

Soutenance 2

Black Hole

Avril 2023



Ronan LEBOUCHER, Lorenzo LOMBARDI, Thomas MORIN, Brian PERRET

Table des matières

1	Introduction	4
2	Objectifs	4
2.1	Pour cette soutenance	4
2.2	Pour la prochaine soutenance	4
3	Networking	4
3.1	Introduction	4
3.2	Custom Network Lobby	5
3.3	Gestion du changement de salle	5
3.4	Points d'apparition des joueurs	5
4	Progression du joueur	6
4.1	Introduction	6
4.2	Le Hub	6
4.3	GameManager	6
5	Sound design	8
5.1	Introduction	8
5.2	SFX	8
5.3	Mixage	8
6	Optimisation du jeu	8
6.1	Introduction	8
6.2	Qualité de rendu	9
6.3	Occlusion culling	10
6.4	Benchmark	11
7	IA	12
7.1	Introduction	12
7.2	Nouvelle mécanique de jeu	12
7.3	Drone	14
8	Création de salles	15
8.1	Hub	15
8.2	Salle 3	16
8.3	Salle 4	16
8.4	Salle 5	17
9	Modèles 3D	19
9.1	Introduction	19
9.2	Boule de Plasma	19
9.3	Générateur	20
9.4	Leurre	20
9.5	Support	20

10 Conclusion	20
10.1 Avance sur le planning	20
10.2 Retard sur le planning	20

1 Introduction

Bonjour, nous sommes Black Hole, l'équipe se compose de Brian, chef de groupe et développeur, Thomas et Ronan qui sont game designers ainsi que notre graphiste : Lorenzo.

Notre équipe est derrière le jeu de reflexion/enigmes nommé UpSide Down se basant sur l'inversion de la gravité grâce à l'objet principal du jeu nommé la "GraviSphère". Le jeu se compose de multiples salles d'enigmes à résoudre en duo. Pour cela le joueur peut interagir avec différents objets tels que des plaques de pression ou encore des cubes que l'on peut déplacer et inverser leur gravité permettant d'ouvrir une porte ou autre mécanisme aidant à la résolution de la salle.

2 Objectifs

2.1 Pour cette soutenance

Pour notre deuxième soutenance, nous voulions avoir toutes les salles d'enigme fonctionnelles et praticables pour les joueurs (sous réserve d'en ajouter ou d'en enlever au fil de notre réflexion).

Ainsi que l'amélioration du multijoueur dans la gestion des changements de salles avec des écrans de chargement.

Nous avons enfin pour projet de faire une IA utilisant le "path-finding" pour trouver le chemin le plus court vers le joueur portant l'objet servant à finir le niveau. L'IA n'apparaîtra que dans certains niveaux et sera représenté par un robot. Si le robot rentre en contact avec le joueur, ce dernier lâchera l'objet qui se désintégrera et réapparaîtra à son emplacement d'origine, obligeant les joueurs à tout recommencer. Cela créera une dynamique où un joueur devra porter l'objet, le rendant vulnérable, pendant que l'autre joueur, en plus d'interagir avec le niveau, devra le protéger du robot en interagissant avec lui pour soit le ralentir, soit le détruire.

2.2 Pour la prochaine soutenance

Finir le jeu. Donc finir la salle 6, paufiner la décoration des autres salles, implémenter l'histoire et le barman qui constituera un guide pour le joueur ainsi qu'un fil entre chaque salle avec un ressort humoristique.

3 Networking

3.1 Introduction

Notre jeu a la particularité d'utiliser la bibliothèque Mirror pour la gestion du multijoueur. Mirror permet la mise en place d'un mode multijoueur rapidement, mais cela reste quelque chose de très basique. Cependant, Mirror offre la possibilité de customiser le système de multijoueur pour créer un système bien plus complexe et modelable.

Notre jeu a justement besoin d'une telle customisation, car celui-ci utilise le chargement de carte durant l'exécution du jeu sans déconnecter les joueurs (à chaque fois que le joueur utilise un portail de téléportation). Ainsi, le système d'apparition des joueurs demande une gestion particulière.

3.2 Custom Network Lobby

La classe Network Lobby a pour fonction de faire la gestion des connexions des différents clients qui sont connectés au serveur.

De plus, grâce à cette classe, nous avons la possibilité de récupérer un ensemble d'Event tels que : - La connexion et la déconnexion d'un joueur - Le statut du serveur (le nombre de joueurs, en chargement, est prêt) - Le statut de la carte (en chargement, est prêt)

3.3 Gestion du changement de salle

Comme dit précédemment, le joueur sera amené à utiliser à plusieurs reprises des téléporteurs qui ont pour effet de charger une nouvelle carte.

