

Rapport de soutenance 2

Nathan BERRY
"Ghlyphe"

Nicolas CENDRIER
"Charles"

Arthur PAMART
"Aïto"

Robin LE BIHAN
"Pink"

4 Mai 2016



Table des matières

Introduction	3
1 Rappel du projet et de la première soutenance	3
1.1 Présentation générale du projet	3
1.2 Avancement lors de la première soutenance	3
1.3 Planning de réalisation	4
2 Travail réalisé depuis la première soutenance	4
2.1 Gameplay	4
2.2 Narration	4
2.3 Interfaces	5
2.4 Level design	6
2.5 Réseau	7
2.6 Musiques	8
2.7 Effets sonores	13
2.8 Textures	14
2.9 Modèles 3D	15
2.10 Animations	16
2.11 Site web	16
3 Travail restant à faire	17
3.1 Gameplay	17
3.2 Narration	17
3.3 Interfaces	17
3.4 Level design	18
3.5 Réseau	18
3.6 Musiques	18
3.7 Effets sonores	18
3.8 Textures	18
3.9 Modèles 3D	18
3.10 Animations	19
3.11 Site web	19
3.12 Installation	19
Conclusion	19
Annexe	20



Introduction

Ce document a pour but de vous présenter l'avancement du projet M.O.H.S. de l'équipe Clockwork Games depuis la première soutenance du 15 Mars 2016. Nous ferons tout d'abord un rappel général sur le projet et sur le travail effectué lors de la première soutenance, puis nous expliquerons tout le travail réalisé depuis celle-ci en détaillant chacune des tâches (Gameplay, Level design,...) et nous terminerons par le travail restant à faire pour la soutenance finale.

1 Rappel du projet et de la première soutenance

1.1 Présentation générale du projet

Notre projet est un jeu vidéo répondant au nom de M.O.H.S. (Mars One Human Settlement) et se déroulant dans une base humaine sur Mars servant de décor pour une télé-réalité. Les joueurs incarnent deux astronautes vivant dans cette base qui se réveillent amnésiques, ils découvrent alors que tout le monde a disparu de la base. Ils devront donc progresser dans la base en passant des phases de plateforme et en résolvant des énigmes qui les mèneront vers la vérité sur ce qu'il s'est passé dans cette base.

1.2 Avancement lors de la première soutenance

Quelques mois auparavant, nous étions déjà assez avancés en ce qui concerne la conception de notre jeu. Il était jouable en multijoueur, et une grande partie des éléments du jeu étaient synchronisés en réseau. Lorsque les joueurs démarraient le jeu, ils se retrouvaient directement dans un niveau souterrain, dans lequel ils étaient peu guidés. Ils pouvaient également évoluer dans un étage supérieur, dont la fin était symbolisée par une porte verrouillée par un digicode.

Le jeu en lui-même était non-texturé, en revanche de nombreux modèles étaient déjà présents. Différents menus d'inventaires basiques étaient également implémentés. En terme de gameplay, le ramassage d'objets était fonctionnel. Le joueur était aussi équipé d'un réacteur dorsal, qui lui permettait d'atteindre des endroits normalement inaccessibles.



1.3 Planning de réalisation

Tâches/Périodes	1 ^{re} soutenance	2 ^e Soutenance	3 ^e Soutenance
Gameplay	*	**	***
Narration		*	***
Interfaces	*	**	***
Level design	*	**	***
Réseau	*	**	***
Musiques	*	**	***
Effets sonores		*	***
Textures		**	***
Modèles 3D	*	**	***
Animations	*	**	***
Site web	*	**	***
Installation			***

* : Commencée ; ** : Avancée ; *** : Terminée

2 Travail réalisé depuis la première soutenance

2.1 Gameplay

Le gameplay du jeu se base sur la coopération des deux joueurs, les énigmes proposées dans les parties habitées de la base proposent une résolution à deux qui sera toujours plus aisée qu'une résolution seul. De plus le travail effectué par chaque joueur est différent même s'ils tentent de résoudre la même énigme, c'est le principe du gameplay asymétrique. Nous avons aussi implémenté du gameplay dynamique avec, par exemple, le Rover, qui est dirigeable sans interface. Il suffit de se placer devant l'écran de ce dernier après avoir mis en place la batterie. Au niveau du gameplay "plateforme", rien a changé, le joueur possède toujours un jetpack à oxygène pour l'aider à avancer.

2.2 Narration

Afin que les joueurs progressent à travers les énigmes proposées, ils devront ramasser des clés USB contenant des documents audio. Ces derniers ont été enregistrés par d'autres astronautes de la base. Par exemple lors du premier niveau, en écoutant les trois documents audio, on peut apprendre que la porte menant au code d'accès de la fin du niveau est endommagée et qu'il faut envoyer le Rover dans un conduit afin de la réparer. Les textes des documents ont été écrits par Robin, Nathan et Arthur ont prêté leur voix pour les enregistrer grâce au microphone de Nathan. Les fichiers sont



au format .wav afin de conserver une bonne qualité audio, mais cela implique une plus grande place en mémoire.

