

Rapport de Projet

GolluM's Lair

**Nazim LAMECHE, Axel THAUVIN,
Hugo SPYROPOULOS**



Rapport de projet

Sommaire

1	Introduction	3
2	Conception du projet	4
2.1	Répartition du travail	4
2.2	Structure du projet	5
2.2.1	Histoire	5
2.2.2	Gameplay	6
2.3	Map	7
2.4	Quêtes	9
2.5	Téléphone	11
2.6	Personnages	13
2.6.1	Gameplay des personnages	13
2.6.2	Hitbox des personnages	14
2.7	Endurance	15
2.8	Objets	16
2.8.1	Lampe Torche	18
2.8.2	Le Radar	19
2.8.3	Le Piège	20
2.8.4	La canette de Redbull	21
2.9	Inventaire	22
2.10	Développement I.A	23
2.10.1	Modèle de GolluM	24
2.10.2	Comportement de l'IA	25
2.10.3	Principe d'absorption	26
2.11	HUD / Menus	28
2.11.1	Sélection des personnages	31
2.11.2	Salon	32
2.11.3	Description des personnages	33
2.11.4	Menus de Victoire / Défaite	35
2.11.5	Menus de statistiques	36
2.12	Conception des personnages	37
2.13	Animations	38
2.13.1	Outil d'animation de Unity	39

Rapport de projet

2.14 Site Web	40
2.15 Réseau	41
2.15.1 Fonctionnement global du réseau	41
2.15.2 Connecter le client avec le serveur	42
2.15.3 Fonctionnement du serveur	43
2.15.4 Lobby	44
2.15.5 Relay	44
3 Avancement du Projet	45
3.1 Avancement global / Comparaison avec le cahier des charges	45
3.1.1 Comparaison entre les soutenances	46
3.2 Personnages	46
3.3 Animations	46
3.4 Téléphone	46
3.5 Quêtes	47
3.6 Intelligence Artificielle	47
3.7 Réseau	47
3.8 HUD / Menus	47
3.9 Character Designs	48
3.10 Téléphone	48
3.11 Carte	48
3.12 Objets	48
3.13 Graphismes/DA	48
4 Notre avis sur le projet	49
5 Tableau d'avancement du projet pour le rendu final	49
6 Conclusion	49
7 Ressources	51

Rapport de projet

1 Introduction

Notre projet, intitulé *Gollum's Lair*, est un jeu d'horreur mêlant exploration et tension, conçu pour être joué en solo ou en mode multijoueur. L'action se déroule au sein de l'EPITA, notre école d'ingénieur en informatique. Le choix du nom *Gollum's Lair* est inspiré du surnom de notre professeur en informatique "GolluM", imposant une atmosphère de crainte et de peur parmi les élèves.

Dans ce jeu, les joueurs se déplacent à travers une reconstruction en 3D du bâtiment A de l'EPITA et doivent accomplir différents objectifs pour échapper aux griffes terrifiantes de GolluM. En fonction de leur classe de personnage, les joueurs bénéficient d'un gameplay unique et peuvent terminer le jeu de différentes manières.

Ce document constitue le rapport final de notre projet et présente l'état actuel de notre travail dans son ensemble.

Rapport de projet

2 Conception du projet

2.1 Répartition du travail

Les différentes parties du jeu à programmer ont été réparties de façon équitable entre les différents membres du groupe. Pour la partie création, conceptions de modèles 3D et conception des menus, nous nous sommes également répartis équitablement les tâches afin que chaque membre du groupe approuve et apporte sa patte artistique. Ces parties graphiques sont non négligeables puisque l'idée principale de notre jeu vidéo repose sur le fait de jouer dans le campus de l'EPITA, modélisé en 3D. Nazim et Axel ont aidé Hugo dans la réalisation de certaines tâches 3D, afin que chaque membre du groupe découvre cette partie fondamentale dans la création d'un jeu vidéo. Nous avons fait en sorte que chacun participe aux différentes phases du projet, afin de découvrir les rouages complets de la création d'un jeu vidéo.

	NAZIM	AXEL	HUGO
Développement I.A.	A	R	
HUD/Menus		A	R
Graphismes/DA		A	R
Animation	A		R
Multijoueur		R	A
Sound/Character design	R		A
Site Web	R	A	
Développement Général	A	R	
Classes des personnages	A	R	

Légende : A = Assistant / R = Responsable

Rapport de projet

2.2 Structure du projet

2.2.1 Histoire

L'histoire a été pensée et écrite, de telle sorte qu'elle soit unique tout en abordant également quelques thématiques amusantes afin de donner aux joueurs la possibilité de s'identifier dans le gameplay. C'est, en effet, pour cela que nous avons choisi l'EPITA et quelques références qui feront sourire certains épitéens.

Un groupe d'amis (suivant le nombre de joueurs qui jouent) se retrouve sur le campus.

Ils reçoivent une notification de leur ami Nazim qui leur demande s'ils peuvent aller l'i3lock en salle machine, car il est parti du campus en oubliant de verrouiller sa session. Une fois rentrés dans le bâtiment, en salle 311, ils remarquent que sa session est effectivement encore ouverte, mais qu'il a laissé des dizaines de gobelets de café en carton et des canettes de Red Bull partout autour de son bureau.

À ce moment précis, une notification de GolluM sur le discord 2027 dit : “Je viens de voir en salle machine 311 le désordre laissé par un élève, j'arrive...”.

L'objectif du groupe d'amis est alors de réussir à s'exfiltrer au plus vite du campus et de prévenir les autorités.

Mais évidemment, GolluM ne les laissera pas faire...

Rapport de projet

2.2.2 Gameplay

Le but de notre jeu est d'explorer, de nuit, le bâtiment A de l'EPITA en atteignant des objectifs pour pouvoir en sortir. Ces objectifs pourront être de différentes sortes :

- Ramasser des items ayant chacun des propriétés uniques
- Exécuter des actions spécifiques (de type QTE ou autres)
- Combiner divers objets entre eux afin de progresser dans le jeu.

Ces différents objectifs sont générés de manière aléatoire afin de procurer au joueur une sensation de renouveau à chaque partie. Le joueur est alors dans une quête perpétuelle pour réaliser ces objectifs, tout en évitant les différentes attaques de GolluM. Les mécaniques implémentées permettent de rendre la présence de GolluM opprassante, rendant ainsi le joueur limité dans ses mouvements.

Les joueurs doivent donc faire preuve d'ingéniosité pour réaliser leurs objectifs tout en survivant à GolluM. Ils disposent également de différents items afin de ralentir GolluM : des pièges à placer, des portes à fermer à clefs ou encore des objets lancables permettant d'attirer GolluM grâce au son émis par ces derniers.