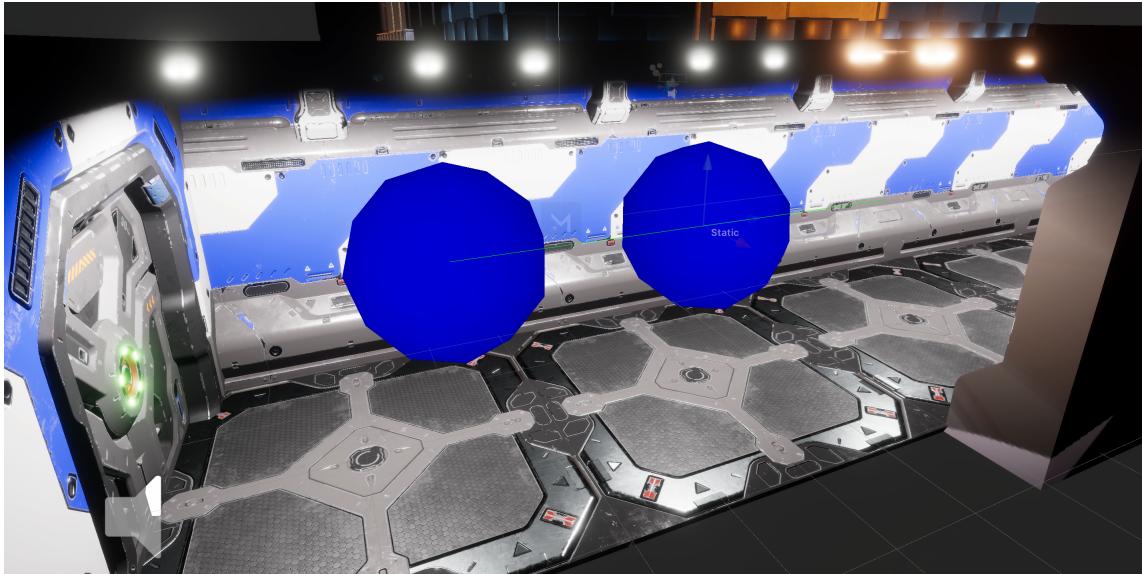
En tant que tel le chargement de carte n'est pas compliqué, il suffit d'utiliser une fonction loadScene du Network Lobby (vue précédemment). Cette fonction sera utilisée lorsque le téléporteur détectera deux joueurs dans leurs portails respectifs.

Mais ce processus n'est pas seulement résumé au chargement de la carte. En effet, comme le chargement des cartes n'est pas une méthode asynchrone, il faut donc afficher au préalable l'écran de chargement.

Il faut donc faire en sorte que les joueurs reçoivent l'information qu'un chargement de carte va être effectué pour qu'ils affichent l'écran de chargement. Pour cela, il existe différentes méthodes. Pour notre cas, nous avons préféré choisir la facilité grâce à l'utilisation des méthodes "sendMessage" fournie grâce à la bibliothèque Mirror. Cela permet d'envoyer à tous les clients un "message" que l'on peut récupérer pour transmettre une information. Dans notre cas pour afficher l'écran de chargement.

3.4 Points d'apparition des joueurs

L'apparition des joueurs et aussi sa réapparition sont des choses importantes dans un jeu. En effet, il est frustrant dans un jeu où la mort est présente de devoir réapparaître au début du niveau. C'est pourquoi nous avons mis en place un système de points de réapparition (spawnpoints). Justement, Mirror offre de base un système de points d'apparition que l'on peut placer, mais celui-ci utilise l'aléatoire pour le point d'apparition parmi la liste des points placés. Néanmoins, certaines salles d'éénigme séparent physiquement les deux joueurs, on ne veut donc pas que les deux joueurs se retrouvent au même endroit. Cette contrainte nous oblige à devoir créer un système d'apparition adapté à nos besoins. Pour cela, nous avons attribué un point d'apparition par joueur, et, lorsque le joueur passe par un "checkpoint" cela vient déplacer son point d'apparition aux coordonnées de "checkpoint".



Vue sur les spawnpoints (debug mode) avec un trait pour la rotation du joueur

4 Progression du joueur

4.1 Introduction

Pour que les joueurs ne soient pas perdus durant leur progression et aussi qu'il ne soit pas possible de commencer par la fin. Nous avons mis en place un système de progression forçant les joueurs à faire les niveaux dans l'ordre que nous avons prévu au préalable.

4.2 Le Hub

Pour cela, nous faisons en sorte que chaque téléporteur s'active en fonction du niveau de progression du joueur. Ce même niveau augmente dès qu'ils terminent une salle.

Ainsi, au commencement du jeu, le seul téléporteur ouvert qui est aussi le plus proche du point d'apparition est le téléporteur vers la salle du didacticiel.

4.3 GameManager

Nous avons aussi prévu que cette progression puisse être sauvegardée par les joueurs pour pouvoir reprendre ultérieurement le jeu sans devoir tout recommencer.

Pour ce faire nous utiliserons une variable dans la classe GameManager en commun entre les deux joueurs, or cette même variable doit être accessible dans n'importe quelle salle où se trouvent les joueurs. Pour répondre à ces deux problématiques, il faut qu'une variable soit synchronisée et aussi avec l'attribut "don'tDestroyOnLoad" qui permet de garder un objet entre plusieurs scènes et donc de ne pas perdre les données stockées à l'intérieur. Néanmoins, Mirror n'offre pas la possibilité de faire ces deux choses sur la même variable.