2.3 Interfaces

L'inventaire permet au joueur d'accéder auxdits documents audio, en effet les trois interfaces présentes auparavant ont maintenant été fusionnées en une seule, un inventaire plus pratique regroupant le lecteur audio et les objets de quête. Il est possible à travers cet inventaire de naviguer parmi les documents audio afin de sélectionner celui que l'on souhaite écouter, ou bien de choisir quel objet de quête est équipé. Le choix de l'objet équipé actuellement se retrouve aussi dans le HUD où ce dernier est affiché en bas à droite, juste au dessus du compteur de fusées éclairantes. Le raccourci tabulation permet de faire défiler les objets de quête sans entrer dans le menu ce qui permet de proposer une expérience de jeu plus immersive. La barre d'oxygène reste inchangée, sa disposition se prêtant parfaitement à son rôle.

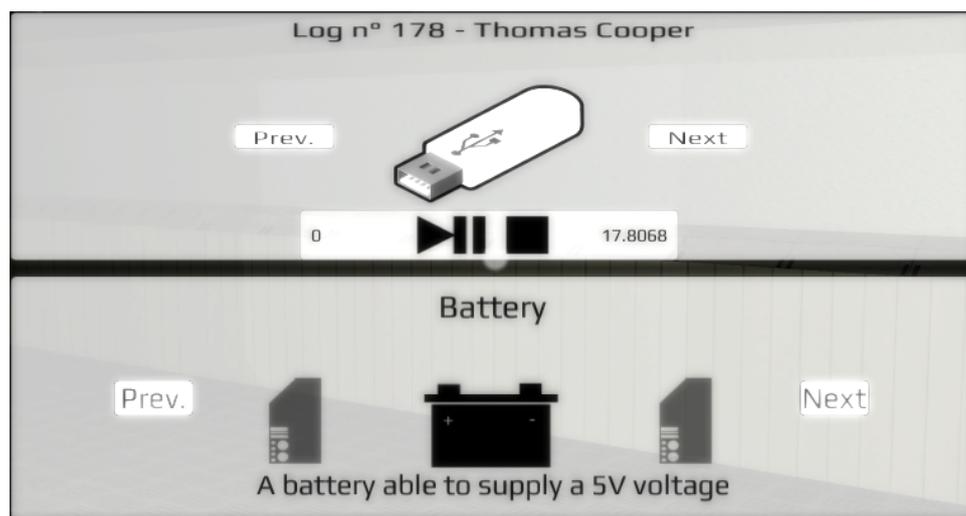


FIGURE 1 – Inventaire

En parallèle, une modification majeure des interfaces réside dans le menu. En effet, inexistant auparavant, le menu apparaît avec de nombreuses fonctionnalités. Ce menu consiste en une pièce ressemblant fortement à un "confessionnal" de télé-réalité. L'écran affiche au lancement du jeu une caméra visant un fauteuil sur la droite, et une panoplie de boutons sur la gauche qui permettent de lancer le jeu en hôte, de se connecter à un serveur, ou d'accéder aux autres parties du menu. Il possède en effet une partie "settings" qui est

caractérisée par un environnement ressemblant fortement aux sous-sols du jeu, une partie "bonus" qui elle ressemble aux parties habitées des niveaux, et une partie "quit game" qui affiche une porte à l'écran et permet de quitter le jeu. Les options permettent pour l'instant de changer le seed de la musique, et de modifier le volume de différents éléments du jeu. Il est probable que le joueur puisse trouver quelques "easter eggs" sympathiques dans ce menu.



FIGURE 2 – Menu principal

2.4 Level design

Dans un souci d'optimisation, il a été décidé de n'afficher qu'une partie de la scène à la fois. Ainsi, une fois que le joueur arrive à la fin de la partie plateforme, il arrive dans un sas. Si le joueur est seul, les portes se ferment immédiatement, alors que s'il y a deux joueurs les portes ne se ferment uniquement lorsque ces derniers sont tous rentrés dans le sas. Une fois les portes closes, un événement se déclenche, la zone avant le sas est désactivée et la zone suivante est activée. Ensuite, la porte s'ouvre laissant les joueurs continuer leur chemin.

Toujours afin d'optimiser, Robin a placé des *triggers* déclenchant l'allumage de certaines lumières. En effet, la présence abondante de lumières était un facteur qui tirait le nombre d'image par secondes vers le bas. Grâce



FIGURE 3 – Fermeture d'un sas

à ces *triggers*, la lumière d'une salle ne s'allume que si le joueur s'y trouve et toutes les autres lumières sont éteintes.

En ce qui concerne le premier niveau en lui-même, il est totalement fini. Toutes les pièces ont été meublées par Robin grâce aux modèles de Nicolas. Robin a également ajouté un labyrinthe fait de conduits d'aération dans lequel le Rover devra progresser. Ce labyrinthe est ponctué d'icônes permettant au joueur d'avoir des points de repères et de résoudre ce labyrinthe.

Le premier niveau étant terminé, Robin s'est attelé à la construction du deuxième niveau. La partie plateforme a été réalisée en accord avec Arthur, le titulaire du *gameplay*. Dans cette partie, les joueurs arrivent dans un couloir se séparant en deux et menant à deux sas. En multijoueur, chacun des joueurs doit aller dans un sas différent, pour qu'ils soient séparés. S'il n'y a qu'un seul joueur, il devra rentrer obligatoirement dans le sas de droite. Le plan de cette partie du niveau est disponible en annexe. La partie énigme est pour l'instant vide et ne propose rien de spécifique.

