ce menu. Nous avons donc convenu d'avoir un menu principal sur lequel les deux joueurs peuvent commencer à prendre en main le jeu. Nous avons décidé de placer le menu principal sur le côté de l'écran afin que les joueurs puissent jouer sur une scène très simple en attendant les choix de l'autre joueur. Sur ce menu, nous avons ajouté un menu déroulant afin de choisir le niveau sur lequel nous souhaitons jouer, une partie consacrée au choix du personnage (fille ou garçon), et enfin un bouton "Jouer" et "Quitter" afin de lancer la partie ou de quitter le jeu.

Romain

Pour la deuxième soutenance, j'ai modifié l'environnement du menu concernant le choix du niveau et des personnages pour qu'il soit plus agréable à regarder qu'un simple environnement basique de Unity avec des plans blancs. J'ai également modifié l'apparence du menu pour qu'il s'adapte mieux à la taille de l'écran, et différencié les canvas/possibilités de chaque joueur dans le menu (à savoir le choix des personnages et les options pour jouer ou quitter le jeu), ajouté une fonctionnalité au bouton "Escape" en jeu, et codé un script pour les Toggles de choix du personnage qui jusque là ne fonctionnaient pas totalement. J'ai aussi fait un script qui lit le nombre de points de vie restants du joueur pour l'afficher sur l'écran. De plus, La première impression du jeu est importante, c'est pourquoi j'ai pris un peu de temps pour améliorer le design du menu qui est plus en accord avec notre police d'écriture dans le jeu. Aussi, le slider pour le volume de la musique, des bruitages est maintenant opérationnel.

Pour finir, dans le menu nous pouvons choisir le niveau dans lequel nous voulons aller, c'est pourquoi il a fallu que j'ajoute une fonctionnalité qui permet de bloquer les niveaux auxquels on ne pourra pas accéder tant qu'on n'a pas réussi le précédent. De plus, je me suis occupé de la coordination entre les différentes scènes, niveaux ; ces tâches furent complexes, mais nous étions obligés d'avoir ces éléments pour présenter un jeu complet.

6 Modélisation 3D & Animation

La modélisation 3D et l'animation de nos personnages étaient des aspects de notre jeu qu'il nous était cher de faire nous-même lors de l'élaboration du cahier des charges. Nous voulions un jeu qui soit au maximum fait par nous, avec nos forces et nos faiblesses, de manière à explorer tous les rôles nécessaires à la création d'un jeu vidéo et comprendre ce qui nous plairait le plus. Finalement, la modélisation et l'animation se sont principalement concentrées sur les personnages de notre jeu, par manque de temps et d'expérience. Cela a été un travail très long et fastidieux, avec beaucoup de complications et de retard.

6.1 Modélisation 3D

Laure

Je me suis occupée de toute la modélisation des personnages seule, ainsi que des bonus que l'on peut retrouver dans les niveaux. Il était assez compliqué en effet de travailler à deux sur la modélisation, car il aurait fallu que chacun prenne en main le logiciel et avance tour à tour sur la modélisation. Cela aurait forcément conduit à des discords concernant les différentes techniques et la conception de chacun sur ce qu'il voulait faire. Ainsi, j'ai décidé de m'en occuper seule, ce qui a demandé énormément de travail et de temps. D'abord de recherche, car n'ayant jamais

utilisé Blender auparavant, j'avais besoin de comprendre l'interface du logiciel, ses nombreux raccourcis claviers (pas si raccourcis que cela...) qui sont indispensables pour éviter d'utiliser les menus déroulants à chaque fois.

J'ai d'abord voulu tester Blender avec quelque chose de moins complexe qu'un corps humain. Nous voulions également des objets collectables pour augmenter le score à la fin du niveau : c'est donc naturellement que j'ai commencé par travailler sur la modélisation de ces collectables. Notre jeu étant centré autour de la musique, j'ai voulu modéliser un collectable qui aurait la forme d'une note de musique. Cela n'a pas été trop fastidieux : j'ai pris un objet 3D basique de Blender, à savoir une sphère, que j'ai étendue légèrement de manière à obtenir la forme d'un œuf pour constituer la tête de la note. Ensuite, j'ai utilisé la fonction "Extrude" pour créer la hampe, puis ai "Extrude" encore une fois pour créer la durée. Le seul souci rencontré lors de cette modélisation a été la duplication de tous les points qui constituaient la note à chaque réouverture du fichier, ce qui m'empêchait à chaque fois de modifier correctement son aspect. Il m'a fallu enlever les points double à chaque fois tout autour de la note, ce qui m'a ralentie, mais heureusement comme la modélisation n'était pas trop compliquée je l'ai finie rapidement et ai pu passer à la modélisation des personnages.