Pour répondre à cette problématique, nous procédons à une séparation de cette variable en deux, l'une sera synchronisé tout le long du niveau, et lorsque le joueur utilisera un téléporteur, il suffit de mettre à jour la seconde variable et donc de garder cette même valeur avec le changement de carte.



Vue sur un téléporteur, un portail pour chaque joueur.
 Au premier plan, le seul téléporteur actif, la progression est celle du tout début.
 Les autres téléporteurs (à l'arrière plan) sont inactifs.



Au fur et à mesure que les joueurs avancent dans le jeu, les téléporteurs se débloqueront.

5 Sound design

5.1 Introduction

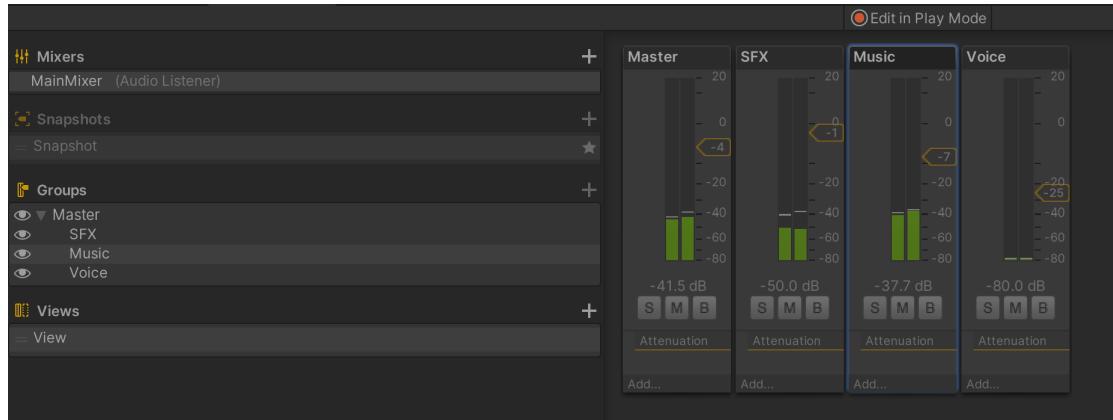
Le son est l'une des choses les plus importantes pour l'expérience. Dans le domaine de l'audiovisuel, on peut montrer son importance grâce à un simple fait. Il est bien plus convenable de regarder une vidéo avec un bon son et une mauvaise image qu'une vidéo avec un mauvais son et une bonne image.

5.2 SFX

Il existe différents types pour les sons : - Les sons d'ambiance, typiquement la musique de fond.
 - Les effets spéciaux (SFX) : bruit de pas, bouton pressé, ...

5.3 Mixage

Dans Unity, le mixage du son se fait grâce à un AudioMixer. Il permet notamment de séparer les niveaux de son et de faire des sliders pour laisser au joueur la possibilité de régler ces différents paramètres.



Vue sur le mixer audio du jeu

6 Optimisation du jeu

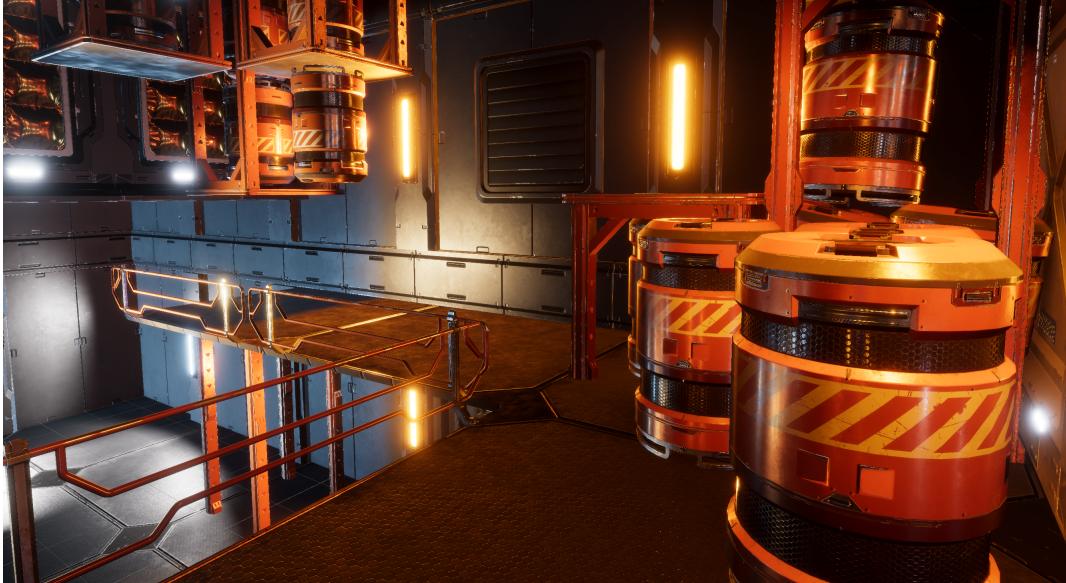
6.1 Introduction

L'optimisation de notre jeu a été l'une de nos priorités pour notre seconde soutenance. En effet, comme notre jeu utilise un rendu des lumières poussé (HDRP) cela a pour conséquence d'utiliser une forte puissance de calcul, que nous avons pu fortement limiter grâce au pré-calcul des lumières. Néanmoins, cela a pour conséquence d'utiliser une partie non significative de mémoire et notamment de mémoire vidéo. Ce qui nous a valu quelques remarques lors de notre première soutenance.