2.5 Réseau

Pour ce qui est du réseau, les bases ayant déjà été posées dès la première soutenance, le seul travail de Robin consistait à synchroniser au fur et à mesure les différents éléments de gameplay. Il a fallu, par exemple, synchroniser les mouvements du Rover. Une autre tâche du réseau était de synchroniser le changement de scène. Ainsi, lorsque l'un des deux joueurs entre le bon digicode, ils sont immédiatement transportés dans la scène suivante.

2.6 Musiques

Nous avons conservé la structure musicale déjà présente précédemment : celle-ci dispose d'une musique composée et d'une musique générée procéduralement.

Les modifications apportées à la musique composée sont plus légères : une modification notable est une montée progressive du volume de la musique lors de son démarrage, afin de ne pas « agresser » le joueur dès son arrivée dans le menu principal. Par conséquent, afin de combler le vide sonore laissé par la montée progressive du volume de la musique, le joueur pourra entendre un simple son binaural, c'est-à-dire des battements de haute fréquence qui ne sont pas réguliers entre les deux oreilles, ce qui donne un effet « spatial » qui convient à l'ambiance générale du jeu.

La musique procédurale, quant à elle, a fait l'objet d'une reconception complète : l'approche générale a été revue et s'est donc suivie d'une réécriture. La raison était que les règles simplifiées de composition musicale donnaient une harmonie que Nathan ne trouvait pas satisfaisante, notamment en ce qui concerne la différenciation entre les passages plus calmes et les moments stressants.

Tout d'abord, il faut savoir que la musique procédurale, comme toute musique « normale » est constituée d'accords, c'est-à-dire de groupes de notes graves organisées, et d'une mélodie, soit une suite de notes aiguës choisies en fonction de l'accord et de règles musicales complexes.

Pour pouvoir comprendre pleinement son évolution, il est intéressant de comparer son fonctionnement lors de la première soutenance et de voir tous les changements progressivement apportés.



Au début, l'algorithme possédait une « table » contenant toutes les gammes possibles, puis une deuxième « table » constituée des accords possibles selon la gamme, et enfin trois dernières « tables » pour connaître les notes possibles selon un accord donné (Une table pour les accords majeurs, une pour les accords mineurs et une pour les accords diminués).

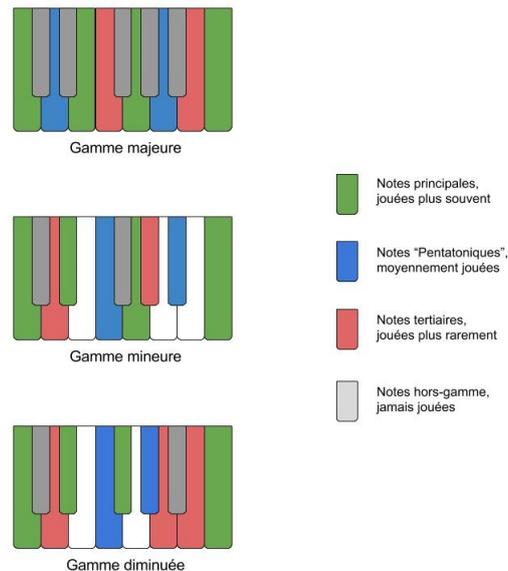


FIGURE 4 – Ancienne chartre des notes

Enfin, pour générer des notes, le programme appelait la fonction « Random » de Unity pour générer un nombre aléatoire, qui servait à déterminer quelle sera la prochaine note (une note après, une note avant, deux notes avant, etc. . .), toujours en restant dans les notes de la gamme.

Depuis, le nombre d'éléments nécessaires à une musique agréable l'oreille a grandement augmenté. Une refonte complète de la génération de nombre aléatoire, du système de gamme, d'accords, de mélodie et de temps a été nécessaire.

Tout d'abord, Nathan a modifié la méthode de génération de nombres aléatoires, et ce pour deux raisons : le programme utilisait avant la méthode « Random » de Unity, mais cette dernière consommait trop de ressources alors que nous n'avions pas besoin d'une génération de nombre si complexe. De plus, pour pouvoir travailler plus facilement sur les fonctionnalités de la musique, il était intéressant d'avoir une musique qui était « fixe », et non soumise au hasard.

Pour atteindre ces deux buts, Nathan a décidé de remplacer toute utilisation de la méthode « Random » par un algorithme simple de génération de nombre. Des recherches approfondies sur Internet ont permis de déterminer qu'un algorithme de type « $(ax + b) \bmod c$ » utilisant un nombre de départ nommé « seed », serait idéal. Un tel algorithme donne une suite de nombres fixe, et doit s'assurer que $(a-1)$ et que c soient premiers pour que la boucle générée soit de taille maximale, c'est-à-dire de la taille du modulo.

Une fois que l'on a des nombres générés "aléatoirement", on peut commencer à voir la construction de la musique elle-même. Dans un premier temps, les différentes notes d'une musique sont réparties sur une "gamme", c'est-à-dire un ensemble de notes fixe prévu à l'avance.

Ici, les notes sont choisies à partir de trois types de gammes, selon le niveau de stress. La première gamme, nommée « Gamme majeure », est utilisée pour un stress léger, et semble plus calme et reposante. La deuxième gamme, quant à elle, est utilisée pour un stress moyen et semble un peu plus triste et mélancolique. La dernière est utilisée en cas de grand stress, et semble plus déconcertante, voire stressante.