Comme notre jeu se joue en coopération entre deux joueurs, nous voulions respecter la parité et modéliser donc un personnage masculin, et un personnage féminin. Dans un premier temps, j'ai réalisé des croquis pour fixer le *chara design* des personnages (voir annexes FIGURE 4) que j'ai réalisé en premier sur le logiciel Paint Tool SAI à l'aide d'une tablette graphique Wacom. Il fallait que les tenues de nos personnages soient en accord avec le contexte de notre jeu, c'est-à-dire l'infiltration dans le château du roi qui a volé les artefacts. Ainsi, il ne fallait pas quelque chose de trop voyant ou extravagant. Cependant, il nous fallait

tout de même les différencier au niveau des couleurs : nous avons opté pour du rouge pour le personnage masculin, et jaune pour le personnage féminin. A l'origine, les deux personnages devaient avoir une capuche sur la tête, ainsi que des combinaisons légèrement différentes. Malheureusement, cela aurait été trop difficile et chronophage à modéliser, donc nous avons dû revoir nos exigences à la baisse et simplifier le design. Finalement, nous avons opté pour le même uniforme pour garçon et fille, enlevé la capuche et simplement différencié les deux personnages par la couleur de leurs vêtements et leurs cheveux.

Après tous ces moments d'apprentissage et de réflexion sur ce que l'on voulait, j'ai commencé par modéliser un "squelette" de personnage, qui me servirait de base sur laquelle travailler pour différencier ensuite chaque personnage. A partir d'un cube et avec la fonction Mirror très pratique, j'ai pu réaliser un « corps » de base en entier, dans le style « cartoonish » que nous voulions donner à nos protagonistes. Avoir travaillé sur le *chara design* en 2D m'a permis de gagner du temps, car cela m'a guidée dans les proportions et l'aspect général du personnage une fois sur Blender et j'ai pu y glisser les croquis en arrière plan et modéliser par dessus. Une fois le squelette modélisé, je me suis penchée dans un premier temps sur la modélisation du personnage masculin. Il a fallu créer à partir du squelette tous les vêtements. J'ai utilisé majoritairement la fonction "extrude" pour les faire ressortir : la veste, le pantalon, les bottes et les gants. Je n'aurais pas cru cette étape aussi longue et fastidieuse, mais le fait est que j'y ai passé plus de temps qu'il n'aurait fallu. Un problème dont je n'ai jamais pu trouver la cause m'a également beaucoup ralenti : en reprenant mon travail là où je l'avais laissé, j'ai retrouvé mon modèle 3D troué au niveau de la tête. Évidemment, j'ai tout de suite voulu boucher ce trou en utilisant "F", qui ajoute une face automatiquement lorsque l'on sélectionne plusieurs points du modèle. Malheureusement, malgré toutes mes tentatives sous différents angles, je me suis toujours retrouvée avec des bugs : les faces n'étaient pas plates

et alignées avec le reste des faces. Lorsque je tournais le modèle sous un autre angle, la face n'était pas plate, comme s'il y avait un bug de texture. Après une bonne heure de recherches infructueuses sur Internet, je me suis finalement résignée à essayer une fois de plus de combler le trou, cette fois-ci en passant à l'intérieur du modèle. Comme celui-ci était uniquement constitué de faces dans l'espace, l'intérieur était donc creux et j'ai pu zoomer assez pour passer à l'intérieur. De là, j'ai réessayé de créer une nouvelle face, et miracle : cela a fonctionné. Après cet incident, j'ai voulu m'attaquer à la modélisation des cheveux du personnage masculin : je savais que le challenge résiderait en la mèche plus volumineuse que le reste que j'avais prévu de faire au moment du *chara design*. Effectivement, cela m'a pris du temps de tout modeler à partir d'un carré simple. Qui plus est, j'ai dû l'intégrer ensuite avec le reste des cheveux, que j'ai partiellement modelé à l'aide d'un autre cube modifié que j'ai pris trop volumineux. J'ai donc eu trop de matière à travailler, ce qui m'a ralenti à cause des trop nombreux ajustements que j'ai dû réaliser. J'aurais sans doute dû créer la totalité des cheveux du garçon à partir d'un seul cube, car il a été difficile d'assembler les deux parties de manière harmonieuse. J'ai ensuite appliqué la couleur sur les vêtements : le plus long a finalement été de sélectionner toutes les faces à colorer de telle ou telle couleur.

