

Compte-rendu Soutenance 1

La Confrérie

Pierre Kelbert, Morgan Vaterkowski, Léo Hantsch, Sébastien Hoehn

14 mars 2018



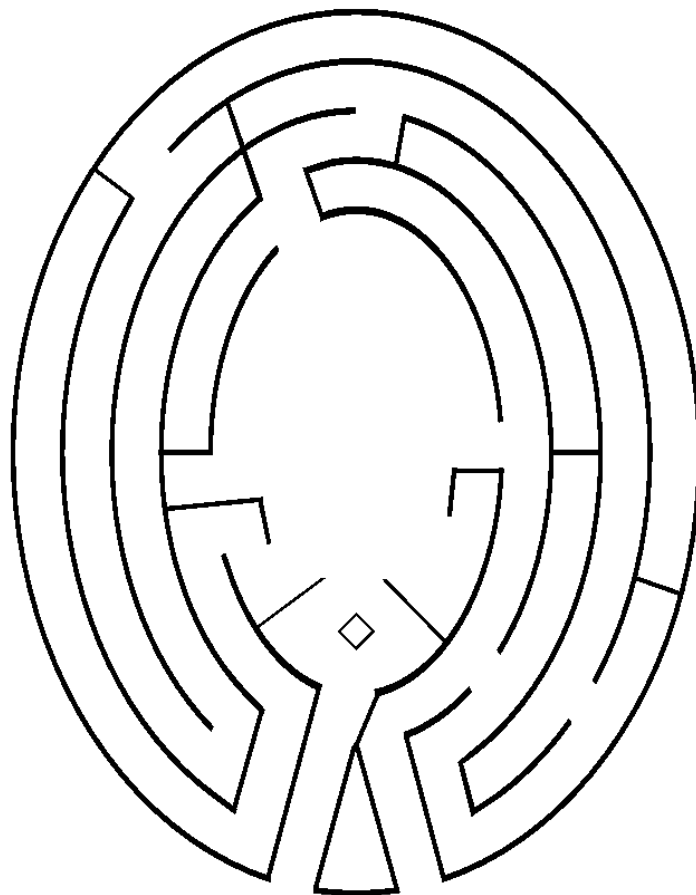
Table des matières

1	Introduction	3
2	Planning	4
3	Détail du travail réalisé	5
3.1	Pierre	5
3.2	Morgan	8
3.3	Léo	10
3.4	Sébastien	14
4	Travail pour la prochaine séance	17
5	Conclusion	18

1 Introduction

Depuis la rédaction du cahier des charges, la réalisation du projet s'est mise en marche et le travail a été ponctué de réunions de groupe pour vérifier l'avancement. Les membres du groupe ont découvert ou redécouvert le travail de groupe et chacun a rencontré ses propres difficultés qu'il a appris à surmonter pour faire fonctionner sa partie. De plus, les contrôles ont constitué une difficulté supplémentaire à prendre en compte. Ce compte-rendu de soutenance décrit ainsi le travail réalisé, compare celui-ci avec le planning et présente le travail pour la prochaine soutenance. Pour cela, le cahier des charges sera repris en premier lieu, puis chaque membre du groupe décrira précisément son travail. Enfin, le travail restant sera présenté avant la conclusion.

FIGURE 1 – Plan du niveau *WestWorld* sur *SketchBook*



2 Planning

TABLE 1 – Reprise du cahier des charges

Tâches	Planning	Réalisé
Niveaux solo	60%	63%
Design 2D	80%	68%
Système de portes	50%	60%
Commandes	95%	95%
IA des monstres	20%	20%
LaTeX	100%	100%
Menu	20%	35%
Réseau	20%	30%
Tutoriel	0%	60%

D'après le tableau 1, les délais sont respectés pour l'intelligence artificielle des monstres. En effet, le path finding a été codé et fonctionne mais doit encore être ajouté aux niveaux. De plus, les triggers ne sont pas encore créés. Le fichier LaTeX est également prêt pour la soutenance. Les commandes aussi respectent les délais puisque le joueur peut déplacer son personnage et changer son angle de vue.

On peut s'apercevoir que nous avons de l'avance dans plusieurs tâches : les niveaux solo, le système de portes, le menu et le réseau. En effet, nous avons réalisé 10 niveaux sur 16 ce qui constitue 63% des niveaux. De plus, en ce qui concerne les portes, celles-ci ont été modélisées par Morgan et leurs animations ont été créées. Il en est de même pour les boutons qui leurs sont associés. Ainsi, seul le code déclenchant les animations lors de l'appui sur les boutons est manquant. Enfin, le menu est constitué de boutons qui permettent de choisir un niveau ou de quitter le jeu et le réseau permet à plusieurs joueurs d'être dans un même niveau.

Par ailleurs, Morgan a commencé le tutoriel et est donc arrivé à 60% de celui-ci en modélisant 3 niveaux sur 4.

Cependant, le design 2D est légèrement en retard par rapport au planning. Nous n'avons en effet que les plans des 16 niveaux modélisés en 3D, soit 68%.

3 Détail du travail réalisé

3.1 Pierre

Voilà que nous en sommes à notre 2^{ème} mois depuis le début du projet en janvier. Après tout ce temps consacré essentiellement à la découverte des différents logiciels qui vont nous accompagner dans la création, nous arrivons aujourd'hui au premier « CheckPoint ».