Rapport de projet

2.3 Map

Le gameplay se déroulant au sein du bâtiment A de l'EPITA, celui-ci a été entièrement modélisé à l'aide de l'application Polycam et de Blender. Pour la version définitive du jeu, Hugo s'est aussi occupé de rajouter l'extérieur de l'EPITA (i.e arbre, plantes, béton, sol, escaliers extérieurs, etc.) ainsi que l'extérieur visible du bâtiment C.



Figure 1: Map complète vue du dessus

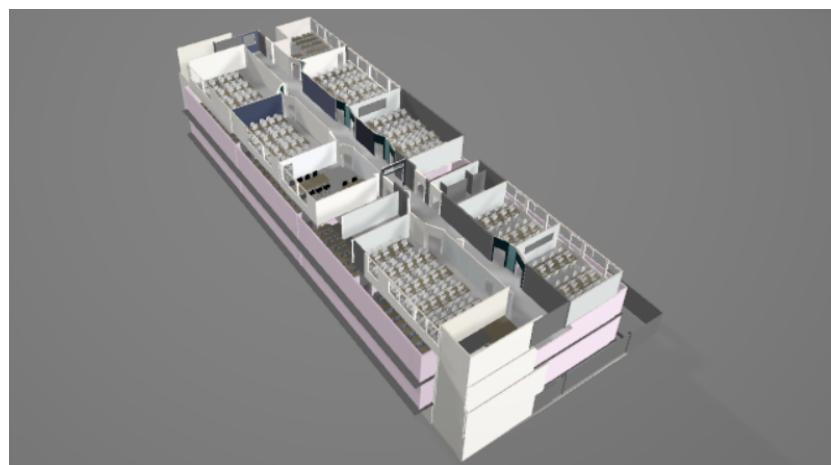


Figure 2: Bâtiment A

Rapport de projet



Figure 3: Salle machine



Figure 4: Modélisation des fameuses poubelles de l'EPITA

Rapport de projet

2.4 Quêtes

Pour que la partie prenne fin, il faut compléter l'ensemble des quêtes, que nous avons séparé en 5 catégories :

- **Les quêtes de Zone** : quêtes qui demandent au joueur d'aller à un endroit spécifique du bâtiment (e.g., aller en salle machine pour verrouiller la session de Nazim).

- **Les quêtes d'interaction** : interagir avec un objet qui se trouve dans la map (e.g., mettre les batteries dans le générateur d'électricité)

- **Les quêtes de récupération** : quêtes qui consistent à récupérer un ou plusieurs objets (e.g., récupérer 3 batteries ; récupérer la clé du bureau de GolluM).

- **La quête du CRI**. quête qui consiste à réparer l'intranet de l'EPITA (le CRI tombe en panne au cours du jeu, et il faudra alors une énigme pour le réactiver)

- **La quête d'attente** : quête qui s'active à la fin du jeu. Il faut que le joueur attendre 1 min et 30 secondes avant l'arrivée de la police, et qu'il puisse s'échapper du bâtiment.



Figure 5: Quête de récupération

Rapport de projet



Figure 6: Quête du CRI

Rapport de projet

2.5 Téléphone

Pour réaliser notre jeu, nous nous sommes inspirés de plusieurs autres jeux vidéos (e.g., slasher) et avons mis en place un système de communication entre les joueurs. Pour cela, nous avons choisi d'implémenter un téléphone, disposé en bas à droite de l'écran, permettant au joueur d'y accéder facilement et de pouvoir y écrire des messages textuels.



Figure 7: Téléphone avec des messages reçus

Rapport de projet



Figure 8: Applications téléphone

Nous avons intégré au téléphone 2 applications :

- **Une messagerie générale/ chat général**, permettant aux joueurs de communiquer entre eux par message.
- **Une messagerie vocale / chat vocal** : notre meilleure fonctionnalité. Les joueurs peuvent parler entre eux grâce à un chat vocal positionnel, qui prend en compte les distances et la position des joueurs. De plus, GolluM est en mesure d'entendre les joueurs quand ils parlent.

Rapport de projet

2.6 Personnages

Nous avons intégré plusieurs personnages jouables en solo et en multijoueur. Chaque personnage possède des caractéristiques spéciales permettant de proposer des gameplay différents selon le choix du joueur. Concernant les designs des personnages, nous avons essayé de les rendre les plus réalistes possibles afin de correspondre le plus fidèlement possible à leur description.

2.6.1 Gameplay des personnages

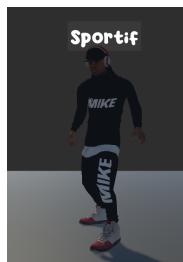
Quatre classes sont jouables et une classe peut être sélectionnable par joueur. Les classes sont :

- **Le Sportif** : possède une endurance plus élevée et une canette de RebBull, lui permettant de courir plus longtemps que les autres personnages.

- **Le Geek** : possède la capacité de pouvoir connaître l'étage dans lequel GolluM est position grâce au radar dont il est muni

- **Le Joker** : Possède un piège, lui permettant de bloquer les déplacements GolluM pendant 10 secondes, ainsi que ceux des joueurs si ces derniers ne font pas attention.

- **La Lockpicker** : possède un trousseau de clefs qu'elle pourra utiliser sur un nombre limité de portes, afin de limiter l'accès à une salle.



Sportif



Joker



Geek



Lockpicker

Figure 9: Personnages

Rapport de projet

2.6.2 Hitbox des personnages

Pour chaque modèle 3D de personnage, nous avons mis au point une "Hitbox" spécifique. Cette Hitbox permet d'avoir une détection des collisions précise, avec des zones de collision sur chaque partie du corps.