Après avoir terminé le personnage masculin, je suis passée au personnage féminin. Ayant pris soin de séparer le corps des cheveux pendant la modélisation du garçon, j'ai pu exporter la modélisation terminée du personnage masculin au format .fbx avant de rendre invisible la modélisation de ses cheveux pour passer à celle de la fille. J'ai pu directement changer la couleur des vêtements en re-sélectionnant la partie rouge pour y appliquer du jaune à la place. J'ai terminé par la modélisation des cheveux de la fille, qui a été plus simple car n'ayant besoin que d'un seul bloc, et ayant une forme plus simple. Cela m'a pris moins de temps et a demandé moins de technique. Après le travail fastidieux qu'a été la modélisation des personnages (voir annexes FIGURE 5 image de gauche)

, il a ensuite fallu passer à leur animation.

6.2 Animation

Laure

L'animation des personnages a pris beaucoup de retard, car dépendait évidemment de l'avancée sur la modélisation des personnages, qui a pris beaucoup plus de temps que prévu. Ainsi, il nous a fallu minimiser le travail d'animation, pour avoir seulement le nécessaire, c'est-à-dire un cycle de marche et une animation de saut. Pour éviter de gaspiller encore plus de temps, je me suis reposée essentiellement sur un tutoriel disponible sur YouTube, le même que celui que j'avais regardé pour me renseigner sur les bases de la modélisation 3D sur Blender. La série de vidéos comprenant une partie sur l'animation, j'en ai profité.

Dans un premier temps, il a fallu créer un véritable squelette aux modèles 3D (voir annexes FIGURE 5 image de droite). En effet, il fallait que chaque partie du corps puisse être déformée pour créer un mouvement, ce qui devait passer par la création d'os en interaction les uns avec les autres. J'ai donc créé les principaux os des bras et des jambes, deux pour le tronc et un pour la tête du personnage. Cette partie n'a pas été très compliquée, mais ce qui a suivi, à savoir le raccordement de tous les os entre eux pour créer un mouvement naturel, l'a été. Surtout au niveau des articulations des jambes et des pieds, j'ai dû suivre le tutoriel à la lettre : après avoir raccordé le pied au mollet, j'ai travaillé sur la position que devait prendre le pied lorsque le personnage plierait le genou. Il fallait que le pied reste parallèle au sol pour mimer un mouvement humain, ce qui n'était pas la position par défaut lors du rattachement des os entre eux. J'ai dû rajouter un os en guise de plante de pied pour guider l'inclinaison du pied, ainsi qu'un point de mouvement derrière le talon et un point "cible" pour guider le mouvement du genou. Tout ceci a été fait dans un premier temps sur la partie droite du squelette du

personnage féminin. J'ai donc dupliqué mon travail et l'ai appliqué à la partie gauche du modèle pour obtenir une symétrie parfaite et indispensable au bon fonctionnement du squelette.

Une fois la partie concernant la création du squelette terminée, il a fallu gérer quelques imperfections lors de la mise en mouvement, particulièrement la rotation des bras et des jambes. En effet, selon le placement des os que j'avais choisi, le modèle suivait bel et bien le mouvement, mais cela donnait quelque chose qui n'était pas naturel : par exemple au niveau de l'épaule, lorsque le bras était levé il se formait un creux entre le cou et l'épaule qui n'aurait pas dû exister. Par conséquent, il a fallu que j'ajoute du "poids" sur certaines parties du modèle. J'ai d'abord activé une fonction miroir qui permettrait de calquer le travail que j'effectuais d'un côté du modèle sur l'autre côté pour gagner du temps et pour obtenir une symétrie totale. Pour le cas de l'épaule, il a fallu en rajouter au niveau de l'os du tronc pour réduire le creux, mais également au niveau de l'os du bras en lui-même pour donner un rendu le plus naturel possible. Même chose au niveau de la tête : lorsque le modèle avait la tête inclinée, les cheveux se "détachaient" de la tête, ce qui était une conséquence de l'attache entre l'os du cou et l'os de la tête, que j'ai supprimée. J'ai également rencontré quelques problèmes du même type au niveau des jambes et des pieds lorsqu'ils étaient pliés, que j'ai résolu de la même manière que pour les bras.