Cela est aussi l'un de nos objectifs principaux pour cette soutenance, c'est-à-dire de faire en sorte d'avoir un jeu fluide même sur un pc sans une carte graphique dédiée et 8Go de ram.

6.2 Qualité de rendu

Ces étapes d'optimisation commencent par mettre en place différentes échelles de qualité misent à la disposition du joueur. Pour cela, nous configurons des fichiers contenant les paramètres de rendu (low, medium, high) Ces fichiers modifient de nombreux paramètres tels que la qualité des ombres, des lumières, des textures ...



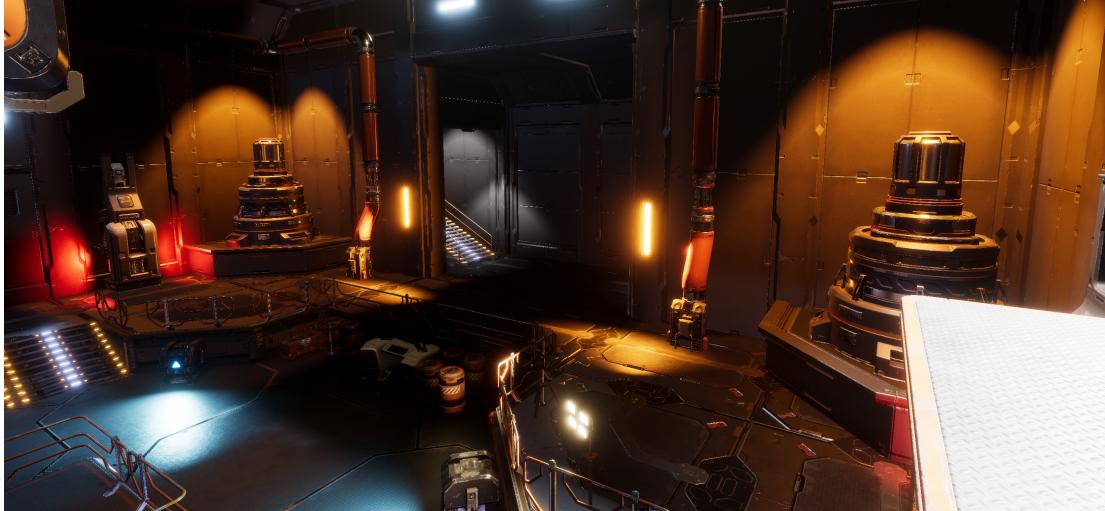
Vue sur la salle 2, qualité : low



Vue sur la salle 2, qualité : medium

Par ailleurs, nous avons aussi remarqué que les shadow maps (ombre dynamique) consomment

une grande quantité de ressources mais rendent le jeu bien plus joli au niveau des ombres. Cependant, sans les ombres dynamiques, le jeu reste très joli. Nous avons donc prévu de mettre directement ce paramètre de rendu modifiable par le joueur.



Vue du didacticiel sans shadow maps (ombre dynamique)

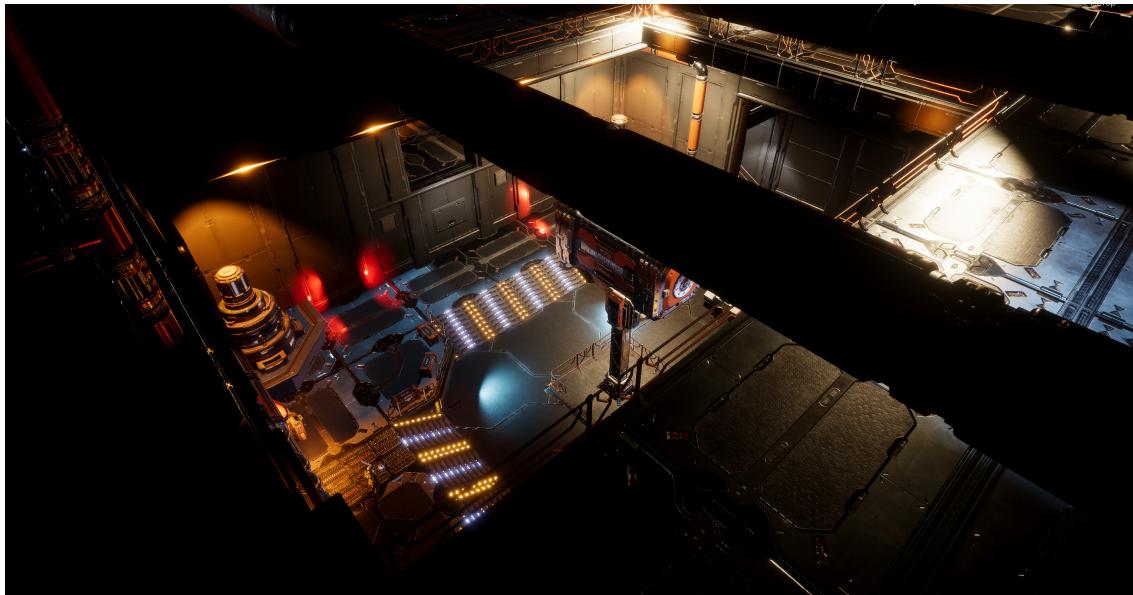


Vue du didacticiel avec shadow maps (ombre dynamique)