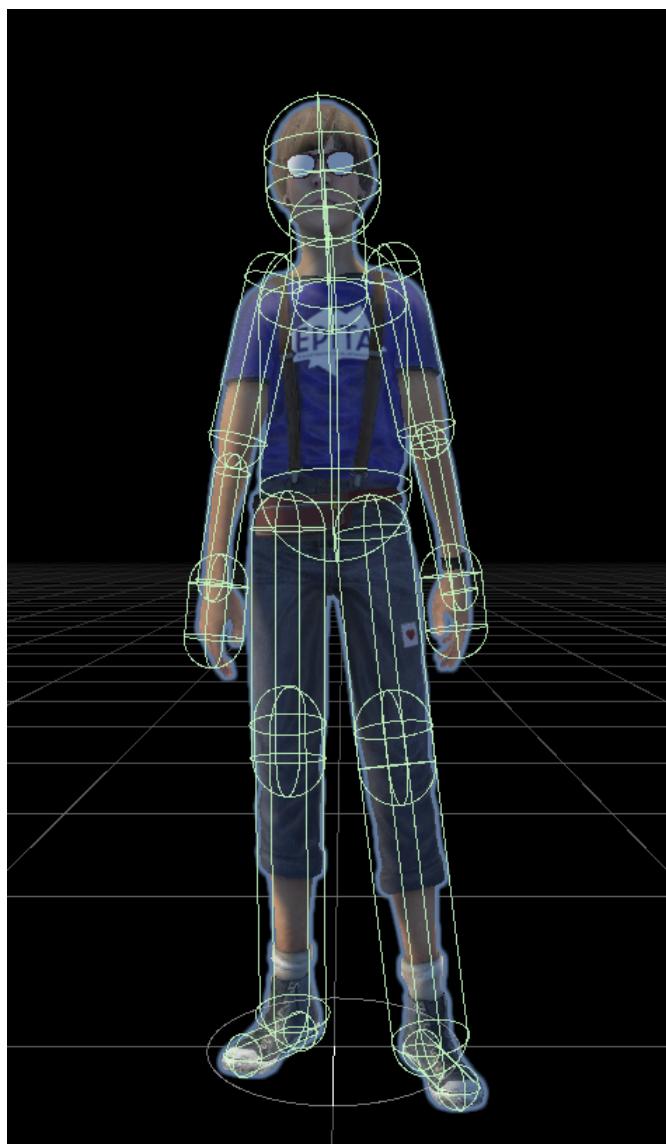


Figure 10: Hitbox

Rapport de projet

2.7 Endurance

Afin d'ajouter de la difficulté dans le jeu, nous avons implémenté de l'endurance pour tous les personnages. Celle-ci se recharge au bout d'un certain temps. Le niveau d'endurance est visible grâce à une zone de progression en bas de l'écran



Figure 11: Endurance

Rapport de projet

2.8 Objets

Plusieurs objets utilisables dans le jeu ont été intégrés et répartis en deux catégories : (i) les objets **communs**, trouvables sur la carte comme les gobelets, les gourdes, etc. et les objets **spécifiques aux joueurs** comme le **radar** pour le Geek, les **clés passe-partout** de la Lockpicker, le **piège à loup** du Joker et la **canette de RebBull** pour le Sportif ; (ii) les **objets de quêtes** à utiliser lors de la partie.



Canette de RebBull



Lampe Torche



Radar



Piège



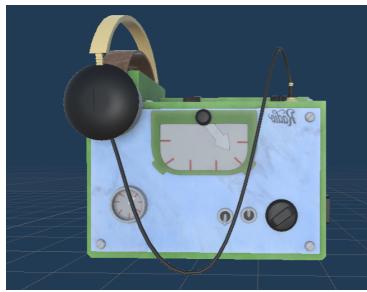
Gourde

Figure 12: Objets utilisables

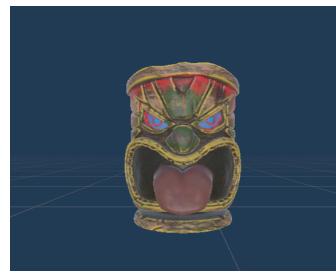
Rapport de projet



Pc de GolluM



Radio



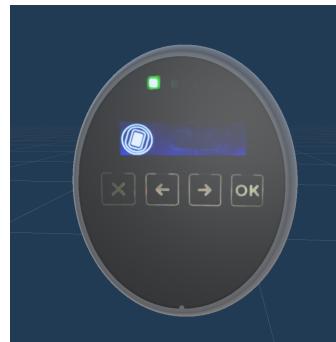
Totem



Sac à dos



Générateur



Lecteur de carte

Figure 13: Objets de quêtes

Rapport de projet

2.8.1 Lampe Torche

Le visuel et le comportement de lampe torche a été modélisé pour être le plus réaliste possible. Pour cela, nous avons intégré une batterie rechargeable. Ainsi, lorsque déchargée, il suffit d'appuyer sur un bouton plusieurs fois d'affilée afin de la recharger à la manière d'une dynamo (attention à la tendinite !), avec la contrainte que le son émis puisse être entendu par GolluM.

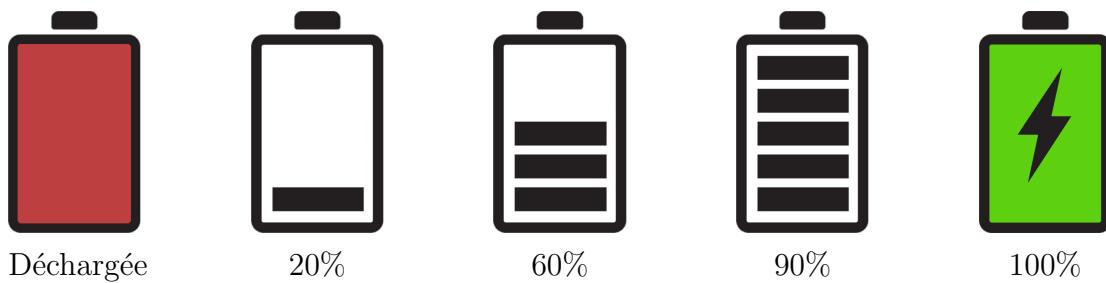


Figure 14: Présentation des niveaux de charge de la batterie de la lampe torche

Rapport de projet

2.8.2 Le Radar

Le **radar** est un objet propre au geek. Il permet de localiser l'étage dans lequel se trouve GolluM dans le bâtiment A en affichant la position de ce dernier par un point rouge sur l'écran du Radar.



Le radar ne détecte rien



Le radar détecte la position de GolluM

Figure 15: État Radar

Rapport de projet

2.8.3 Le Piège

Le **piège** est un objet propre au Joker, de type piège à ours, il permet de bloquer toute entité vivante mettant le pied dedans. Destiné à GolluM, il peut par mégarde piéger d'autres joueurs

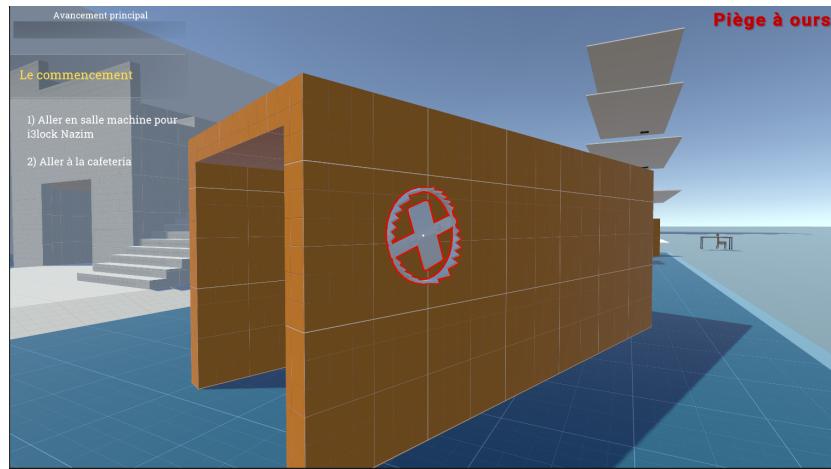


Figure 16: aperçu du placement du piège sur le mur



Figure 17: aperçu du placement du piège sur le sol

Rapport de projet

2.8.4 La canette de Redbull

La canette de **Red Bull** est un objet propre au sportif, lorsque celui-ci est fatigué et n'a plus d'endurance, la cannette de RebBull restaurera l'équivalent de la moitié de sa barre d'endurance.