Une fois le squelette créé et les mouvements du modèle rendus plus naturels, je pouvais passer à l'animation à proprement parlé. Par manque de temps, nous avons décidé de ne faire que deux animations principales : un cycle de marche, pour que l'on puisse se déplacer dans les niveaux sans avoir l'impression de contrôler un bâton, et une animation de saut. Dans un premier temps, il a fallu se placer dans une fenêtre "Action Editor" pour pouvoir enregistrer les mouvements qu'allait devoir effectuer le personnage. J'ai choisi de faire un cycle de marche sur 16 frames,

par conséquent il fallait donner 16 poses différentes au personnage et les enregistrer pour créer l'animation. Le plus difficile dans cet exercice a été de donner au personnage des positions naturelles, c'est-à-dire qu'il fallait penser non seulement à la position des bras et des jambes mais aussi au mouvement des hanches, des épaules et du tronc, au poids du corps qui devait passer d'une jambe à l'autre. J'ai décomposé le mouvement en 3 frames clés : la pose de départ du personnage, la pose de fin, et la pose intermédiaire. Entre le début et le milieu, puis entre le milieu et la fin, il a évidemment fallu que je détaille le mouvement, ce qui m'a forcé à réellement décomposer un mouvement que nous faisons tous tous les jours sans y prêter attention. Je me suis d'ailleurs appuyée sur des travaux d'animation papier pour tenter de mieux comprendre le déroulement d'un pas. Il a fallu beaucoup d'ajustements pour tenter d'obtenir un résultat fluide, mais finalement je suis assez satisfaite du travail fourni sur le cycle de marche. Le déroulement de l'animation de saut quant à lui est sensiblement le même que l'animation de marche : définir le nombre de frames sur lesquelles effectuer l'animation, réfléchir au positionnement du personnage à chaque instant, ajuster les mouvements pour obtenir un rendu fluide.

7 Site Internet

Lors de l'élaboration du cahier des charges, nous ne nous étions pas étendus sur ce que nous voulions réaliser concernant le site Internet, car c'était avant tout le jeu qu'il fallait développer. Pour autant, celui-ci a rapidement été pris en considération et sa construction a démarré dès la validation du cahier des charges, en même temps que le jeu lui-même. Cela nous a permis d'y réfléchir posément et la grande avancée réalisée par Gaël pour la première soutenance nous a permis de nous concentrer entièrement sur le jeu ensuite.

Gaël

Le site internet a représenté un gros défi, il devait être esthétique fonctionnel et aussi le moins chronophage possible. C'est donc pour cela que j'ai décidé d'utiliser un éditeur local afin de gagner du temps en mise en page mais pouvoir tout de même avoir accès à mon code source. J'ai donc d'abord appris à utiliser Open Element, l'éditeur gratuit que j'exploite pour créer notre site, et ensuite j'ai modélisé le site en implémentant les pages au fur et à mesure du projet et leur pertinence vis-à-vis de notre avancement.

Quant au design, le site internet avait deux points importants sur lesquels il fallait travailler. Tout d'abord l'esthétique. En effet, la population passe de plus en plus de temps sur internet et a donc l'œil de plus en plus affûté concernant l'esthétique des sites. Outre cela, Internet étant devenu le média le plus utilisé, il fallait que le site soit esthétique et agréable d'utilisation car il sera probablement la première image qu'aura l'utilisateur de notre jeu avant même de jouer. Ainsi, pour faire un site esthétique j'ai consulté les sites des grandes firmes internationales telles que *Tesla* et *Apple* mais aussi des jeux le plus joués tel que GTA V ou Fortnite, afin de dégager les différents points qui faisaient de ces sites des sites agréables d'utilisation. Une fois ce travail effectué, j'ai travaillé sur le design des pages déjà existantes du site.

Pour cela, j'ai d'abord épuré les pages du site une à une en les rendant le plus compréhensible possible, c'est-à-dire en permettant à chaque utilisateur, quel qu'il soit, de s'y retrouver sur la page et de savoir le plus rapidement possible où trouver ce qu'il cherche. Ce travail a donc consisté en rendre clair et lisible les différentes parties de chaque page. Une fois ce travail d'allègement effectué, il a fallu rendre esthétique le design de la page. Pour cela il fallait d'abord que les couleurs utilisées ne soient pas trop nombreuses, que le fond de chaque page ne fasse pas trop minimaliste et enfin qu'il n'y ait pas un surplus d'images ou de texte. J'ai

donc choisi en accord avec les autres membres du groupe de faire un site à dominante sombre. Suite à cela, nous avons choisi d'utiliser du bleu comme couleur pour les titres, et blanc cassé pour les textes. Après, la dernière étape a consisté en choisir les images de fond de chacune des pages. Pour cela, il a fallu donc choisir des images ou des vidéos en rapport avec les thèmes de chaque page. Ainsi, pour la page d'accueil on a choisi des braises qui volent vers le haut de la page, les braises évoquant les torches et donc l'aventure. De plus, la page d'accueil étant la première page que voit celui qui consulte notre site, l'invitation à l'aventure est donc évidemment une invitation implicite à jouer à notre jeu. Dans cette même logique d'adéquation du fond au thème de la page, chaque fond a été choisi avec soin.