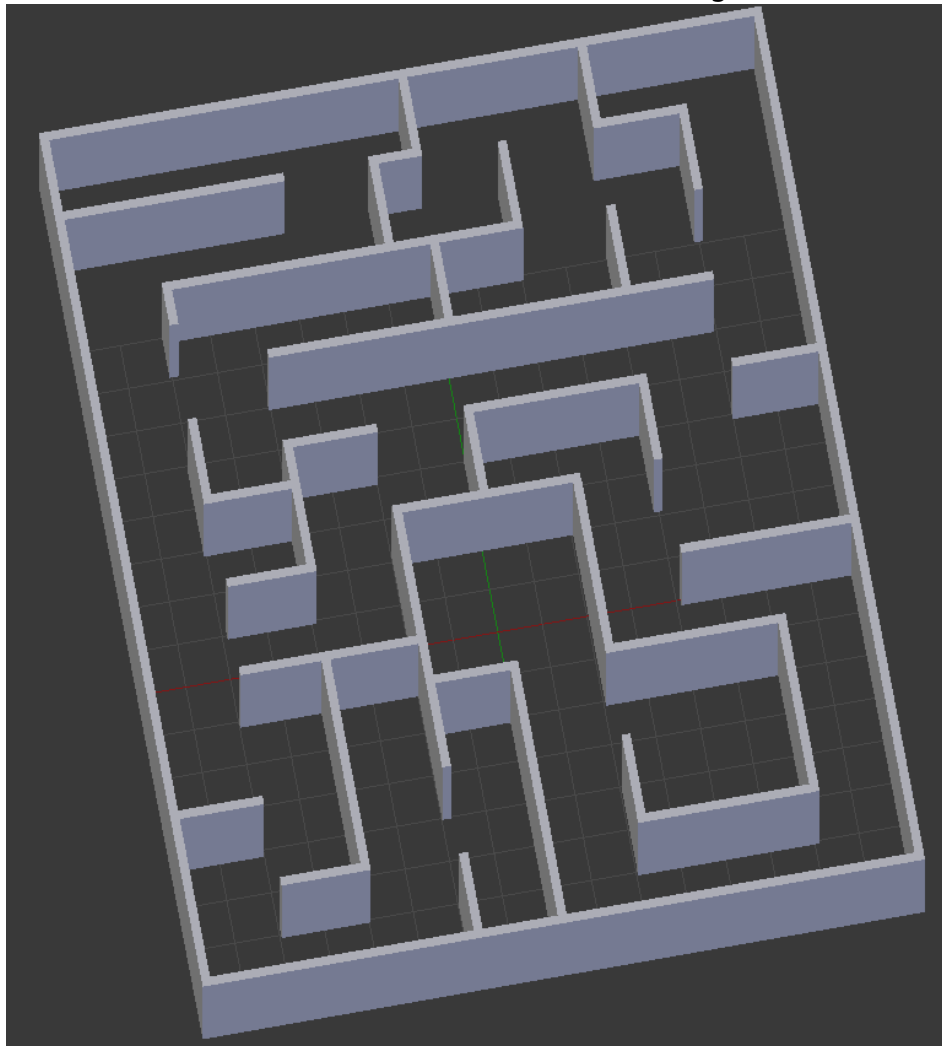
Pour cette première soutenance, le premier objectif qui m'était fixé était un objectif commun à tous les membres de la Confrérie : la réalisation des premiers niveaux. L'idée derrière la création des premiers niveaux est le fait qu'ils sont essentiels au bon déroulement des prochaines étapes dans la conception du jeu...

Face à ce premier objectif, je dois reconnaître que je ne savais pas vraiment par où commencer; voilà pourquoi dans un premier temps, je me suis reposé sur la présentation *Bien Démarrer Son Projet* (BDSP) pour trouver quelques pistes. C'est à travers leurs explications que j'ai fait mes premiers pas sur *Blender* et *Unity*. C'est une fois ce petit bagage de connaissances acquis que j'ai pu commencer à mettre en œuvre mes « talents artistiques » pour remplir mon premier objectif (enfin)!

Bien qu'il soit habituel de réaliser les cartes sur *Unity*, je me suis d'avantage servi de *Blender* pour concevoir les murs des niveaux en raison de l'aspect pratique du logiciel. Parlons-en d'ailleurs, *Blender* est pour moi un logiciel dont la prise en mains n'est pas évidente en raison de ses commandes de navigation (avancer, reculer, zoomer, ...) qui ne sont pas naturelles. Cependant et malgré ce point négatif, force est de constater qu'il s'agit d'un outil riche et complet pour la conception d'objets 3D.

Revenons-en à mes cartes. Tous mes niveaux sont donc des objets constitués de murs conçus sur *Blender* (voir figure 2). Les autres éléments de la map faisant appel à un script tel que les sols, portes, boutons, ... seront rajoutés directement dans les différentes scènes du jeu à partir de *Unity*.

Une fois les premières maps terminées après plusieurs laborieuses heures, est venu le moment pour moi de m'attacher à mon second objectif LE RÉ-SEAU... ! Un jeu étant toujours plus amusant s'il offre la possibilité d'y faire participer d'autres personnes (et je ne parle pas que de la famille!). J'ai accordé une grande importance à ce point.

FIGURE 2 – Niveau de Pierre sur *Blender 3D*

Pour la mise en réseau, j'ai fait le choix d'utiliser le service multijoueur *Photon Engine*. Ce dernier avait été abordé par les « anciens » lors de la même présentation précédemment citée. De plus, il s'agit d'un système qui reste souvent cité dans ce contexte de projet en raison de sa facilité de mise en place.

Le cahier des charges imposait que le travail de mise en réseau soit débuté et c'est donc à ce titre que j'ai créé une nouvelle application sur l'interface web de Photon. Malheureusement, avancer d'avantage dans cette partie nécessite que les commandes du personnage soient achevées, ce qui n'est pas encore tout à fait le cas. Je n'ai donc pas poussé plus loin cette

partie.

A titre informatif pour la suite des évènements, j'estime être à 30% du travail total nécessaire à l'aboutissement d'un système multijoueur. Il s'agit là d'une valeur vue sous un angle pessimiste, mais qui à mon goût reste préférable pour éviter toute mauvaise surprise... En raison d'un nombre faible de tests, je ne peux cependant pas exprimer mon avis sur Photon, si ce n'est que la création d'une tâche multijoueur reste un jeu d'enfant. A l'heure actuelle, je remplis donc mes objectifs liés au cahier des charges en ce qui concerne la première soutenance.