Figure 18: Canette de Redbull

Rapport de projet

2.9 Inventaire

Pour pouvoir stocker tous les objets, chaque joueur dispose d'un inventaire, chacun disposant de 3 place de stockage disponible. En parallèle de celui-ci, mais invisible pour les joueurs cette fois, nous avons rajouté un inventaire de quêtes . Cet inventaire est utilisé lorsque certaines quêtes nécessitent la récupération d'objet, et est commun à tous les joueurs de la partie.



Figure 19: Inventaire en Jeu

2.10 Développement I.A

L'intelligence artificielle englobe un ensemble de théories et de techniques permettant de constituer une technologie autonome. Dans notre jeu, l'IA correspond au comportement de GolluM (le monstre). Ce dernier peut réagir aux bruits et aux mouvements des alentours dans un certain rayon que nous avons nous-mêmes défini. Tant que l'IA détecte le joueur, celui-ci va définir le meilleur chemin possible afin d'accéder le plus rapidement à ce dernier. L'idée derrière cette IA est de procurer au(x) joueur·s de la peur, puisque les déplacements de GolluM seront liés à leur·s capacité à ne pas se faire détecter. (Lors du développement du jeu, il nous est déjà arrivé d'avoir des sursauts de frayeur, car Gollum nous a surpris lorsque nous codions les personnages)

Rapport de projet

2.10.1 Modèle de GolluM

Nous avons tout d'abord trouvé le modèle du monstre. Nous souhaitions avoir un ennemi qui fait peur aux joueurs, puis à partir de ça nous avons théorisé le comportement de l'intelligence artificielle. Pour que le projet porte bien son nom, nous avons mis un point d'honneur à récupérer un vrai modèle de Gollum, et sommes fiers de le présenter pour le rendu final du jeu.

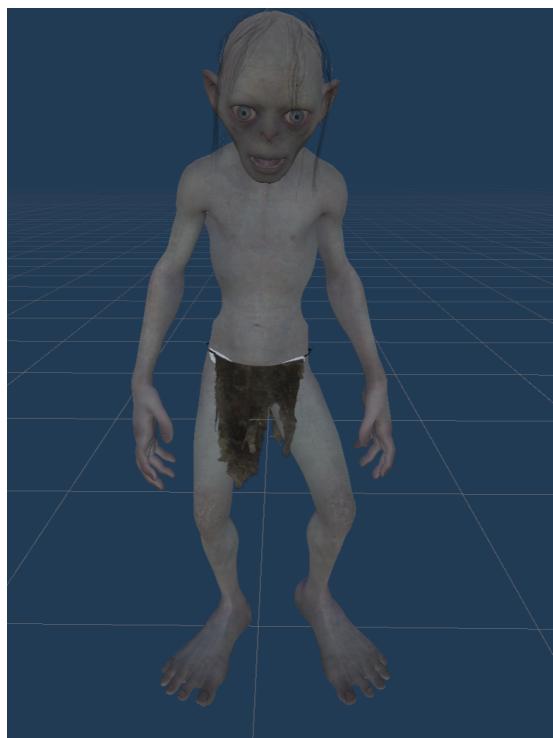


Figure 20: GolluM

Rapport de projet

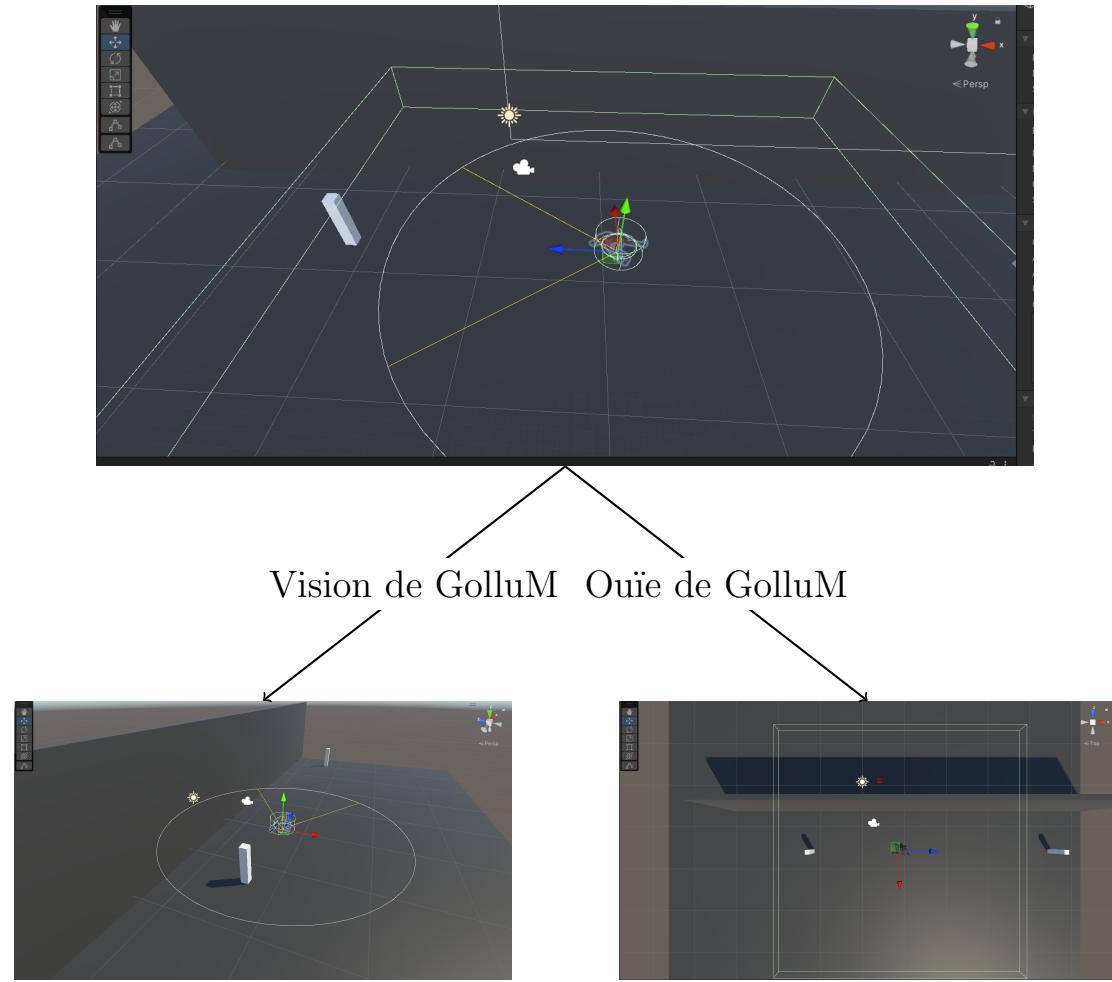
2.10.2 Comportement de l'IA

Pour le comportement de GolluM, nous avons intégré deux comportements différents pour détecter les joueurs :

- **L'ouïe**, sa zone de détection de bruit qui lorsqu'il entend un bruit dans sa zone de détection va définir le meilleur chemin vers l'origine du son. Comme présenté ici dans la figure 3 avec le rouge. Cette zone comprend tout types de bruits (joueurs qui parlent, objets qui font du bruit, et bruit de pas des joueurs).