La deuxième étape pour rendre le site agréable d'utilisation a été d'en faire un site complet, c'est-à-dire de permettre à n'importe quel utilisateur de notre site de trouver tous les renseignements qu'il souhaite obtenir sur notre site. Ainsi, j'ai ajouté à la page avancement du projet une barre représentant les pourcentages de l'avancement du projet tout au long de sa création. Elles sont à présent toutes à 100%. J'ai également ajouté la page "Espace des Gammarois", un espace commentaires où les joueurs peuvent trouver des réponses aux questions qui reviennent souvent et laisser leurs impressions sur le jeu, mais aussi une adresse e-mail à laquelle ils peuvent envoyer leurs questions.

8 Récits personnels de la réalisation

8.1 Romain

Ce projet m'a permis avant tout de voir et assimiler le processus de réalisation d'un projet en partant de zéro, et plus particulièrement d'un

jeu. Désormais, à chaque fois que je joue, je me demande comment j'aurais codé telle ou telle chose. Je suis souvent en mesure d'imaginer les bases, même si elles ne sont pas les meilleures, d'une méthode qui pourrait donner de bons résultats. J'apprécie les jeux beaucoup plus qu'avant, et mes connaissances en C# et sur le fonctionnement de Unity se sont améliorées.

Même si je pense avoir progressé dans certains domaines, je suis cependant plutôt déçu des erreurs que j'ai pu faire lors de la conception des scripts et lorsque j'ai cherché à résoudre certains problèmes, notamment au niveau du réseau : je me suis à de nombreuses reprises dirigé sur des voies trop compliquées, et ai dû, au final, réécrire et refaire beaucoup de choses. Néanmoins, je sais que si ce projet était à refaire, j'ai les capacités et le savoir permettant de diminuer la quantité de travail nécessaire pour parvenir à un résultat similaire, et mes longues heures de debug me seront au final utiles pour d'autres projets où des erreurs ou des problèmes similaires peuvent apparaître.

8.2 Louis

Ce projet m'a permis d'apprendre énormément sur la façon de travailler en équipe et la façon de communiquer avec les différents membres de mon groupe. Ce ne fut pas un travail facile. En effet, nous avons rencontré quelques soucis dans différents domaines. Cependant, nous avons toujours réussi à les surpasser ou à les contourner pour toujours avancer et ne pas rester bloqués sur certains éléments. Ce projet m'a également permis d'avoir une nouvelle approche du C# : le travail effectué sur les différents scripts présents dans le jeu a été extrêmement différent de ce que l'on a pu travailler et apprendre au cours de nos différents Travaux Pratiques. Ainsi, ce projet m'a apporté énormément de choses, que cela soit sur le plan humain, sur le plan personnel ou sur le plan technique.

Pour ce qui est de l'aspect humain, le travail en groupe a nécessité d'être à l'écoute, d'être souvent disponible et de savoir faire des concessions pour continuer d'avancer dans une bonne ambiance de travail. Il était primordial de ne pas perdre notre bonne entente durant tout le projet afin d'éviter des différends et de finalement perdre du temps dû à des problèmes de communication. Sur le plan personnel, ce projet m'a permis de voir un peu plus comment se déroule la création d'un projet. En effet, je n'avais jamais eu à réaliser un travail de cette ampleur en groupe. J'ai donc dû apprendre les différentes étapes de conception et de création. Cela m'a permis de comprendre en quoi consistait un cahier des charges, comment se déroule une soutenance de projet et enfin comment rendre un projet fini et le présenter devant un jury. Finalement, sur le plan technique, j'ai décidé d'essayer le maximum de choses, de ne pas réaliser qu'une seule tâche ou qu'un seul groupe de tâches très spécialisées. J'ai ainsi pu toucher à du Gameplay en élaborant plusieurs scripts, à du multijoueur, à la gestion d'un menu ou même à la création de niveau. Ce projet me permet aujourd'hui de me sentir plus à l'aise sur un grand nombre de domaines qui me permettront, si c'est à refaire, de réaliser ce projet en un temps réduit.

8.3 Laure

Travailler à la réalisation de GAMMA m'a permis d'acquérir beaucoup de connaissances dans les domaines dont je me suis occupée, c'est-à-dire le Level Design et l'édition des niveaux, la modélisation 3D, les animations, et les quelques scripts auxquels j'ai pu toucher. Ce sont des domaines auxquels je m'intéressais avant de venir à EPITA, et que j'ai trouvé intéressants de tester pour ce premier projet. Au final, cela m'a permis d'y voir plus clair sur ce qui me plaisait ou pas : j'ai découvert que la modélisation 3D et l'animation nécessitaient une réelle formation à part entière, et n'ayant eu aucune préparation avant de me lancer dans ce projet, je me suis très vite perdue dans ces deux notions qui ont été très

fastidieuses à mener à bien. Finalement, je suis déçue de ma performance concernant la modélisation surtout, mais je suis également agréablement surprise et satisfaite concernant la conception et la modélisation des niveaux 2 et 3 sur Unity, auxquelles j'ai pris plaisir.