Mon travail en revanche ne s'est pas arrêté là puisque je me suis également occupé d'une partie non présente dans le cahier des charges : la charte graphique du jeu (en ce qui concerne son aspect de communication). Pour cette partie, contrairement à la conception 3D, j'avais déjà quelques bagages concernant le design. J'ai donc créé un logo pour représenter l'équipe et le projet dans son ensemble.

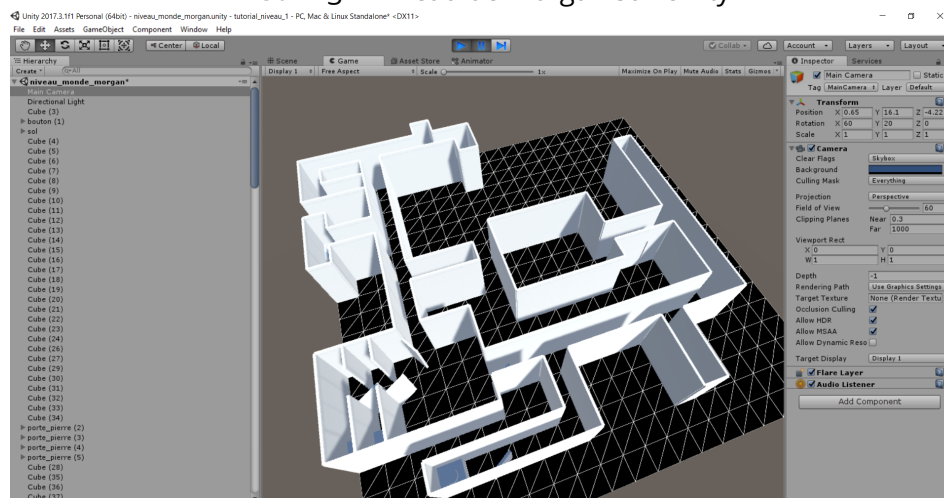
Pour la création du logo, je suis tout d'abord parti de quelques croquis (il n'y a honnêtement rien de mieux) pour finalement basculer sur du dessin vectoriel grâce au logiciel open source *Inkscape*. L'intérêt pour nous d'utiliser ce format d'image (je ne vais pas détailler ce qu'est le vectoriel, *Wikipédia* le fait bien mieux) est tout simplement le rapport entre temps et qualité. En passant par cette solution nous obtenons un rendu qui reste sobre certes, mais propre, le tout en peu de temps. Il nous a fallu un peu moins d'une heure pour en arriver au résultat que vous pouvez observer en première page de ce document (voir aussi figure 11 page 20). Dès lors, la rapidité du concept est évidente...

Face à tout le travail que j'ai pu accomplir jusqu'à présent (et il m'en reste un paquet), je vais conclure que le résultat quant à l'avancement respecte les contraintes du cahier des charges. Le projet de mon côté suit donc le déroulement tel qu'il a été planifié et validé dans nos dernières versions du cahier des charges. Je pourrais vous détailler quels sont mes objectifs pour la prochaine revue de projet (je m'en réjouis déjà), cependant je pense qu'il s'agit d'une tâche tout bonnement inutile puisque le cahier des charges ainsi que l'avant-dernière partie de ce compte-rendu (tableau récapitulatif) vous donneront déjà ces informations. La seule chose que je puisse dire c'est que pour le moment il n'y a pas d'accroc sur nos prévisions et nous mettons tout en œuvre (moi compris) pour que cela perdure!

3.2 Morgan

Personnellement, je n'ai pas utilisé *Blender* puisque mes niveaux ont été directement créés sur *Unity* (voir figure 3). J'ai ainsi commencé par la création des portes et des boutons. Ceux-ci sont tout d'abord dessinés et leurs animations créées, les rendant ainsi prêts à être utilisés dans le programme C#. Je n'étais pas peu fier de moi lorsque j'ai réussi à créer ma première animation de porte (l'ouverture classique).