- **La vue**, sa vision, celle-ci est représentée par le cercle qui a pour origine GolluM. Nous avons séparé cette fonctionnalité en deux parties; la première son regard (zone jaune sur la figure 3 ci-dessous) représentant la direction de son regard, la seconde sa zone de vue totale (cercle blanc sur la figure 3). Le cercle correspond au champ de vision maximal qu'aurait GolluM autour de lui.

Rapport de projet



2.10.3 Principe d'absorption

Évidemment, si le champ d'écoute de GolluM était limité par les murs, le jeu ne serait pas amusant. Pour parer ce problème, nous avons réfléchi à un système : le principe d'absorption du son. Comme dans la réalité, chaque objet de notre jeu a une valeur d'absorption du son qui explique au jeu si l'objet laisse passer le son ou non. Par exemple, un mur en brique absorbe 90% du son, donc GolluM entend le son produit par le joueur que si ce son provient de juste derrière le mur. En revanche un mur en bois qui absorbe 30% du son implique que GolluM peut entendre assez loin derrière le mur.

Rapport de projet

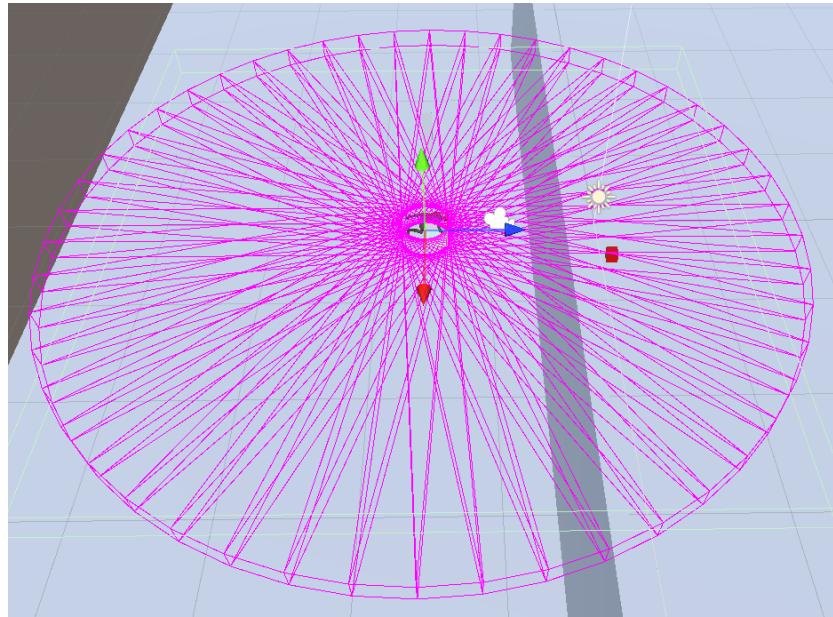


Figure 21: radius absorption

Ainsi, lorsqu'un son est émis, le rayon de détection du son de GolluM est alors modifié selon le facteur d'absorption du mur. En voici un exemple.

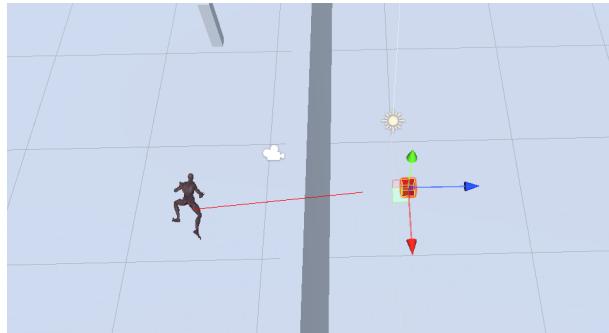


Figure 22: radius absorption

Le son est émis par le cube rouge, et la ligne allant de GolluM vers le cube représente le vecteur direction, tant que le vecteur ne touche pas le cube, GolluM considère qu'il n'a pas entendu le bruit.

Rapport de projet

2.11 HUD / Menus

L'HUD correspond à un ensemble d'informations affichées en périphérie du centre de l'écran et renseignant le joueur sur son personnage ou son environnement. Les menus pour lancer une partie en solo, ou bien en multijoueur, la barre vie, etc.

Ces menus ont été améliorés pour le projet final afin de les mettre en accord avec le thème "horreur" que nous nous sommes imposés.

De ce fait, nous avons tout connecté avec un menu principal, plus propre, et avons ajouté un menu de paramètre en essayant d'y ajouter le plus de fonctionnalités possible.

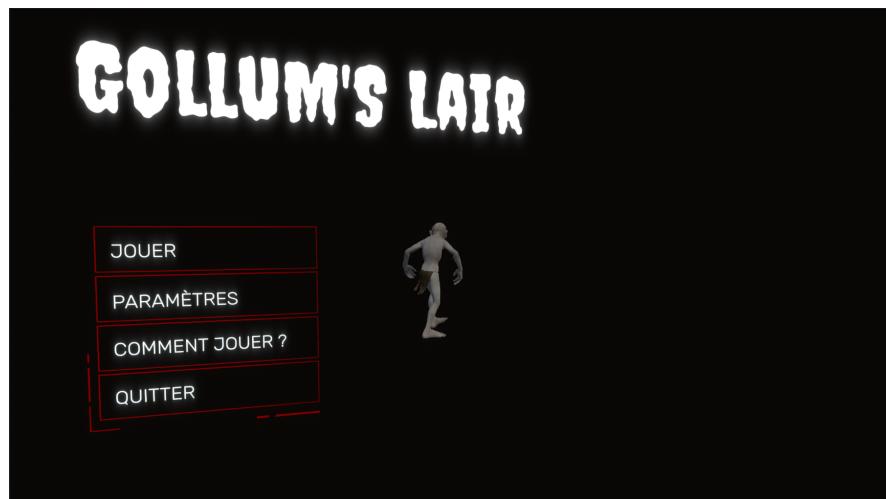


Figure 23: Menu principal

Nous avons intégré un menu principal afin d'organiser l'ensemble des autres menus. Cet ajout permet de mieux ordonner les menus et sont répartis en quatre sections : Jouer , "Paramètres" , Comment jouer et Quitter .