Il convient de se rappeler que les schémas ci-après représentant les trois types de gamme sont représentés avec la note « Do » (ou « C ») comme note de départ. Pour toute autre note de départ, l'algorithme décale toute la table pour correspondre à cette dernière. Par exemple, si la note de départ est le « Ré », qui se trouve deux notes à droite du « Do », l'algorithme décalera la table de deux notes vers la droite pour la faire correspondre au « Ré ».

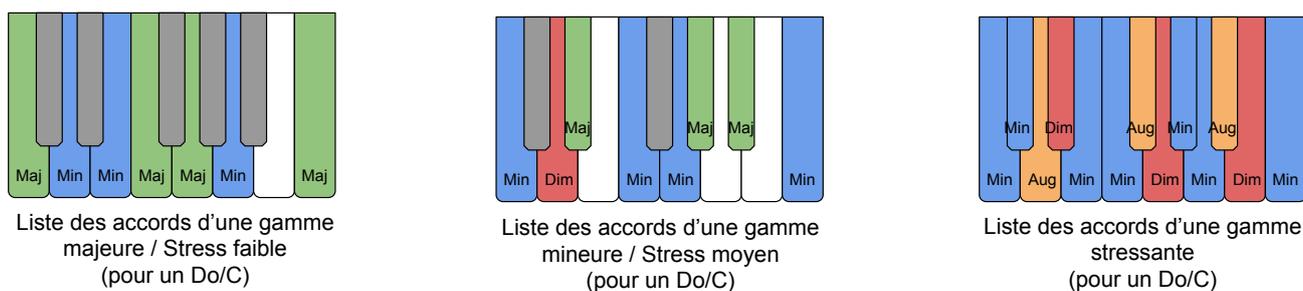


FIGURE 5 – Chartre des accords en fonction d'une gamme donnée

À partir de ces différentes gammes, l'algorithme choisit à intervalle régulier des accords. Ceux-ci sont générés entièrement à partir de nombres "aléatoires", et peuvent être de plusieurs types : les accords majeurs qui sont plus joyeux et calmes, les mineurs qui sont plus tristes, les diminués qui font des sons plus « lourds » et les augmentés qui sont stridents. Il faut bien noter que les gammes majeures n'utilisent que les deux premiers types d'accords, alors que les gammes mineures n'excluent que les accords augmentés. De la même façon, les gammes stressantes n'utilisent pas les accords majeurs. Il faut savoir que le temps entre deux accords est un intervalle régulier appelé « mesure ».

Ensuite, à partir de chaque accord joué est générée une table de notes mélodiques qui pourront être jouées à la main droite, et celles-ci sont réparties entre quatre niveaux de consonance.

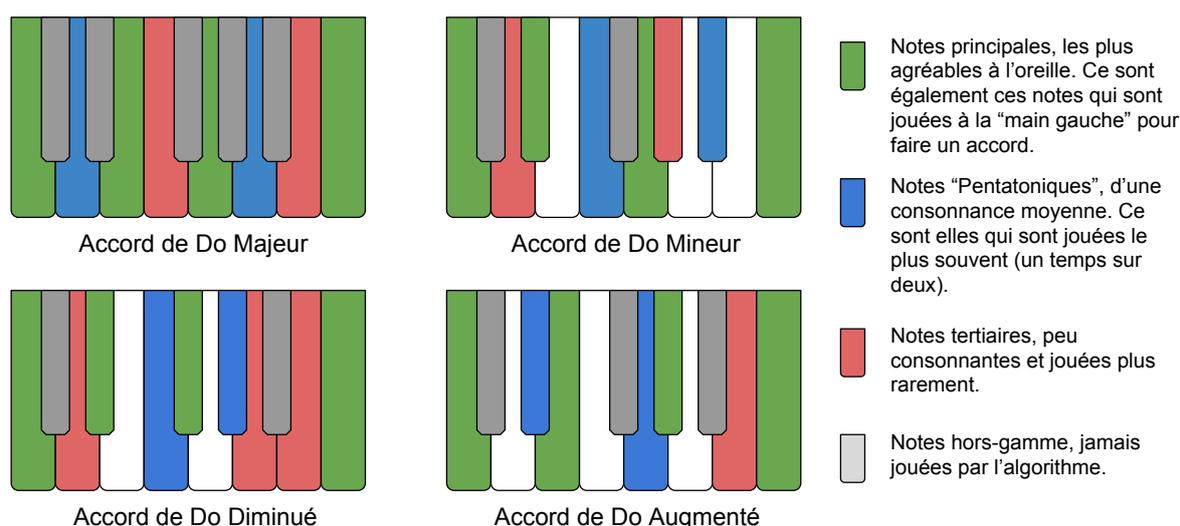


FIGURE 6 – Charte des notes mélodiques en fonction d'un accord donné

En ce qui concerne le choix de la note en elle-même, l'algorithme prend comme point de départ la note précédemment jouée. À partir de ce point, cinq événements sont possibles :

1. On sélectionne la prochaine note au-dessus de la note actuelle qui correspond au niveau de consonance du temps en cours ;
2. On sélectionne la prochaine note en dessous de la note actuelle qui correspond au niveau de consonance du temps en cours ;

3. On sélectionne la prochaine note au-dessus de la note actuelle qui correspond à un niveau de consonance au-dessus de celui du temps en cours ;
4. On sélectionne la prochaine note en dessous de la note actuelle qui correspond un niveau de consonance au-dessus de celui du temps en cours ;
5. On sélectionne la note fondamentale de l'accord en cours.