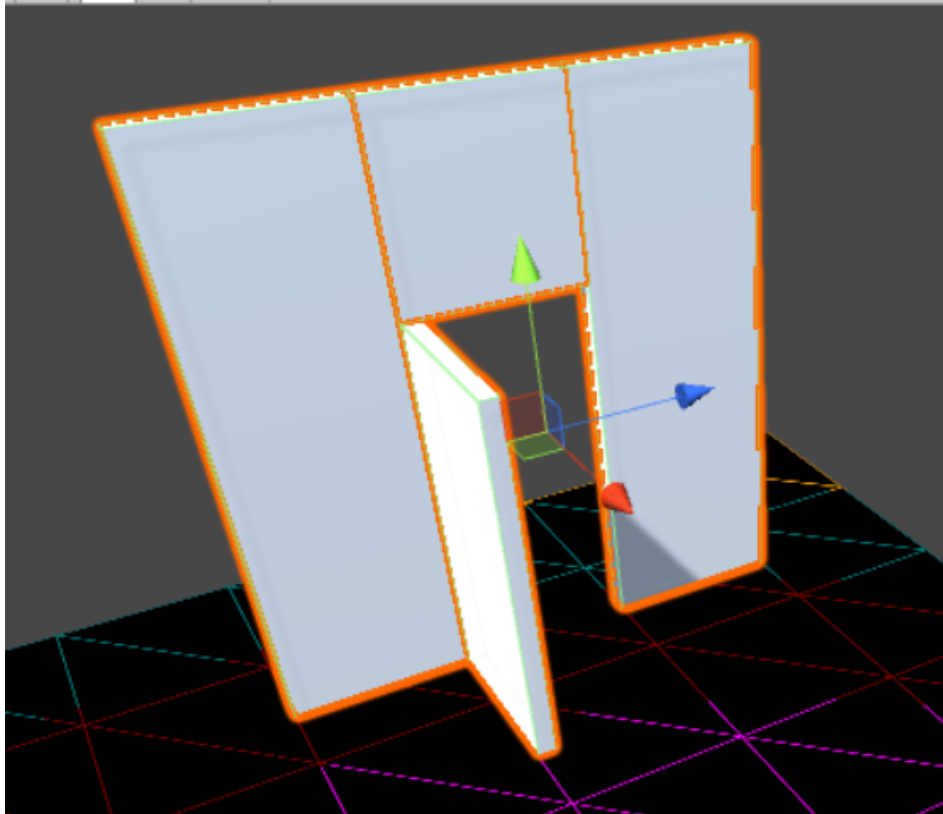
FIGURE 3 – Niveau de Morgan sur Unity



Cependant, j'ai dû par la suite recréer cette animation sur un autre objet *Unity*, car la Confrérie avait posé des normes pour la largeur des murs. Les miens ne les respectant pas, et de loin, les animations n'ont pas pu être adaptées. J'ai donc créé des portes plus fines qui s'ouvrent en couissant, ainsi que d'autres qui s'ouvrent en effectuant une rotation (voir figure 4), créant alors un mur sur le reste du couloir.

J'ai donc réalisé les premières ébauches de mes niveaux sur *Paint*, afin d'avoir un guide en passant sur *Unity*. J'ai ensuite créé une scène « porte » qui contient mes trois types de portes différentes. Cela permet une création facile des niveaux : j'importe tout d'abord ma scène « porte », je copie ensuite les éléments utiles sur ma nouvelle scène, puis j'écarte la scène porte de ma nouvelle scène. Pour modéliser les murs, je crée une colonne en 3D et je la duplique autant de fois que nécessaire avant de positionner tous les éléments aux places correspondantes. J'ai effectué ces actions pour chacun de mes niveaux (sauf certains du tutoriel).

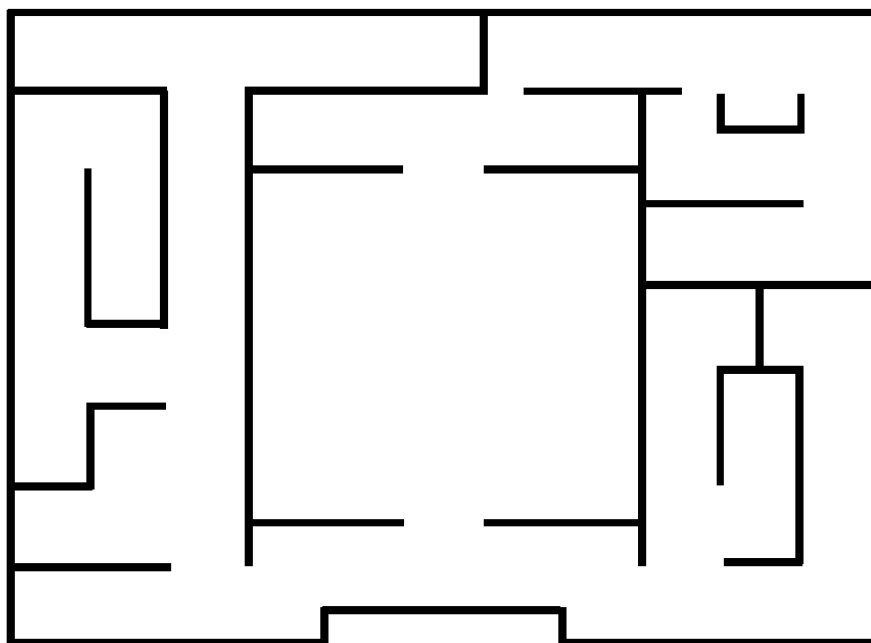
FIGURE 4 – Porte avec rotation



Les plans de mes niveaux (et de ceux des autres membres du groupe) avaient été préalablement dessinés sur *SketchBook* afin d'obtenir des plans précis et aisément transposables en 3D (voir figure 5).

Cependant, le script C# me donne toujours du fil à retordre, mon niveau étant passable sur la programmation orienté objet. Mais m'a plus grosse incompréhension reste sur les commande à effectuer dans la console pour mettre en commun mon travail grâce au logiciel *Git*. Même si j'ai compris le fonctionnement de celui-ci, à savoir récupérer les fichiers en ligne avant de les modifier afin de créer différentes versions d'un même fichier, certaines actions me restent indéterminées.

Globalement, pour résoudre les problèmes que j'ai rencontrés sur la création de mes niveaux et de mes animations, seuls la patience, le reste du groupe la Confrérie et les tutoriels *Unity* disponibles sur le site officiel ont pu m'aider.

FIGURE 5 – Plan d'un niveau sur *SketchBook*

3.3 Léo

Lors de la répartition des tâches, j'ai choisi de coder les différentes commandes du joueur car il s'agit d'un aspect de la programmation d'un jeu vidéo qui m'intéressait beaucoup. En plus de cela, j'ai également choisi de m'occuper de la création du menu principal du jeu sur *Unity* du fait de ma précédente expérience dans le domaine (je me suis effectivement occupé de la création du menu lors de mon projet ISN de Terminale). Enfin j'ai, comme tous les autres membres du groupe, été assigné à la tâche de création de niveaux.

Et c'est d'ailleurs sur cette tâche que j'ai commencé à concentrer mes efforts. Cette partie du projet m'a beaucoup plu car j'ai dû faire appel à des compétences artistiques, c'est-à-dire des compétences qui diffèrent des compétences scientifiques que nous utilisons quotidiennement dans le cadre de nos études dans l'ingénierie. Après plusieurs essais, une idée de concept me vint à l'esprit : celui de créer un raccourci qui permettrait au joueur de finir beaucoup plus rapidement le niveau. Bien sûr, il est évident que la porte pour débloquer ce raccourci serait beaucoup plus difficile à déverrouiller que les autres portes qui suivent le chemin d'origine.