Rapport de projet



Figure 24: Menu des touches

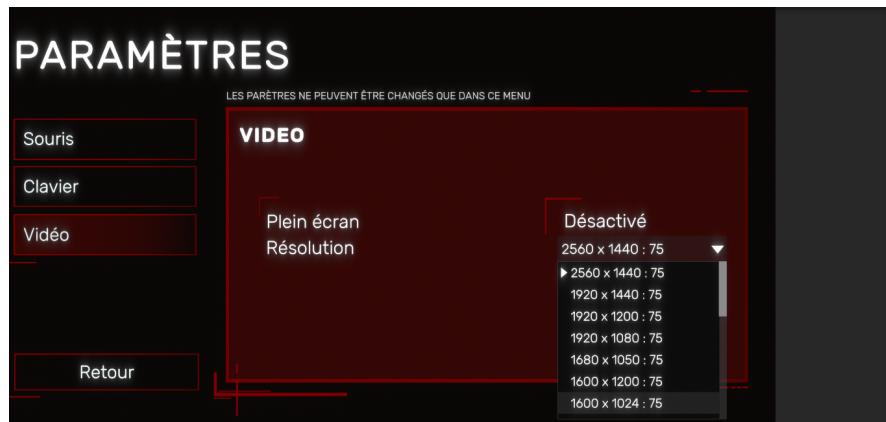


Figure 25: Menu des résolutions

Rapport de projet



Figure 26: Menu du Lobby

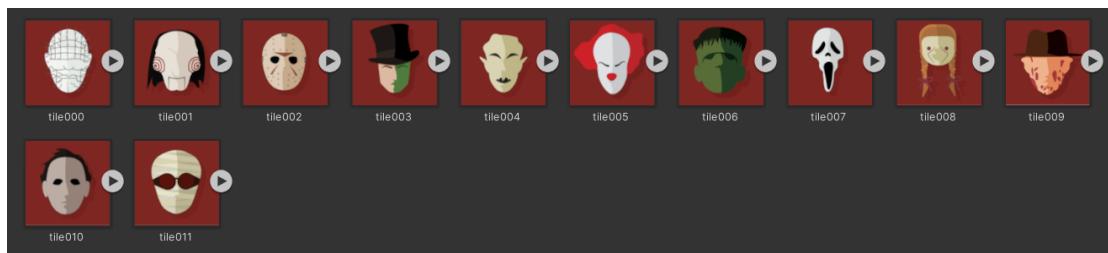


Figure 27: Icônes des joueurs

Nous avons implémenté un système d'avatars (faisant référence à des personnages horrifiques) pour les joueurs, afin de rendre le salon de jeu (lobby) plus attrayant et plus agréable à regarder.

Rapport de projet

2.11.1 Sélection des personnages

Pour le menu de sélection des personnages, nous nous sommes inspirés de grands noms comme Overwatch et Valorant. Ces menus permettent la sélection des personnages, incluant le logo du personnage sélectionné, ses caractéristiques, mais également la sélection que les autres joueurs ont faite en réseau.

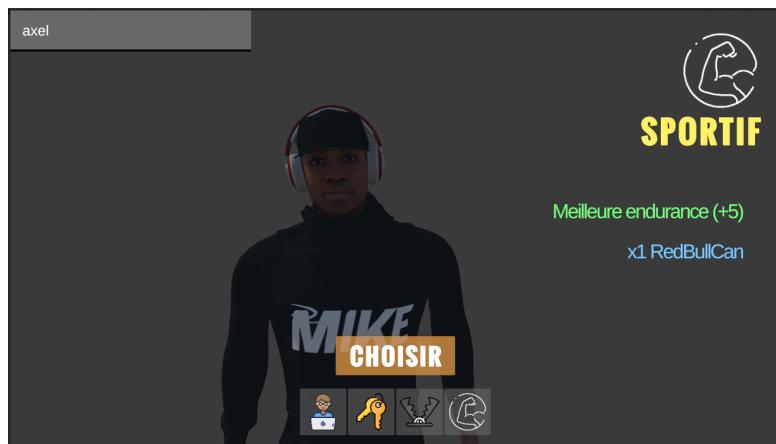


Figure 28: Menu sélection des personnages

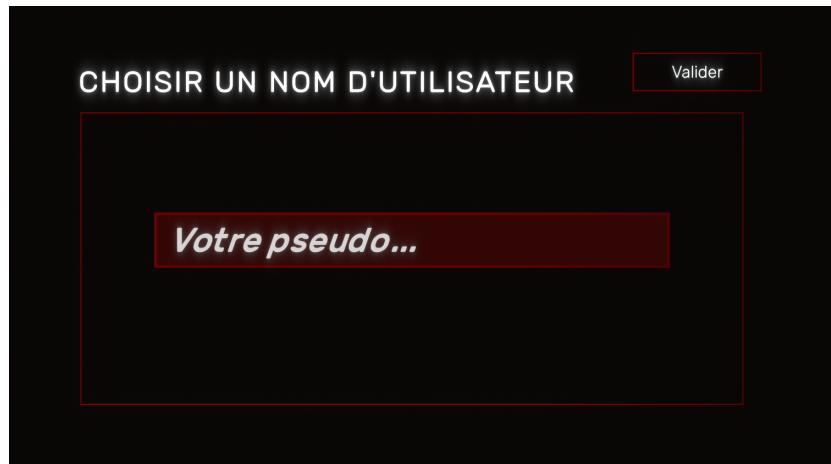


Figure 29: Menu choix du pseudo

Rapport de projet

2.11.2 Salon

Notre projet étant en réseau, nous avons mis en place un système de salon, s'actualisant automatiquement au cours du jeu en fonction des personnes qui y rentrent. Ainsi, il est possible de créer un salon de jeu jusqu'à quatre joueurs.



Figure 30: Menu des salons

Rapport de projet

2.11.3 Description des personnages

Nous avons ajouté de nombreuses descriptions, sous forme de fiche, pour chaque personnage du jeu tel que :