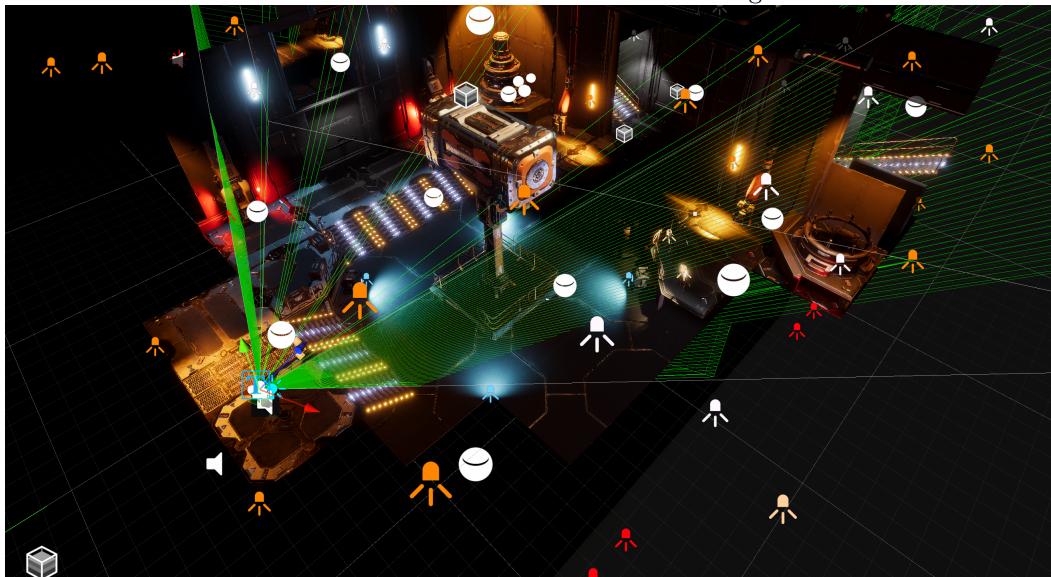
6.3 Occlusion culling

Enfin, nous avons mis en place l'occlusion culling, qui permet de ne pas faire le rendu des objets que la caméra ne peut voir. Cela permet de faire un gain de performance non négligeable.

L'occusion culling se place sur les objets dit "static" (qui ne peuvent se déplacer). Dans notre cas, nous avons appliquer l'occlusion culling sur les murs, sols, différents objets, ... Mais celui-ci n'est pas appliqué aux barils, joueurs, chaises, ...



Vue du didacticiel sans occlusion culling



Vue du didacticiel avec occlusion culling

6.4 Benchmark

Suite à ces étapes d'optimisation, cela a permis de gagner une grande puissance de calcul pour fournir plus d'images par seconde (ips) et donc de permettre une bien meilleure immersion pour les joueurs, notamment dûe à une meilleure fluidité du jeu.

Quelques chiffres sur 2 configurations d'ordinateur en full HD :

PC n°1 :

- Processeur : Intel core i7 1165g7
- Ram : 16Go DDR4
- Carte graphique : Intel Iris

Résultat :

- 35-40 fps constant en qualité low
- 25-30 fps constant en qualité medium

Ordinateur n°2

- Processeur : AMD Ryzen 2600X 5Ghz (overclock)
- Ram : 32Go DDR4 4000Mhz (overclock)
- Carte Graphique : NVidia GTX 1060 6Go (overclock)

Résultat :

- 100-120 fps constant en qualité low
- 70-90 fps constant en qualité medium

Conclusion : On remarque une forte différence de performance notamment dûe à la carte graphique et à sa mémoire vidéo. Lors de nos tests sur l'ordinateur n°2, nous avons pu noter une consommation d'environ 4Go de mémoire ram et 3Go de mémoire vidéo.

7 IA

7.1 Introduction

Notre jeu utilisera une IA qui sera utilisée dans une mécanique a partir de la salle numéro 5.