J'ai pour le moment réalisé trois niveaux dont deux qui ont déjà été modélisés en 3D. On peut noter que mon premier niveau ne comporte pas de raccourci car il s'agit du premier niveau (qui est déjà très facile). Pour mon deuxième niveau, j'ai eu l'idée de faire passer le raccourci derrière le début du niveau, ce qui m'a permis d'ouvrir de nouvelles possibilités, comme par exemple faire varier la taille du raccourci ou encore la forme générale du niveau. De plus, une autre liberté prise dans mon troisième niveau (qui n'est pas encore modélisé en 3D) est la forme circulaire de mes murs, ce qui apporte un tout nouveau style de niveau, se rapprochant plus des labyrinthes grandeurs nature (exemple : le labyrinthe végétal « labyrinthus » à Ribeauvillé). La dernière nouveauté de ce niveau est que je choisis de placer le monstre au centre du niveau ; et selon les boutons actionnés par le joueur, le monstre parvient plus ou moins rapidement au joueur via différentes galeries. Le niveau en lui-même constitue un réseau de galeries (c'est d'ailleurs pour cela que j'ai décidé de nommer ce niveau « Gallery »).

Après avoir modélisé les niveaux sur papier, ce sont Morgan et Pierre qui se sont chargés de la modélisation 2D. Après cela, en me basant sur leurs modélisations 2D, il me fallut modéliser les niveaux en 3D à l'aide du logiciel *Blender*, logiciel que je ne connaissais pas avant cela. En effet, j'avais déjà modélisé le complexe sportif de mon collège en 3D via *SketchUp* en cours de technologie de troisième. Et comparé à *SketchUp*, je dois admettre que j'ai eu plus de difficultés à m'approprier *Blender*. La première difficulté fût de maîtriser la caméra et les différents zooms mis à disposition par *Blender*. En effet, j'ai rapidement compris à mes dépens qu'une mauvaise utilisation de la caméra pouvait rapidement conduire à de sévères maux de tête.

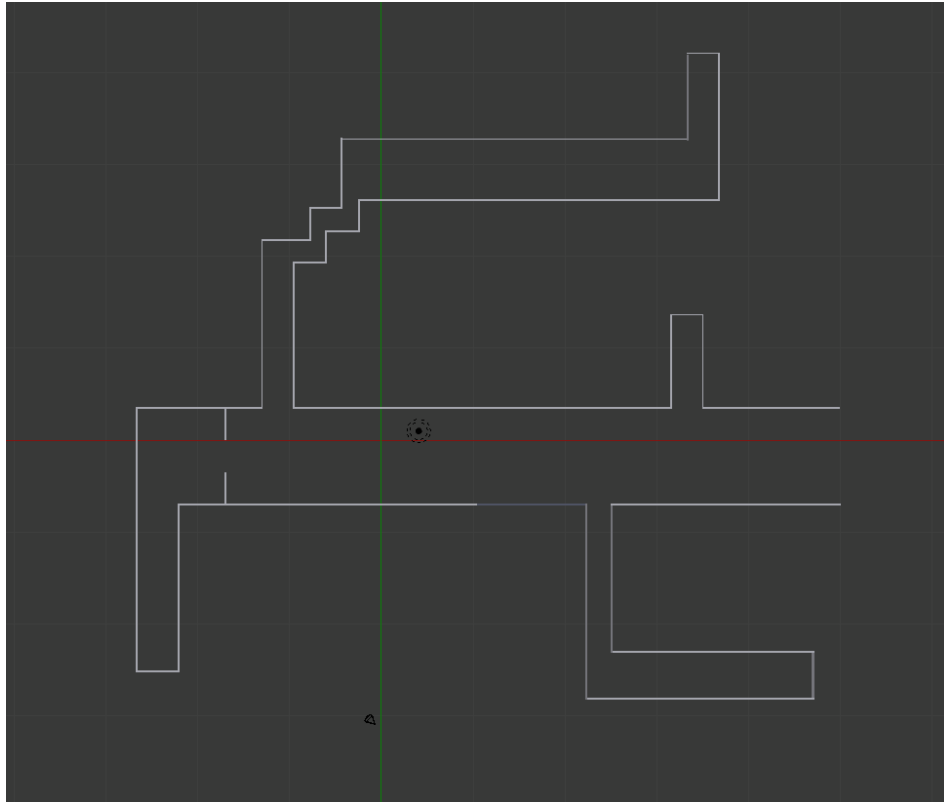
La finition des murs 3D constituait la seconde difficulté majeure dans la modélisation des niveaux en 3D. En effet, après avoir créé deux murs, il me fallut systématiquement zoomer au maximum sur les bords de deux murs perpendiculaires pour m'assurer qu'ils ne débordent pas, c'est-à-dire qu'une face ne dépasse pas par rapport à une autre.

En plus de cela, nous avons dû nous mettre d'accord avec les autres membres du groupes sur l'épaisseur et la hauteur des murs, de manière à fixer une norme et ainsi ne pas rencontrer de problème lors de la mise en commun des différents niveaux et de l'animation des portes de Morgan (qui doit respecter des dimensions très strictes). Ainsi, nous avons fixé l'épaisseur des murs à 0.1 m et la hauteur à 2 m.

De manière générale, pour modéliser un niveau, je commençais par modéliser un mur qui suivait un axe horizontal (X), un autre qui suivait l'axe

vertical (Y). Ensuite, je dupliquais et modifiais la taille de ces murs en fonction des besoins. Pour agrandir ou raccourcir un mur, je passais en « Edit Mode », sélectionnais « Verticle Face » et enfin agrandissais ou rétrécissais la face du mur selon les besoins (voir figure 6).

FIGURE 6 – Niveau de Léo sur *Blender 3D*



Parallèlement à la conception des niveaux, il me fallut coder les commandes du joueur, à savoir avancer tout droit, à gauche, à droite, reculer et bouger la caméra à l'aide de la souris. Ce qui était étrange dans la réalisation de ce code était que certaines manières de coder le déplacement fonctionnaient alors que d'autres non. Je m'explique : après avoir regardé et compris la méthode proposée dans la vidéo *Bien Démarrer Son Projet*, j'ai cherché des alternatives sur Internet. Et après en avoir trouvée une qui fonctionnait, je m'en suis inspiré pour créer mon code des commandes. Mais rien à faire, le code ne fonctionne pas, et le personnage sur *Unity* ne bouge pas, et cela même en reprenant de toutes les façons possibles le code. C'est pour cette raison que j'ai décidé d'abandonner cette version du code pour revenir à celle proposée dans la vidéo, beaucoup plus simple à reproduire.