Figure 31: Description du Geek

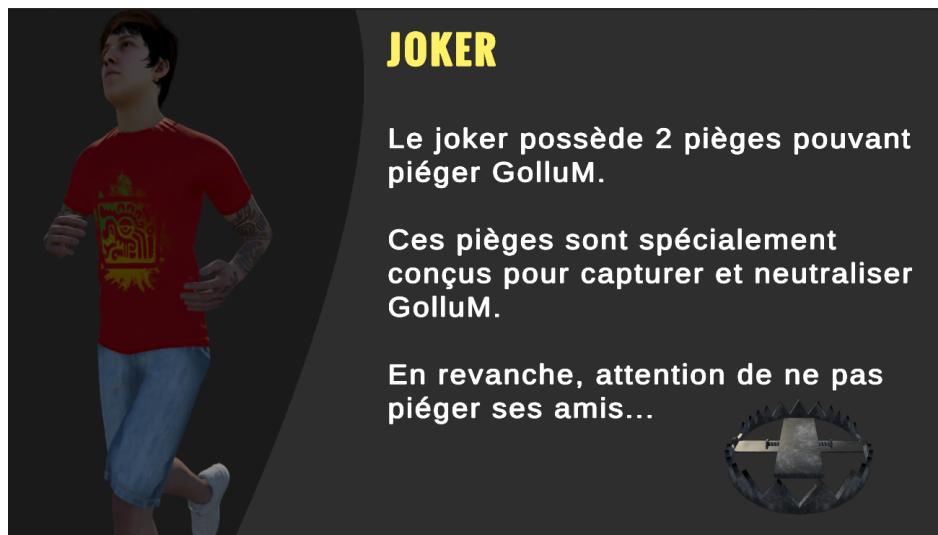


Figure 32: Description du Joker

Rapport de projet



Figure 33: Description de la Lockpicker



Figure 34: Description du Sportif

Rapport de projet

2.11.4 Menus de Victoire / Défaite

La partie se termine de deux façons : la première "Victoire", lorsque les joueurs ont achevé la mission finale qui est d'appeler la police et de sortir de la rejoindre sur le terrain du campus ; la seconde "Défaite" lorsque GolluM tue tous les joueurs.

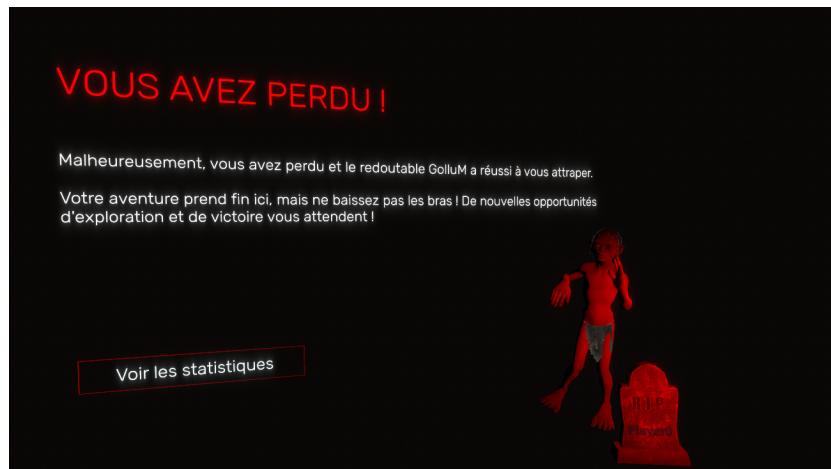


Figure 35: Menu de défaite



Figure 36: Menu de victoire

Rapport de projet

2.11.5 Menus de statistiques

Pour ajouter de la compétitivité dans le gameplay, nous avons créer un menu de statistique qui s'affiche à la fin de la partie, celui-ci effectue un classement selon, le nombre de quête complété, le temps passé, l'état des joueurs (mort / vivant).



Figure 37: Statistiques

Rapport de projet

2.12 Conception des personnages

Les personnages ont un design assez simple et réaliste, on a voulu respecter un maximum la direction artistique et les descriptions pour chaque personnage que nous nous sommes imposés afin d'avoir une identité propre à notre jeu. De plus, nous avons ajouté quelques références avec EPITA.



Figure 38: Conception des personnages

Rapport de projet

2.13 Animations

Les animations ont été prises en majorité sur internet, nous avons fait une sélection d'animations réalistes. Tel que, s'accroupir, sauter, marcher, courir, marcher à gauche/droite, marcher en étant accroupi, etc.



Accroupi Attente

Figure 39: Animations : position statique



Saut Marche croupi Courir

Figure 40: Animations - Mouvement

Rapport de projet

2.13.1 Outil d'animation de Unity

Unity possède un outil très puissant concernant les animations appelées Animator. Pour ce faire, on utilise un principe de Schéma. Nous avons tout d'abord un état de base appelé "idle" correspondant à état d'attente du personnage, lorsqu'il ne fait rien. Puis, lorsqu'on active une touche correspondant à une action, celle-ci va enclencher une transition de "idle" vers l'animation de l'action. Cela a pour effet d'améliorer le réalisme des actions.

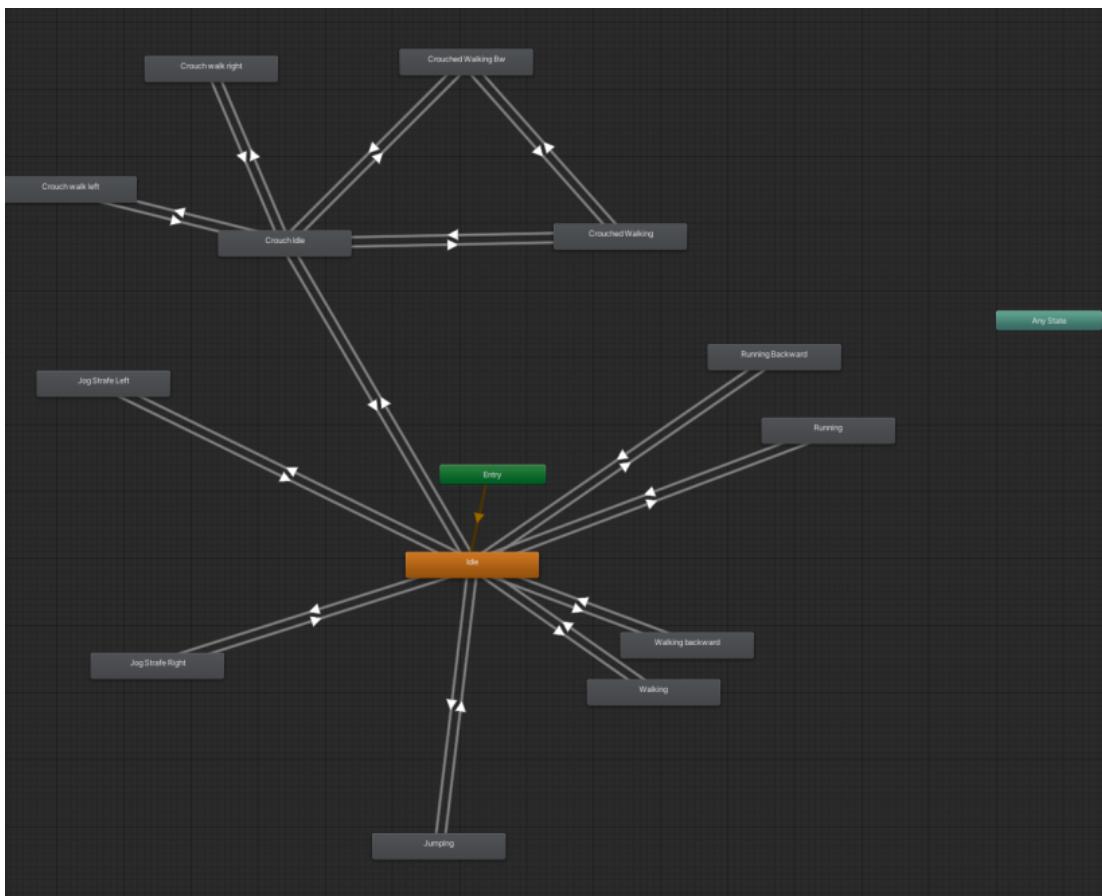


Figure 41: Schéma d'animation

Rapport de projet

2.14 Site Web

Un site web a été créé pour notre groupe afin de regrouper toutes les informations requises. Celui-ci est composé de :