En tant que chef de ce projet, je me suis également occupée de la logistique. J'ai réellement tenté de poser les bases d'un cadre sain au sein duquel travailler, ai fait mon possible pour organiser notre planning du mieux possible. Je voulais à tout prix éviter les fameux "rush projet" dont ont pu nous parler les Spé, qui auraient été à mon avis contre productifs et très compliqués à mener à bien. Au lieu de cela, nous avons réussi à nous réunir tous les lundi pour faire un point sur le travail qui avait été effectué et sur ce qu'il restait à faire. Je me suis également occupée de superviser l'organisation du cahier des charges, des deux rapports de soutenance, et du rapport de projet final. Je suis satisfaite du travail que nous avons effectué, et suis extrêmement soulagée d'avoir eu des camarades impliqués et motivés à finir GAMMA.

8.4 Gaël

Le projet GAMMA a représenté une vraie aventure, source d'apprentissage sur tous les plans mais aussi source de nombreux problèmes, techniques et créatifs. Ce fut aussi une expérience humaine : au départ nous ne nous connaissions pas tous personnellement, nous avons donc appris tout au cours de la création de GAMMA à se connaître.

La partie technique telle que je l'entends représente toute la partie du projet où il y a eu concrétisation physique des différentes parties du projet, par exemple la modélisation ou bien encore la création de scripts. J'ai travaillé sur trois parties techniques, d'abord la modélisation des niveaux 1 et 4, ensuite la création du site internet, et enfin l'enregistrement des mélodies pour le niveau musical. Tout d'abord, la partie modélisation a représenté pour moi deux défis, celui d'apprendre à utiliser Unity et

les Assets, ensuite celui de maîtriser la modélisation en 3D pour un jeu vidéo, ne l'ayant pratiqué auparavant que pour la modélisation d'objet dans le cadre de la matière Sciences de l'ingénieur au lycée. Le premier s'est avéré être assez long, en effet il a fallu tout d'abord que je m'entraîne à maîtriser Unity à l'aide de cours vidéo, et de tutoriels gratuits et accessibles sur Internet. Une fois ces cours et tutoriels regardés et appris, j'ai eu à les pratiquer en modélisant le premier niveau. C'est donc là que j'ai affronté le deuxième défi : celui de la modélisation en 3D, qui m'a imposé d'adapter mes plans dessinés en 2D aux problèmes qu'ils ont rencontré lors du passage à la troisième dimension. De plus, la remodelisation nécessaire suite à la première soutenance a été assez long et a nécessité beaucoup de travail personnel autant sur le temps demandé que sur la difficulté des problèmes rencontrés. Ensuite, l'élaboration du site Web a seulement nécessité la prise en main de l'éditeur, étant donné que j'ai déjà eu l'occasion de créer un site web. Il en a été de même pour la musique. Néanmoins, même si j'ai appris beaucoup grâce à cette partie technique qui m'a été très utile (en effet j'emploie maintenant les logiciels que j'ai appris à maîtriser fréquemment maintenant), elle a de loin été celle que j'ai le moins apprécié faire car elle a été la plus fastidieuse et la moins sollicitante intellectuellement.

Concernant la conception des niveaux, j'ai élaboré le niveau 1 avec Laure et le niveau 4 avec elle mais aussi avec Louis. Ce travail de conception des niveaux a été le plus intéressant, c'est dans ces moments de création que l'on pouvait le plus apporter notre vision du jeu et aussi le plus apprendre à nous connaître. De plus, il a été très instructif au sens où ce n'est pas souvent que l'on a à penser un jeu vidéo, on y a donc tout à apprendre. La partie création du site Web a quant à elle été beaucoup plus personnelle. En effet, même si le design a évolué aux demandes et en fonction de la vision du groupe, j'ai été le seul membre du groupe qui ai travaillé dessus techniquement. La création du site fut instructive car elle m'a mené à comprendre pourquoi et comment étaient faits les

sites des grandes firmes des nouvelles technologies. J'ai ensuite essayé de respecter les codes afin de faire un site agréable d'utilisation. Enfin, il y a eu le travail de création de la musique qui a représenté le défi qui m'effrayait le plus car j'ai commencé les cours de piano que cette année, et que même si j'en joue depuis trois ans, j'avais peur de ne pas pouvoir produire quelque chose. Et finalement à force de travail, d'écoute et de jeu, j'ai produit plus de morceaux que nécessaire.