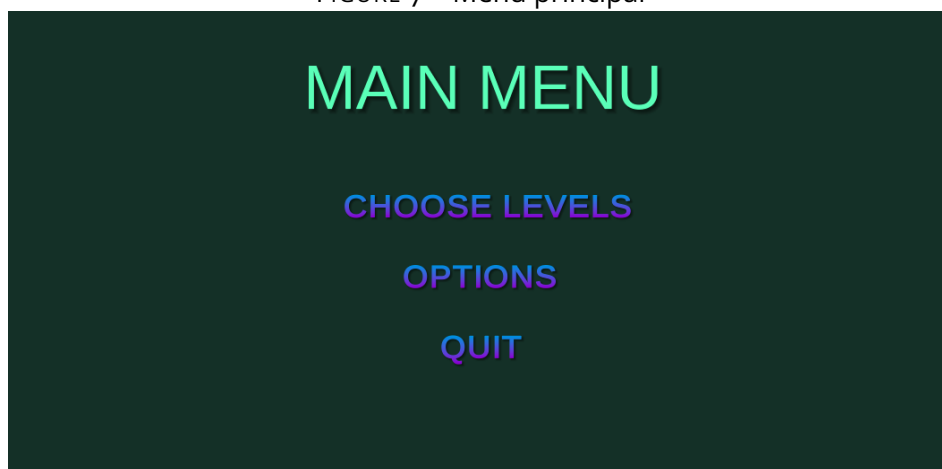
Mais après avoir codé le mouvement du personnage, il fallut coder le déplacement de la caméra via le mouvement de la souris. En soit, la méthode se rapproche du déplacement du personnage, à ceci près que la sensibilité de la souris et la notion de rotation sont les deux notions qu'il ne fallait pas oublier d'intégrer au code.

En ce qui concerne la vue à la première personne, j'ai d'abord dû rattacher la « Main Camera » de ma scène sur *Unity* à mon joueur, pour ensuite l'intégrer aux yeux de mon personnage.

Une dernière amélioration à faire en dehors de l'interaction avec les boutons serait de rattacher le changement de vue de la caméra avec le personnage car, pour le moment, lorsque que le joueur tourne sa caméra à droite, il doit également appuyer sur la flèche droite pour pouvoir avancer « tout droit » au lieu d'appuyer sur la flèche du haut comme il serait amené à le faire sur bon nombre de jeux PC.

Ma dernière mission fût de créer le menu principal du jeu sur *Unity* (voir figures 7). Comme précisé précédemment, je n'ai pas rencontré de difficulté particulière pour comprendre le fonctionnement d'un menu étant donné que j'en avais déjà implémenté un pendant mon projet ISN de Terminale. Le grand changement résidait dans la création du menu sur *Unity* en C# (mon précédent menu était codé en HTML5 couplé à du CSS3). En effet, la manière de créer un fond et des titres changeait par rapport au HTML5/CSS3 puisque la plupart des manipulations pour créer un menu (telle que la modélisation de fonds colorés, de titres et de boutons) ne requièrent pas de code sur *Unity*.

FIGURE 7 – Menu principal



Néanmoins, j'ai retrouvé la même difficulté pour relier deux pages du menu. En effet, après avoir codé le changement de page rattaché à un bouton, il ne fallait surtout pas se tromper au moment de rattacher le code sur le bouton depuis *Unity*. Je dois cependant admettre que la création de titres, boutons et fonds colorés est plus facile et riche sur *Unity* que via du CSS3, notamment grâce à l'Asset « TextMeshPro » qui apporte une grande diversité en ce qui concerne la coloration et la taille de la police.

3.4 Sébastien

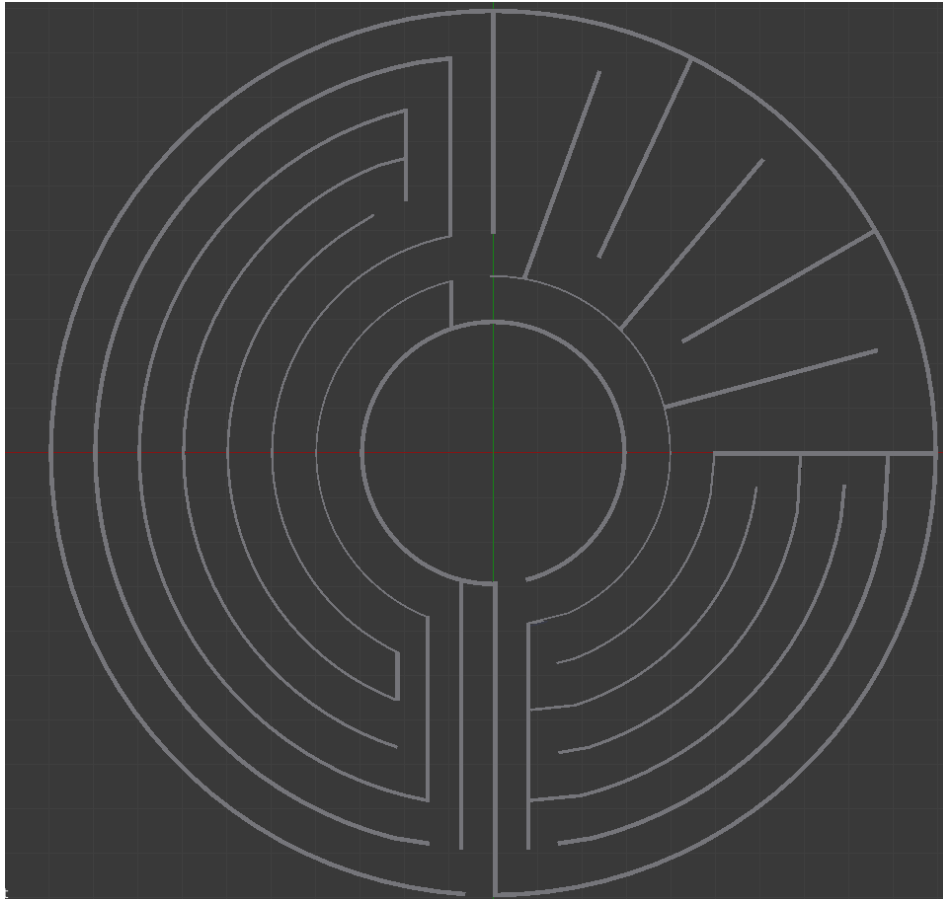
A la création du groupe, j'ai été choisi pour être le chef de projet. Je n'avais que très peu d'expérience dans ce domaine et j'ai découvert en quoi ce poste consistait pour un projet tel que celui-ci. J'ai ainsi dû prendre des décisions, ce que j'ai du mal à faire en temps normal. De plus, j'ai choisi d'effectuer des réunions de l'équipe pour contrôler l'avancement des différentes tâches et donner des objectifs à cours termes plus évidents à visualiser et à ne pas laisser traîner en longueur. Ces réunions ont débuté pour la rédaction du cahier des charges et se sont poursuivies par la suite avec une fréquence moins haute. Le déroulement type de ces réunions était le suivant : tout d'abord, chaque membre montrait son travail depuis la dernière réunion. Les autres membres donnaient alors leurs avis ainsi que des conseils pour améliorer cette réalisation. En deuxième partie, les nouveaux objectifs étaient définis et les membres échangeaient leurs idées pour que chacun ait une première idée de comment réaliser sa tâche. Enfin, un compte-rendu de la réunion était réalisé pour ne pas oublier d'idée et que les objectifs soient inscrits dans un document auquel chacun avait accès.