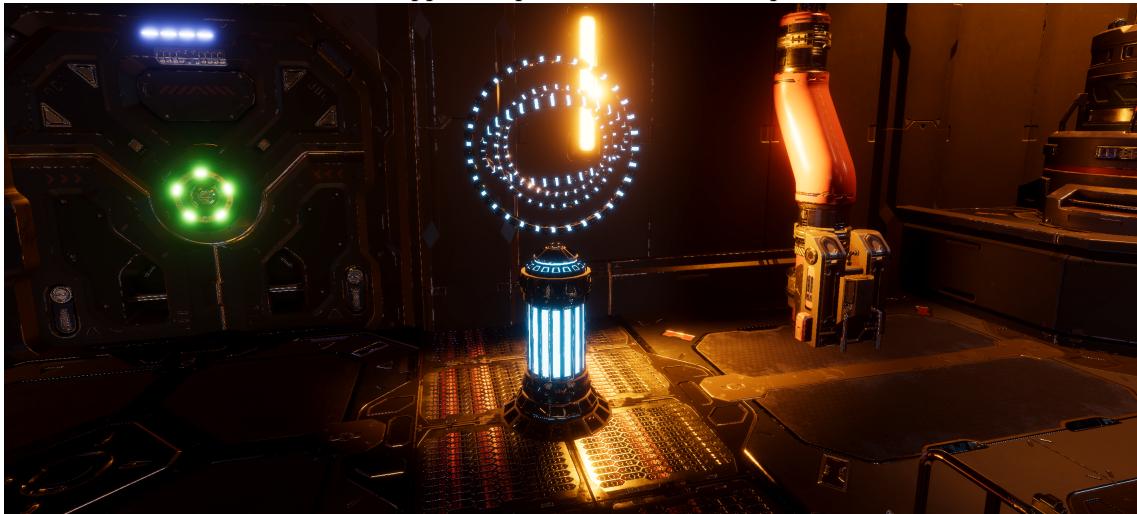
7.2 Nouvelle mécanique de jeu

Cette nouvelle mécanique ajoute un ensemble de nouveaux éléments avec lesquels le joueur pourra interagir.

- Une boule de plasma qui se contrôle comme "le cube", mais qui a comme particularité d'exploser et de tuer tous les joueurs si celle-ci rentre en collision avec les objets, sauf quand elle est tenue par un joueur ou sur les supports de boule plasma et génératrice.
- Un support de boule de plasma qui permet de générer les boules de plasma et aussi de les poser en sécurité.
- Un génératrice qui, quand une boule de plasma est posée dessus, permet d'actionner une porte.
- Unurre qui, quand celui-ci est activé, attire le drone vers lui



Vue du support et par dessus la boule de plasma



Vue du générateur



Vue duurre

7.3 Drone

Le drone sera l'incarnation de l'IA. Globalement sa fonction est de percuter la boule de plasma pour la faire exploser et donc faire échouer les joueurs.

Pour cette tâche nous utiliserons un "Pathfinder". Un "Pathfinder" est une IA qui permet de choisir son propre chemin d'un point A à un point B. Dans notre cas, cette IA devra éviter les objets (collider) et pouvoir voler, comme c'est un drone.

Unity offre de base un "Pathfinder" très performant et simple à mettre en place du nom de "navMesh". Cependant, celui-ci a un gros problème c'est qu'il ne fonctionne que dans un plan en 2 dimensions, dans notre cas il ne pourrait pas voler.

Nous avons trouver un algorithme nommé "A* search algorithm" qui permet de le mettre en place mais cela prend beaucoup plus de temps que prévu. Nous avons donc fait le choix de ne pas l'implémenter pour cette soutenance pour mieux comprendre comment marche l'algorithme et mieux l'implémenter.

Néanmoins, la mécanique de base fonctionne très bien sans l'IA, comme il existe déjà le danger de faire exploser la boule de plasma.



Vue du drone

8 Création de salles

8.1 Hub

Cet endroit est le fil rouge du jeu et pour cela on a voulu faire un endroit qui diffère des salles d'énigmes et pour cela un lieu ensoleillé était le plus approprié et qui pour nous émettait le plus de contraste avec le reste du jeu, ce qui était le but recherché.

Le Hub est une archipel où chaque îles contient deux portails, pour chacun des joueurs, amenant vers la salle voulue. L'accès à ces îles se fait par des ponts en bois ou alors par un parcours en sautant sur des obstacles pour atteindre l'île. Certains accès aux îles permet aussi de montrer des indices sur la salle auquelle les joueurs vont se confronter.

Le centre du Hub sera le lieu principal de l'histoire avec une sorte de fête autour d'un bar et de son barman. Celui-ci offrira des indices pour les salles que les joueurs feront, ce barman pourra aussi intervenir lors des différentes salles si le joueurs a des difficultés.



Vue du Hub

8.2 Salle 3

Cette salle consiste en une succession d'énigmes ou de phases de recherche d'indice dans l'environnement. Une fois l'énigme résolue ou l'indice trouvé, le joueur doit transmettre l'information à l'autre joueur, sous la forme du numéro de la porte qu'il doit traverser pour avancer. Si un joueur choisit la mauvaise porte, il meurt automatiquement.

Ces énigmes peuvent prendre différentes formes. On a par exemple une énigme visuelle où le chemin vers un mur que le joueur doit traverser est indiqué au plafond, avec un chemin lumineux. Une autre énigme est le numéro de la porte mais en code césar, que le joueur doit résoudre. Cela peut également être une simple énigme textuelle, qui mène à l'obtention du numéro de la porte.