Enfin, le plan humain du projet est non négligeable. En effet, nous avons tous une manière de fonctionner différente, et avec le stress et la pression on a tendance à avoir les nerfs à vif, ce qui aurait pu susciter des tensions. Finalement, malgré quelques tensions dues à un souci de communication dont j'étais responsable, le projet s'est effectué dans une très bonne ambiance et ce jusqu'à la fin. On a appris à se connaître, à s'adapter aux comportements et aux aptitudes de chacun, ce qui finalement nous a permis de mener à bien notre projet.

Finalement, ce projet m'a appris beaucoup de choses sur le plan technique, créatif et humain. Et même si c'est par hasard que je me suis retrouvé à EPITA, j'y ai pu à mon grand étonnement créer des choses dans des domaines que je n'aurais jamais pensé pouvoir faire grâce au projet. En effet, c'est grâce au projet que j'ai composé mes premières mélodies, et bien qu'elles soient simples et qu'elles n'aient rien de grandiose, cela reste une expérience folle pour moi, de même pour le site Web. Alors même si je n'ai toujours pas envie de faire de l'informatique mon métier à la fin de ce projet, je reste très heureux d'avoir pu le faire. Même s'il y a eu des nombreuses difficultés, cela a été enrichissant et je ne regrette pas d'avoir eu le groupe que j'ai eu, ni d'avoir le projet que l'on a fait.

Conclusion

Pour conclure et partager quelques derniers mots, ressentis ou avis, ce projet aura été une expérience peu banale et très enrichissante, c'était une première pour nous tous du moins à cette échelle. Il nous aura beaucoup apporté : des connaissances, du savoir-faire, un esprit d'équipe plus affûté, de l'ambition, une certaine confiance en nous et en ce qu'on a produit, mais aussi et il ne faut pas l'oublier, du stress, des soucis... qui nous ont néanmoins appris à nous débrouiller seuls, à trouver des alternatives, à chercher par nous-mêmes et à nous montrer curieux. Les points positifs concernant ce projet sont très nombreux, et les petits désagréments que nous avons rencontrés au cours de l'avancée de ce projet ne nous seront au final que profitables pour compléter notre expérience. Finalement, le jeu que nous présentons, GAMMA, est conforme à nos attentes et au cahier des charges que nous avons soumis au début de ce deuxième semestre : un platformer 3D, multijoueur, musical, original et presque entièrement fait par nos soins : nous avons conçu et nous sommes approprié chaque script qui permet aujourd'hui de jouer à GAMMA, avons conçu puis modélisé les quatre niveaux de plateformes et les niveaux musicaux ainsi que nos personnages et collectables, nous avons même créé de toute pièce la musique et un site internet sur lequel télécharger le jeu et trouver toutes les informations nécessaires sur celui-ci. En partant de zéro, c'est-à-dire sans autre connaissance du monde du jeu vidéo que notre expérience de joueur, nous sommes arrivés à modeler un jeu conforme à nos attentes, assez pour que nous en ressentions de la fierté.

Cette aventure nous a également rapprochés, et permis de mieux nous connaître nous-même ainsi que les autres membres du groupe. La cohésion de groupe est restée solide malgré tous les obstacles que l'on a pu rencontrer et malgré les quelques différends qui nous ont opposés. Chacun a mené à bien ce qu'il devait faire, a pris ses responsabilités et a porté le projet là où il en est aujourd'hui. Nous croyons pouvoir affirmer

que créer un jeu vidéo est moins difficile qu'on ne le pensait au début pourvu que l'on ait la motivation et l'envie de mener le projet à bien.

Annexes & Sources

8.5 Sources

Logiciels :

- Open Element : Pour la création du site Web
- Anvil Studio : Pour l'enregistrement des mélodies
- Unity : Pour la modélisation des niveaux
- Audacity : Pour les retouches des mélodies
- Blender : Pour la modélisation et l'animation des personnages
- Paint Tool SAI & Fire Alpaca : Pour l'élaboration du *chara design* des personnages
- Rider : Pour l'élaboration de tous les scripts C#

Sites Web :

- Open Classrooms : Pour les cours vidéo sur Unity
- Wikipedia : Pour divers renseignements
- Unity : Pour la documentation et les tutoriaux sur Unity
- Pexels Video : Pour les vidéos libres de droit de fond des pages du site Web
- Pexels Image : Pour les images libres de droit de fond des pages du site Web
- Videvo : Pour les vidéos libres de droit de fond des pages du site Web
- Overleaf : Pour la rédaction des différents documents textes
- Deezer : Pour l'écoute des musiques dans le cadre de la recherche d'idées pour les mélodies