Par exemple, si on est sur un temps « bleu » et que la note précédente est un sol :

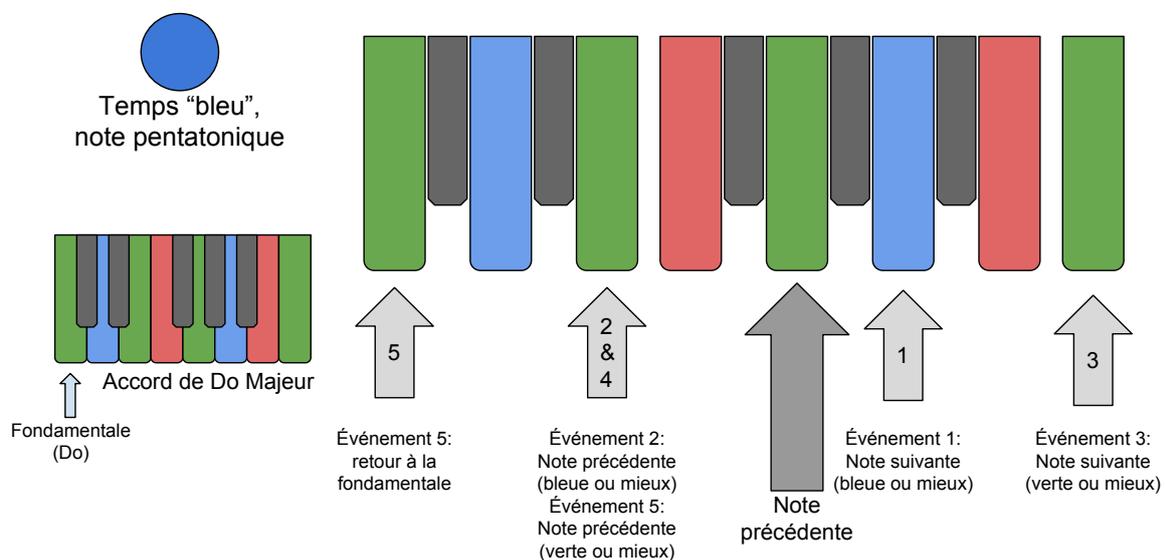


FIGURE 7 – Schéma des événements possible à chaque changement de note

Ensuite, ces cinq événements sont associés à cinq éléments d'un tableau de probabilité. Ils sont fixés au départ de telle sorte à ce que leur somme équivaille à 80%. À chaque fois qu'un événement se produit, sa probabilité associée baisse, et les autres probabilités augmentent. Cela permet de moduler la probabilité de chaque événement en temps réel afin d'éviter qu'ils ne se répètent trop. Les 20% restants correspondent à une probabilité fixe qu'un silence, c'est-à-dire un temps où aucune note n'est jouée, soit joué à la place d'une note classique.

Enfin, la musique procédurale possède également un découpage dans le temps, divisé en ce qu'on appelle "mesure". Une mesure est divisée en un certain nombre de temps, nombre qui dépend du niveau de stress. à chacun de ces temps, on associe un certain niveau de consonance : une note joué pendant ce « temps » sera d'un niveau de consonance supérieur ou égal à celui demandé : les notes « vertes » sont donc toujours valides, les notes « bleues » pourront seulement êtres jouées pendant un temps « bleu » ou « rouge » et une note « rouge » seulement pendant un temps « rouge ».

En cas de stress bas, le nombre de temps par mesure est fixé à huit. Plus le stress augmente, et plus le nombre de temps est variable : à 100 de stress, c'est-à-dire le maximum, le nombre de temps par mesure varie entre trois et treize.

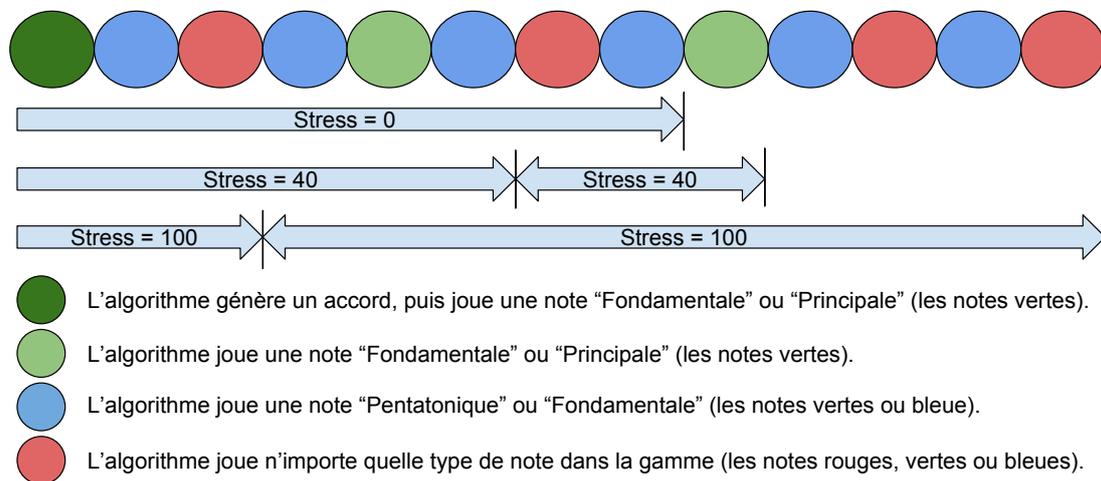


FIGURE 8 – Charte des notes jouées en fonction des temps d'une mesure

Au final, l'algorithme dépend d'un générateur de nombre aléatoire limité qui reboucle sur lui-même après un grand nombre d'occurrences, et chaque note jouée dépend de ce nombre, mais aussi de la note précédente, d'un tableau de probabilité dynamique et du stress. On a donc la certitude, étant donné le nombre de possibilités, que le caractère répétitif de cette musique procédurale ne pourra pas être repéré par une oreille humaine.