En ce qui concerne la réalisation des tâches qui m'étaient incombées, à savoir le début de l'intelligence artificielle des monstres et la réalisation de niveaux, j'ai commencé par une phase de recherches documentaires sur internet avant de passer à la réalisation.

Pour la réalisation des niveaux, j'ai étudié des labyrinthes existant déjà afin de m'inspirer de ceux-ci. Ainsi, l'un des labyrinthes est circulaire comme celui du film *Le Labyrinthe*, alors qu'un autre labyrinthe provient de la série *Westworld* (voir figure 1). J'ai alors dessiné mes niveaux sur papier avant de les envoyer à Morgan pour le design 2D. A partir des plans réalisés par ce dernier, j'ai pu modéliser mes niveaux en 3D à l'aide du logiciel *Blender 3D* (voir figure 8). J'avais déjà expérimenté quelques fonctionnalités de ce logiciel il y a plusieurs années mais j'ai eu besoin de reprendre les bases de son utilisation. La prise en main n'était pas intuitive mais après quelques heures

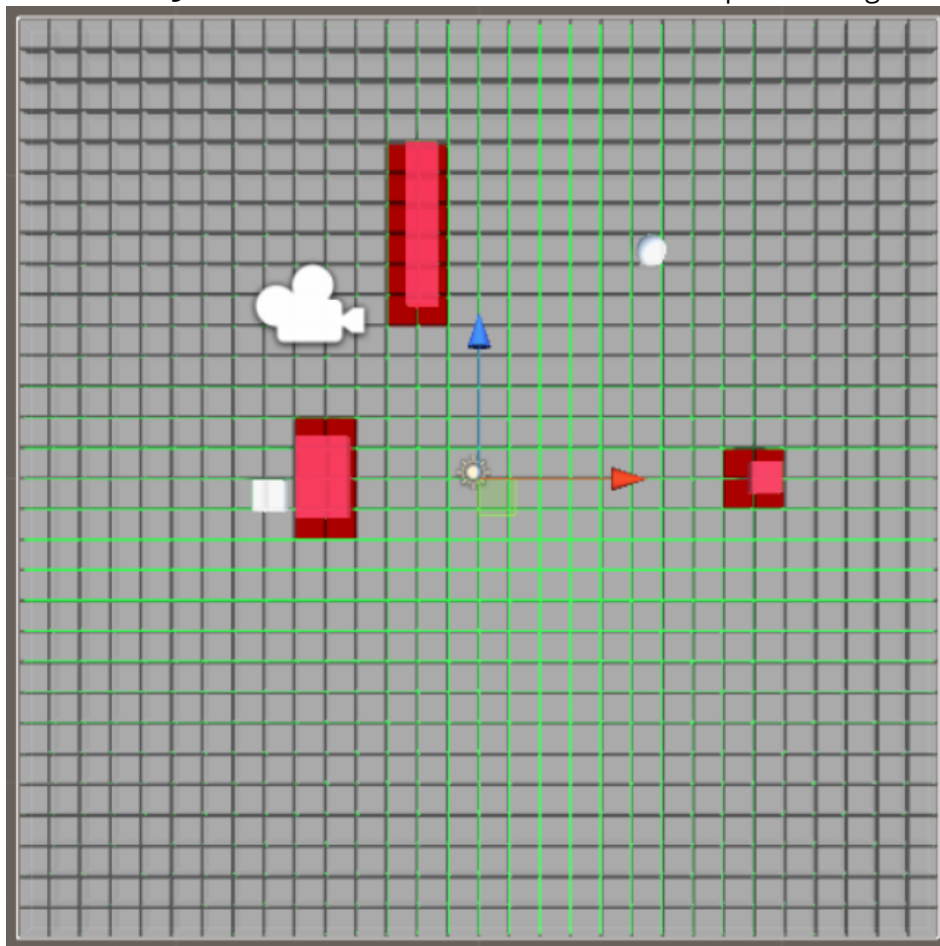
le rythme de création s'accélère et j'ai pu découvrir des techniques plus avancées. Il était ensuite simple d'importer les niveaux dans des scènes sur le logiciel *Unity* et de les utiliser en tant qu'asset.