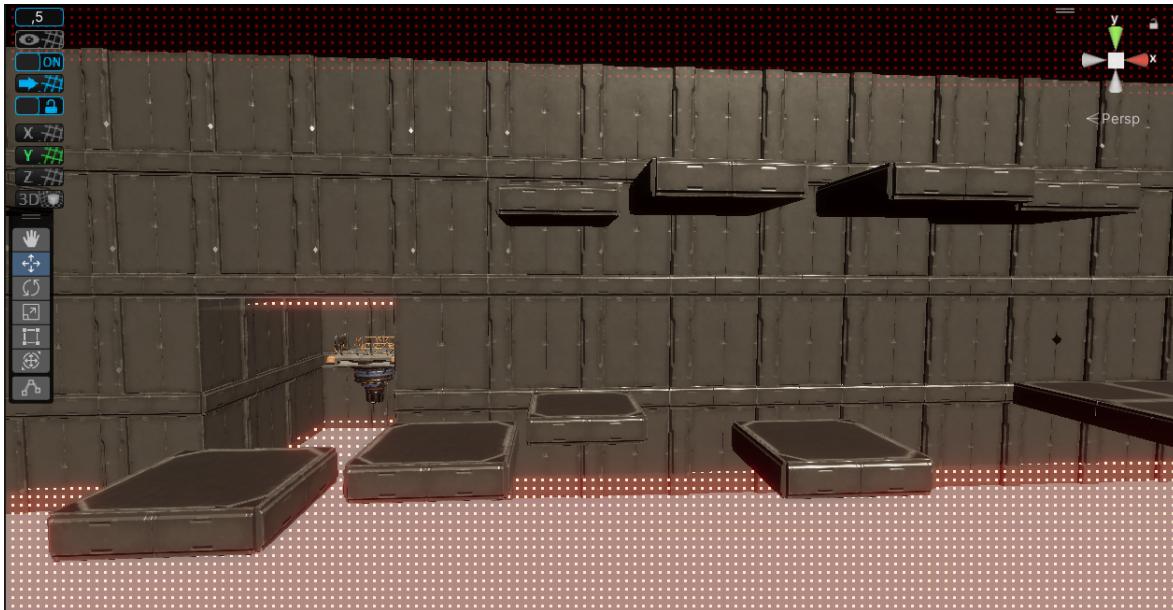
Pour la recherche d'indices dans l'environnement, cela se traduit par la recherche dans le décor (au sol et au plafond) d'un numéro caché dans une texture, un objet ou autre.

On a également ajouté une section à la fin de la salle qui est uniquement un décor à explorer, sans indice ni énigme, mais qui servira à cacher des éléments de l'histoire ou des éléments de décor qui révéleront au joueur des informations sur l'histoire autour du jeu. Pour la salle 3, cette section à explorer est inspirée d'un open-space, avec un espace inspiré d'une salle de réunion. L'exploration de ce genre de salle est optionnelle, et si le joueur ne souhaite pas perdre de temps à la visiter, il peut directement emprunter un portail pour retourner au hub et enchaîner sur la salle suivante.

8.3 Salle 4

Pour cette salle, le principe est de progresser au travers de la salle à l'aide de plateformes qui se déplacent automatiquement (de gauche à droite ou de haut en bas), et d'autres qui sont fixes. La difficulté principale réside dans le fait que si le joueur touche le sol ou le plafond, il meurt directement. Il faut donc que le joueur progresse dans la salle en sautant de plateforme en plateforme et en changeant de gravité sans jamais tomber au sol ou au plafond. Sans oublier le fait que durant

le niveau, chaque joueur devra faire attention à bien actionner les boutons qu'ils trouveront afin d'activer des plateformes essentielles à la progression de l'autre joueur.



Vue d'une partie de la salle 4 avec plateformes fixes et plateformes automatiques

La majorité de la salle va devoir être parcourue de manière horizontale, mais il y a une section verticale, qui nécessite un plus grand sens du timing car cette section est retrécie à la taille du joueur, donc si celui-ci se rate, il meurt automatiquement car cette fois-ci ce sont les murs qui sont mortels. Visuellement, la partie mortelle de la salle (murs ou sols et plafonds) est représentée par une texture "holographique" rouge.

La coopération au sein de cette salle se base sur le fait que chaque joueur devra appuyer sur les boutons qu'ils trouveront sur leur chemin. Au début de la salle, deux chemins s'offrent à eux, et chaque joueur devra partir dans un chemin différent (les deux chemins ne sont pas séparés par un mur, chaque joueur peut donc se voir et s'aider). A d'autres moments, les joueurs devront s'attendre à des paliers où chacun devra appuyer sur un bouton pour ouvrir la porte à l'autre, ainsi même dans les parties où les joueurs sont séparés, ils doivent s'attendre avant de passer à la section suivante.

8.4 Salle 5

Pour cette salle les joueurs devront faire face à un nouvel adversaire, l'IA, ainsi que de nouveaux objets créant de nouvelles mécaniques leur permettant d'avancer :

- La boule de plasma

La boule de plasma sera l'objet principal des nouvelles mécaniques.