2.7 Effets sonores

Le jeu a été agrémenté de plusieurs effets sonores qui participent activement à l'immersion du joueur dans l'univers qui lui est proposé.

Tout d'abord, des sas ont été implémentés à plusieurs points clefs du jeu. Pour des raisons narratives, un jet de fumée est projeté sur les joueurs.



Afin d'accompagner cet événement visuel, un simple son de libération d'air compressé a été ajouté.

Dans la même lignée, un son similaire a été ajouté lors de l'ouverture d'une porte : deux sons d'air compressés ont été accélérés et superposés, et un troisième son de lourd claquement métallique a été rajouté au début. À l'inverse, pour la fermeture des portes, nous avons inversé les sons de compressions d'air, et le bruit métallique a été placé à la fin.

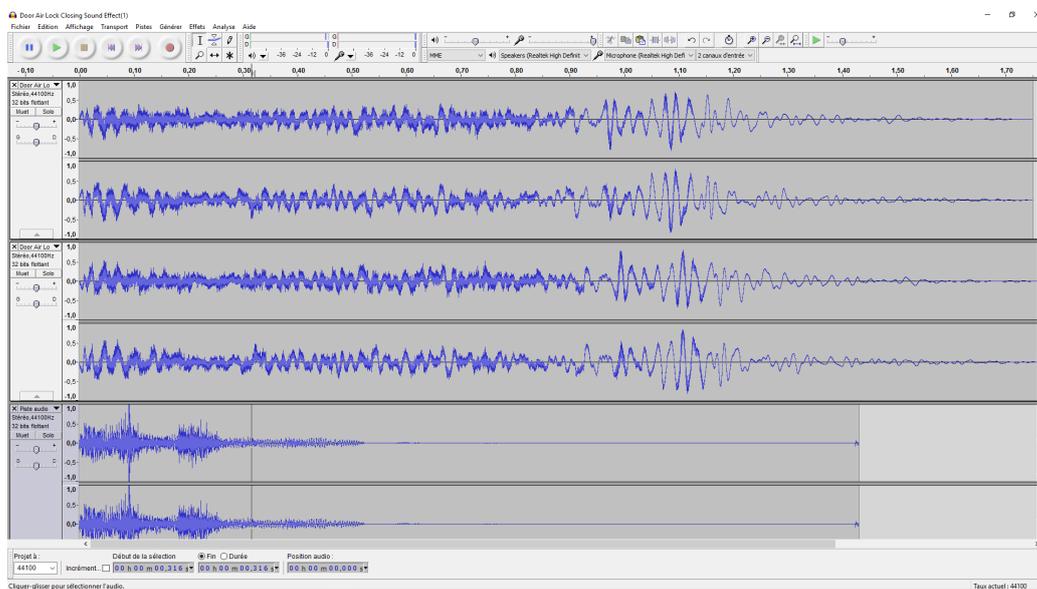


FIGURE 9 – Les trois sons superposés pour l'ouverture de la porte

Les bruits de pas et celui du crépitement des fusées éclairantes étaient déjà présent auparavant, le jeu a désormais une bonne base d'effets sonores.

2.8 Textures

Pour cette soutenance, Nicolas s'est attelé a une partie importante des graphismes : les textures. Ce sont elles qui donnent toute l'ambiance, le thème et le ton du jeu. Il a donc cherché et téléchargé des textures photo-réalistes gratuites. Après avoir sélectionné ses textures, il les a réduites et retravaillées sur GIMP pour qu'elles soient moins lourdes en mémoire et



qu'elles s'ajustent correctement sur un même modèle. Il a alors assigné sur Blender à chaque texture un "material" qu'il a ensuite appliqué à certaines parties de ses modèles (revêtement de sol au sol...). Puis, sur Unity, il a appliqué chacune des textures aux modèles présents dans la scène et a commencé à s'occuper des détails des textures : grâce aux différents outils de Unity, Nicolas a réalisé une "normal map" de chacune de ses textures, celles-ci servant à donner plus de réalisme à une texture et à lui donner un certain relief. Ensuite, il a mis à l'échelle chacune des textures pour qu'elles soient de la même taille dans le rendu en jeu. En effet, les patrons Blender réalisés ne sont pas tous de la même taille en fonction du modèle et pas forcément dans le bon sens (scale). Il s'est ensuite occupé de les joindre entre elles sur différents modèles en les centrant plus ou moins sur ceux-ci (offset). Ainsi, à l'heure qu'il est, les textures sont appliquées aux couloirs, sols, plafonds et murs de l'environnement du joueur. Elles donnent sa véritable ambiance au jeu et le rendent plus "fini". Il reste encore quelques défauts/détails à régler pour avoir un meilleur rendu.