FIGURE 8 – Niveau de Sébastien sur *Blender 3D*



Pour réaliser le path finding des monstres, j'ai d'abord étudié les différents algorithmes existants (je suis par ailleurs parti du TP C# 5) avant de choisir l'algorithme A* qui est une amélioration de l'algorithme de Dijkstra et est souvent utilisé dans les jeux vidéo. Ensuite j'ai créé une scène contenant un plan, des obstacles, un monstre et un objectif afin d'implémenter cet algorithme et de le tester. Sur la figure 9, on peut voir la grille générée pour créer le path finding : les cases blanches correspondent à celles sur lesquelles le monstre peut marcher, les rouges correspondent à celles où il y a un obstacle.

FIGURE 9 – Vue du dessus de la scène contenant le path finding



Tout au long de mes réalisations, j'ai rencontré différentes difficultés. J'ai d'abord dû apprendre à utiliser *Blender 3D*. Pour cela j'ai recherché des tutoriels sur *Youtube*. Ensuite, j'ai commencé sur *Unity* avec la conférence organisée par l'association *Gconfs : Bien démarrer son projet*. Enfin, des tutoriels m'ont permis de comprendre l'algorithme de path finding A^* même si l'implémentation n'est pas évidente. En tant que chef de projet j'ai également dû faire face à des retards de certains membres du groupe ou à des tensions et découragements passagers.

Pour finir, $\text{\LaTeX}$ est facile d'utilisation même si on peut facilement s'exasperer en voulant un rendu exactement tel qu'on l'imagine.

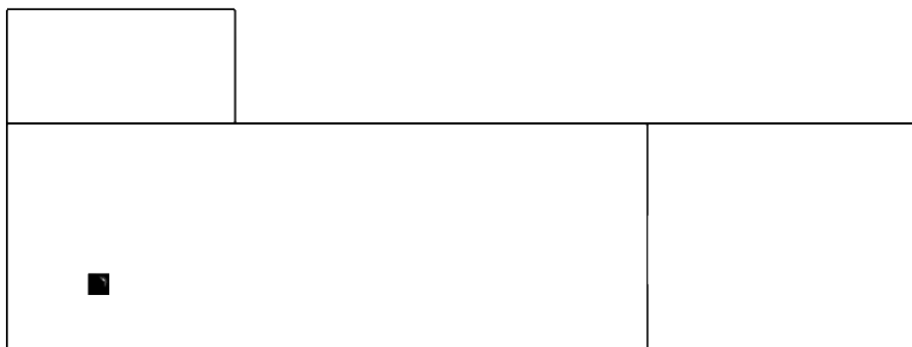
4 Travail pour la prochaine séance

TABLE 2 – Cahier des charges

Tâches	Planning
Niveaux solo	95%
Site internet	100%
Design 2D	100%
Système de portes	100%
Tutoriel	70%
Commandes	100%
IA des monstres	60%
LaTeX	100%
Menu	70%
Map multijoueur	80%
mini-map	20%
Réseau	75%

On peut s'apercevoir grâce au tableau 2 que le tutoriel respecte quasiment les critères de la deuxième soutenance (voir figure 10). Ceci permettra à Morgan de rattraper son léger retard sur le design 2D.

On peut également voir que de nouvelles tâches doivent être débutées. Celles-ci sont : le site internet, la map multijoueur et la mini-map. Par ailleurs, les tâches déjà commencées doivent être continuées et certaines même terminées (systèmes de portes et commandes).

FIGURE 10 – Un des niveaux du tutoriel sur *SketchBook*

5 Conclusion

À travers ce long rapport de soutenance, nous avons abordé plusieurs points très importants que nous allons rapidement reprendre. Voilà maintenant 2 mois (les plus rigoureux diront 1 mois et 26 jours) que nous travaillons sur les projets et à ce stade nous tirons nos premières constatations sur le chemin qui a été fait. Grâce à ces premières données nous avons dans un premier temps comparé l'avancement réalisé avec l'avancement planifié (d'après le cahier des charges) [Voir page 4]. De là, nous pouvons constater que la majeure partie des objectifs ont été remplis dans les temps. À titre d'exemple, parmi la « ToDo » liste importante pour la première soutenance, seul le design 2D est légèrement en retard. À contrario d'autres parties sont largement en avance sur les prévisions comme le tutoriel (60% au lieu de 0%) et d'autres plus légèrement comme le système de portes/boutons ou encore le système de mise en réseau. Il s'agit là d'écarts tout à fait normaux, il est en effet très rare d'atteindre un avancement réel concordant sur tous les points avec un avancement planifié. Point rassurant tout de même, nous n'avons aucun retard important à signaler, ayant la chance d'avoir une équipe d'une part toujours au complet, mais d'autre part composée de personnes capables (jusqu'à présent) de rebondir en cas de difficulté, et cela, dans un bon esprit d'équipe. Dans la suite du document chacun d'entre nous a développé sa perception des choses, comment son travail a été effectué, dans quelles conditions, etc. Sans reprendre point

par point cette partie, de manière générale, cette première partie du projet a été l'objet de découvertes, autant d'un point de vue du travail en groupe que d'un point de vue technique avec la prise en mains de nouveaux logiciels (pour le meilleur et pour le pire...).

Finalement cette première phase de notre projet se conclut sur une évolution qui suit son cours, sans problème majeur pouvant empiéter sur le bon déroulement. Nous arrivons en effet à suivre nos délais tels qu'ils avaient été planifiés et validés dans le cahier des charges.

De plus, nous pouvons assister à une bonne entente au sein du groupe, sur le plan relationnel déjà une mais également sur le plan professionnel, point très important à préciser! Face à de bonnes conditions, nous ne pouvons que nous encourager mutuellement à poursuivre sur cette lancée en restant évidemment à l'écoute des critiques qui pourraient nous parvenir par les encadrants!

FIGURE 11 – Autre design du logo