- Leurre

Leurre permettra d'occuper l'IA pendant un temps, ce qui permettra aux joueurs de ne pas se soucier de l'IA pendant ce laps de temps.

- Le réceptacle

Cet objet fera le départ, c'est-à-dire là où la boule de plasma sera posée au début de l'éénigme, mais aussi pourra faire la liaison entre un autre receptacle. Par contre dès que la boule de plasma sera enlevée de son receptacle elle ne pourra plus s'y ratacher.

- Le générateur

Il fera la fin de l'éénigme avec la boule de plasma, lorsque la boule de plasma sera mise sur le générateur alors cela déclenchera l'ouverture d'une porte.

Dans cette salle, nous avons deux parties, la première se présente sans l'IA ce qui permet d'introduire les nouveaux objets et faire comprendre aux joueurs les différentes fonctionnalités. Pour la deuxième partie, l'IA fera son apparition pour ralentir la progression des joueurs avec les mêmes objets que la première partie.

A leur arrivée dans la salle les joueurs auront devant eux la boule de plasma, dans cette salle il y a trois portes dont deux qui contiennent un boutons pour ouvrir la troisième porte et ainsi prendre la boule de plasma pour traverser la porte et la déposer sur un générateur dans la salle suivante permettant l'ouverture d'une porte qui fera l'accès à la seconde partie de l'éénigme.

Après la fin de la première partie, les joueurs arriveront dans une nouvelle salle où ils trouveront encore une fois une boule de plasma mais cette fois-ci dès qu'ils la transporteront l'IA les poursuivra pour qu'elle soit lachée et que le niveau recommence. C'est pour cela qu'un des joueurs devra atteindre unurre caché soit par un passage secret mis en évidence par une lumière rouge ou soit en traversant toutes la deuxième partie et le trouver. Une fois leurre activé le second devra parcourir et traverser la salle jusqu'à déposer la boule sur le générateur pour activer la porte de fin le temps duurre.



Vue du dessus de la salle 5

La réelle difficulté de cette salle c'est de percevoir les différentes variable entre le temps du leurre la position éventuelle des joueurs comme celle de l'IA mais aussi de faire une salle multijoueur en y associant les nouveaux objets, ainsi les salles et les énigmes deviennent de plus en plus complexes à imaginer.

9 Modèles 3D

9.1 Introduction

La mécanique de la salle 5 repose sur le transport d'un objet dangereux qui au moindre choc explose et oblige les joueurs à tout recommencer. Pour cela, il fallait des objets uniques au bon fonctionnement du niveau. Tout en restant fidèle au pack d'asset, il fallait que les objets qui devaient être faits main soient futuristes et s'harmonise bien avec l'environnement.

9.2 Boule de Plasma

La boule de plasma est comme son nom l'indique, une boule, qui est à la fois un l'objet qui permet de finir le niveau et en même temps une dangereuse bombe instable. Pour cela, il fallait faire objet qui contient une grande quantité d'énergie pour activer un générateur, mais qui reste en même temps instable. En partant de cette idée, la boule de plasma est constituée d'une couche extérieure en verre servant à contenir le "plasma" mais qui laisse sous-entendre qu'au moindre choc

ce dernier peut se briser. Pour ce qui est de l'intérieur, le plasma a été fait directement dans unity à l'aide de Shader Graph qui permette de modifier, de styliser et d'animer la texture en temps réel.

9.3 Générateur

Le générateur est l'objet que l'on doit déposer la boule de plasma pour ouvrir la porte de fin de niveau. Cet objet doit donc pouvoir contenir une sphère, pouvoir utiliser l'énergie qu'elle renferme et surtout, la maintenir pour éviter que cette dernière explose au moindre choc. Le générateur est donc constitué d'une partie fixe au sol et de 4 anneaux l'évitant et tournant tout autour de la boule de plasma qui devra être placé en leur centre.

9.4 Leurre

Le leurre sert à attirer le robot qui poursuit la sphère, de cette façon, il a fallu faire un objet qui est capable de produire une énergie suffisamment puissante pour attirer le robot. Il fallait de plus que cet objet reste fixe. Ainsi, le leurre est une grosse sphère qui produit de l'énergie positionnée au-dessus d'un pilier.

9.5 Support

Le support est juste un objet où apparaît la boule de plasma au début du niveau et à chaque fois qu'elle se fait détruire, mais peut aussi servir à poser la sphère si le joueur veut s'en libérer pour activer un bouton ou autre sans que la boule de plasma se détruise.

10 Conclusion

10.1 Avance sur le planning

La seule avance que l'on pourrait percevoir, c'est la décoration et la mise en place de la lumière dans les salles 4 et 5 qui n'étaient pas attendues.

10.2 Retard sur le planning

Nos seuls retards sur le planning se situent sur l'IA qui suite à des imprévus n'a pu être implémenté pour cette soutenance. Nous avons aussi un léger retard sur la construction des salles avec encore la salle 6 à faire. Nous avons préféré garder du temps pour bien réfléchir et finir en beauté.