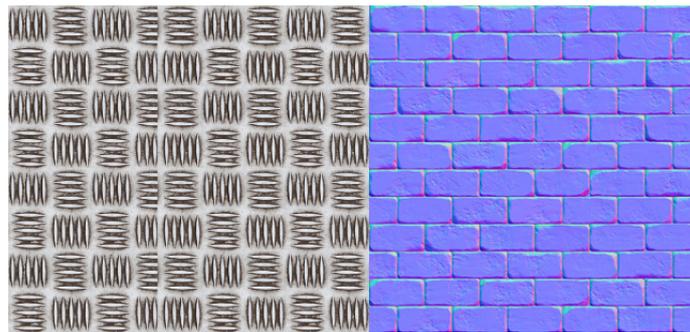


FIGURE 10 – Exemple de texture non alignée et de normal map

2.9 Modèles 3D

En ce qui concerne la tâche des modèles 3D, Nicolas, toujours grâce à Blender, a réalisé de nouveaux modèles utiles à l'aménagement du décor. Par exemple, pour le dortoir, des lits et des étagères ont été créés. C'est d'ailleurs depuis ce dortoir que l'on peut contrôler le Rover qui lui aussi a été modélisé sur Blender, les écrans ont par conséquent reçu une refonte pour pouvoir accueillir des "renders textures" de Unity sur lesquelles est affichée la vision du Rover. D'autres pièces de la base ont aussi obtenu de nouveaux modèles tels que la salle d'eau commune, avec des cabines de douche/toilettes et des robinets, mais aussi la salle à manger/salon avec des chaises. Ces nouveaux modèles permettent de meubler notre scène et rendent les salles moins vides, plus habitées et réalistes.

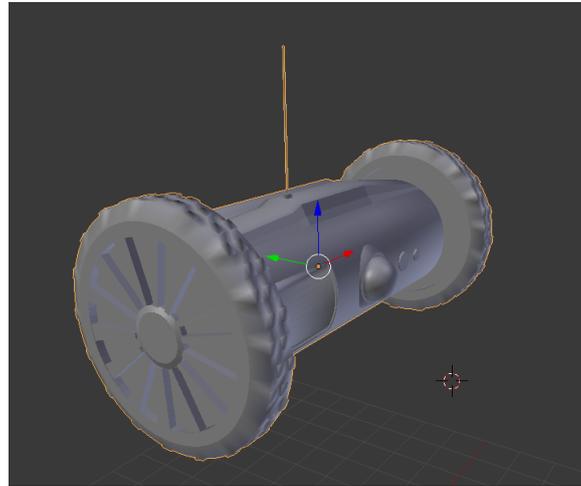


FIGURE 11 – Le Rover modélisé sur Blender

2.10 Animations

Le système d'animations a été complètement retravaillé. En effet les animations étaient avant gérées séparément, c'est à dire qu'il y avait un nombre défini d'animations possible. Maintenant, un "Blend tree" permet de mélanger des animations afin d'obtenir par exemple des animations de marche en diagonale.

2.11 Site web

Le site web, quant à lui, est presque entièrement fini, les parties sont dans l'ensemble rédigées et le style graphique bien défini : on garde donc un fond noir pouvant représenter l'espace et des polices, gratuites, modernes blanches ou oranges en fonction de leur utilisation (titres ou texte). Des nouveaux liens de téléchargement et des sous-parties ont été ajoutées depuis la dernière soutenance rendant la page plus remplie et détaillée. Le site est donc maintenant composé d'une partie présentation du projet avec une description de notre jeu et quelques capture d'écran, une partie présentation du groupe, avec sa création et la présentation des membres en sous parties, une partie téléchargement plus complète, avec les PDFs de nos soutenances et de l'utilisation du jeu et enfin une partie contact.

3 Travail restant à faire

3.1 Gameplay

Nicolas en tant que suppléant au gameplay a commencé à coder un mini-jeu dont il a eu l'idée. Le but du mini jeu est de réparer un boîtier de câbles électriques. Pour cela le joueur doit relier les différents câbles sectionnés entre eux afin de rétablir le courant. La résolution de cette énigme se base sur un sorte d'équation a plusieurs inconnues dont on connaît le résultat, Chaque partie de câble ayant sa propre valeur et les parties reliées entre elles représentent un opérateur (multiplication, addition...). Sur le schéma ci dessous le chemin relié vaut $3 * 8 + 6$ ce qui donne bien 30 : la lampe est donc allumée. Ce mini-jeu sera aussi affecté par un niveau difficulté : le nombre de fils et la valeur des nombres augmentera lorsque celui ci sera élevé et inversement. Il sera aussi jouable en multijoueur ou chacun des joueurs aura accès a une partie seulement du tableau électrique et devra communiquer les nombres présents sur sa partie à son coéquipier. Le principe du jeu est entièrement codé mais il reste à l'implémenter dans Unity et ainsi réaliser la partie graphique et la partie interaction avec le joueur ainsi que la synchronisation multijoueur pour permettre d'y jouer à deux.

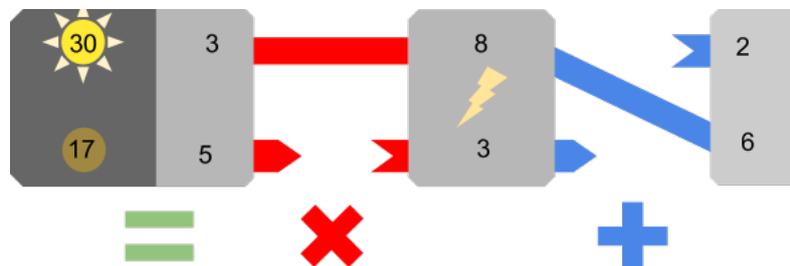


FIGURE 12 – Exemple du mini-jeu avec une faible difficulté

3.2 Narration

Pour la prochaine soutenance, il faudra écrire et enregistrer les documents audio servant aux futures énigmes.