- Tesla : Pour la recherche d'idées pour le site Web
- Apple : Pour la recherche d'idées pour le site Web
- GTA 5 : Pour la recherche d'idées pour le site Web
- Fortnite : Pour la recherche d'idées pour le site Web
- Youtube : Pour les nombreux tutoriels concernant tous les aspects du jeu (notamment les tutoriels de la chaîne "Brackeys", "Sebastian Lague")

Assets :

- Cartoon Temple Building Kit Lite (<https://assetstore.unity.com/packages/3d/environments/dungeons/cartoon-temple-building-kit-110397>) : Pour l'aspect graphique de chaque niveau de GAMMA
- Photon Unity Networking : Pour gérer toute la partie réseau

8.6 Captures d'écran

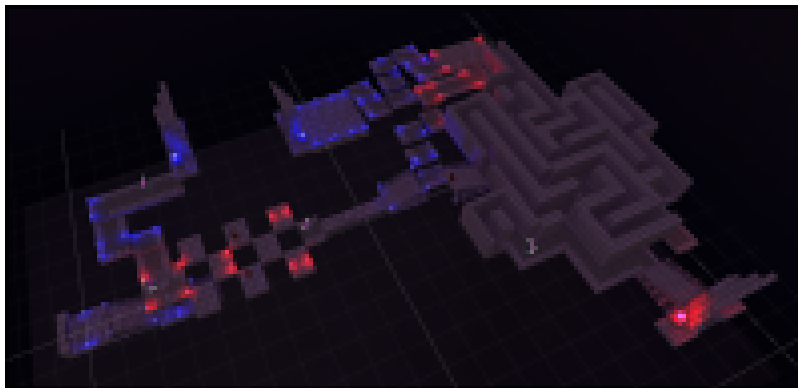


FIGURE 1 – Aperçu de la modélisation du niveau 1

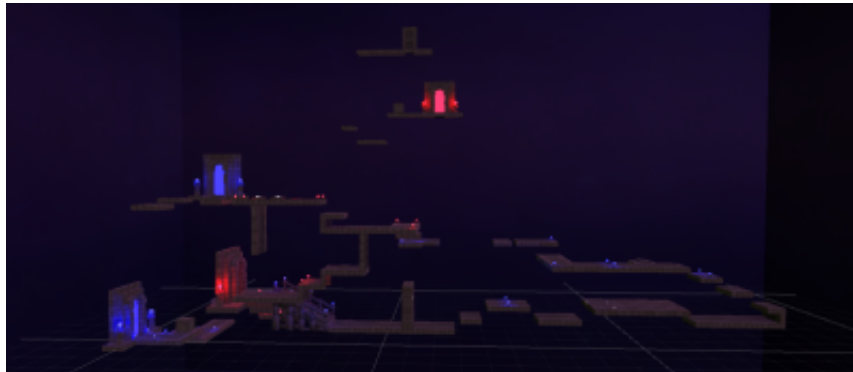


FIGURE 2 – Aperçu de la modélisation du niveau 2

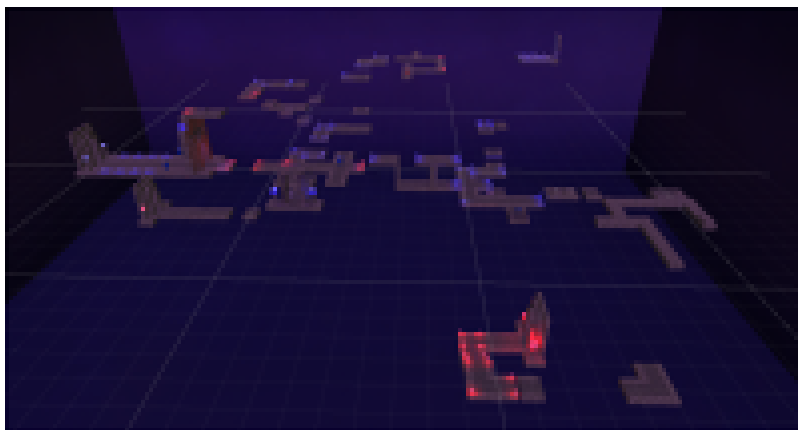


FIGURE 3 – Aperçu de la modélisation du niveau 3



FIGURE 4 – *Chara Design* original des personnages



FIGURE 5 – Étapes de modélisation et d'animation