3.3 Interfaces

En plus de l'interface des mini-jeux, présentés précédemment, les sections "Bonus" et "Settings" du menu principal restent à compléter.

3.4 Level design

En ce qui concerne le level design, il faudra créer de nouveaux niveaux (plateforme et énigme).

3.5 Réseau

Pour la partie réseau, il suffira de synchroniser les différents éléments de gameplay qui seront présents dans les futurs niveaux.

3.6 Musiques

En ce qui concerne la musique composée, elle nécessite seulement un enregistrement avec de vrai instruments pour un rendu plus agréable. Cependant, on peut considérer la composition d'autres musiques pour d'éventuelles parties du jeu qui en aurait besoin.

La musique procédurale, quant à elle, est toujours améliorable. L'art de la composition musicale est extrêmement compliquée à retranscrire en algorithme. Cependant, on peut imaginer la mise en place de différents style de musique selon les zones de jeu, ou l'inclusion d'autre instruments.

3.7 Effets sonores

En terme d'effets sonores, les bruitages principaux sont en place. Cependant, on peut considérer l'ajout d'autres sons plus légers pour appuyer un peu plus l'ambiance générale du jeu, qui se veut par moment stressante.

3.8 Textures

Pour la prochaine soutenance il reste encore à appliquer des textures à des objets moins importants, cela rajoutera à l'ambiance et au réalisme de la scène et permettra une meilleure immersion. Il reste encore aussi à rendre le rendu graphique plus "parfait" en continuant le travail commencé par Nicolas, qui est d'ajuster chacune des textures pour qu'elles "collent" entre les différents modèles sans être coupées ou décalées. Quelques changements de textures seront peut être effectués pour que le rendu plaise à l'ensemble du groupe.

3.9 Modèles 3D

La tâche des modèles est en grande partie terminée. Cependant il se peut que nous ayons besoin de nouveaux modèles en fonction de ce que l'on va ajouter comme éléments de jeu pour la troisième soutenance.



3.10 Animations

Au niveau de animations, il manque une animation de saut. Il se peut que soit rajoutées des animations de réacteur dorsal et d'action.

3.11 Site web

Le site web n'est pas encore totalement fini : il reste encore à héberger notre page et donc faire en sorte que les liens de téléchargement soient fonctionnels. Le site pourra aussi, éventuellement, être développé pour version mobile (tablettes, téléphones...) et le rendre un peu plus interactif et dynamique avec une galerie d'image ou des menus déroulants par exemple.

3.12 Installation

Le jeu en soi n'a pas encore d'installateur à proprement parler, il faut donc l'implémenter.

Conclusion

En conclusion, nous sommes satisfaits de l'avancement de notre projet. L'aspect graphique semble de plus en plus satisfaisant, et le choix artistique qui s'était dégagé lors de la précédente soutenance est maintenant pleinement assumé.

De plus, le jeu est désormais jouable. La notion de "Gameplay Asymétrique" abordée dans le cahier des charges est maintenant concret, et le résultat est plaisant.

Au niveau de la conception, le jeu n'a aucun retard à déplorer. Il avance comme prévu et nous espérons que ce rythme continuera jusqu'à la dernière soutenance.



Annexe

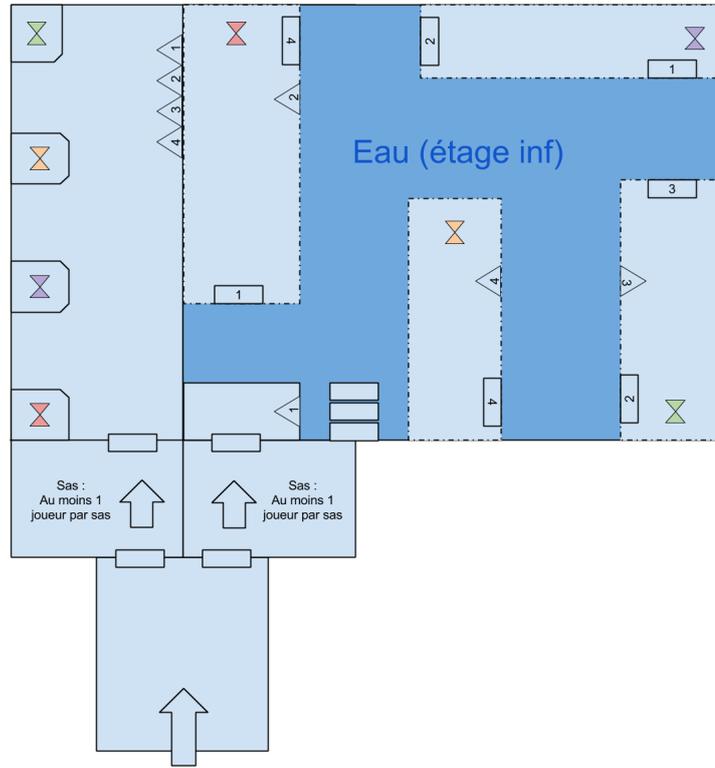


FIGURE 13 – Plan de la partie plateforme du niveau 2