- **L'accueil/index**, représentant la page d'accueil ainsi que la page de téléchargement.
- **Documents**, contenant l'ensemble de nos documents (rapports de soutenances, plan de soutenance, cahier des charges, etc.).
- **Chronologie**, retracant tous les ajouts et les théorisations que nous avons faites depuis la formation du groupe.
- **Logiciels**, énumérant tous les logiciels utilisés pour notre projet.
- **Contact**, Notre adresse mail, numéro de téléphone de chacun.

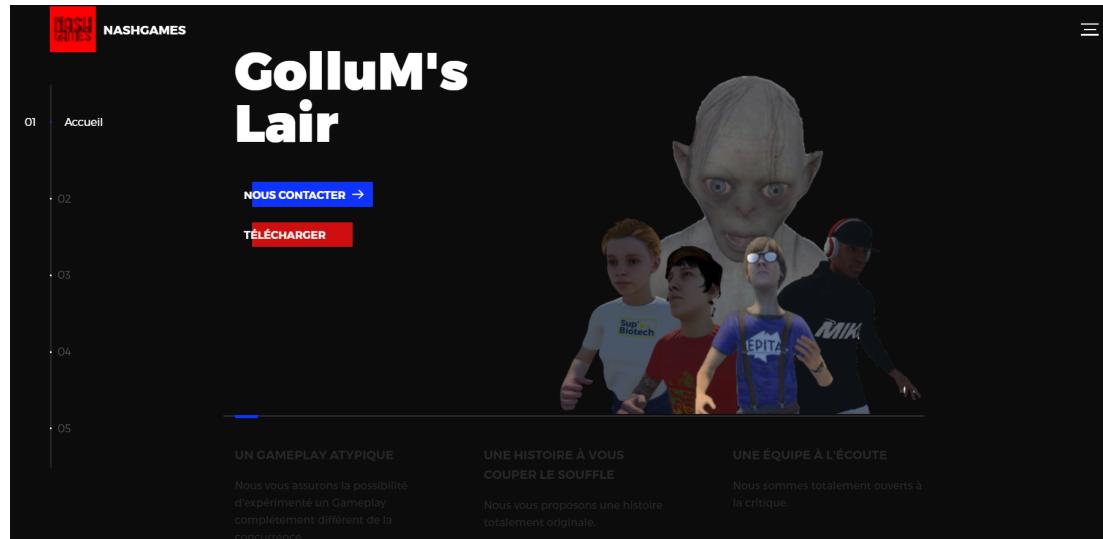


Figure 42: Accueil du site

Rapport de projet

2.15 Réseau

2.15.1 Fonctionnement global du réseau

Le réseau est une partie importante de notre projet puisque c'est cette partie qui s'occupe de relier tous les joueurs entre eux. Pour une explication plus concise, nous découperons l'explication du réseau en plusieurs parties, chacune étant un composant majeur de notre implémentation du réseau. Dans un premier temps, voici un schéma qui résume toute l'infrastructure, que nous expliquerons dans les parties suivantes.

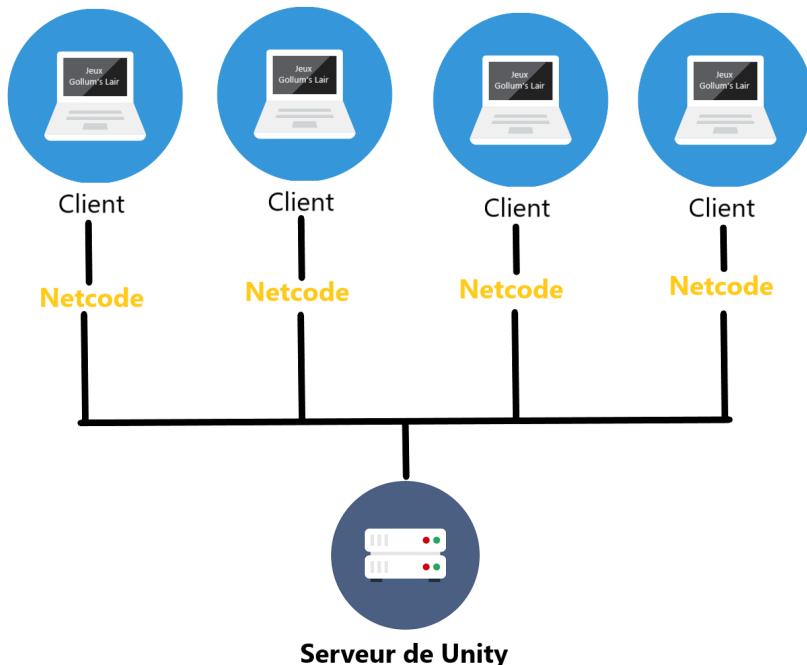


Figure 43: Schéma de l'infrastructure du réseau de notre jeu

Rapport de projet

2.15.2 Connecter le client avec le serveur

Nous utilisons la bibliothèque nommée "Netcode for GameObject" pour récupérer les informations du serveur. Cette bibliothèque s'occupe, en lui fournissant, l'adresse IP du serveur, de récupérer tous les joueurs et leurs positions pour chaque client. Une fois cela fait, Unity s'occupe de placer pour chaque client tous les joueurs avec les différents composants (classes, stamina, objet) que nous communiquons en réseau. Nous avons certaines difficultés ce processus en place. En effet, le service de Unity "Netcode for GameObject" interdit par défaut la modification quelconque d'objets en réseau par le client (par exemple, le client ne peut pas déplacer son personnage en réseau, car le serveur ne lui a pas donné la permission de déplacer des objets).

Rapport de projet

2.15.3 Fonctionnement du serveur

Afin de communiquer avec les joueurs, le serveur dispose de différents dispositifs. En effet, Netcode a besoin de savoir où trouver les joueurs et leurs informations (pseudo, classes), mais de savoir, mais également est encore dans la partie ou non. Le serveur étant fourni grâce à Unity, les services cités ci-dessous sont entièrement pris en charge par le serveur. Nous n'avons donc pas besoin de nous occuper du "backend" du serveur, ce qui nous enlève une bonne épine du pied. Ainsi, voilà le schéma représentant notre implémentation du serveur, permettant de faire communiquer les joueurs ensemble.

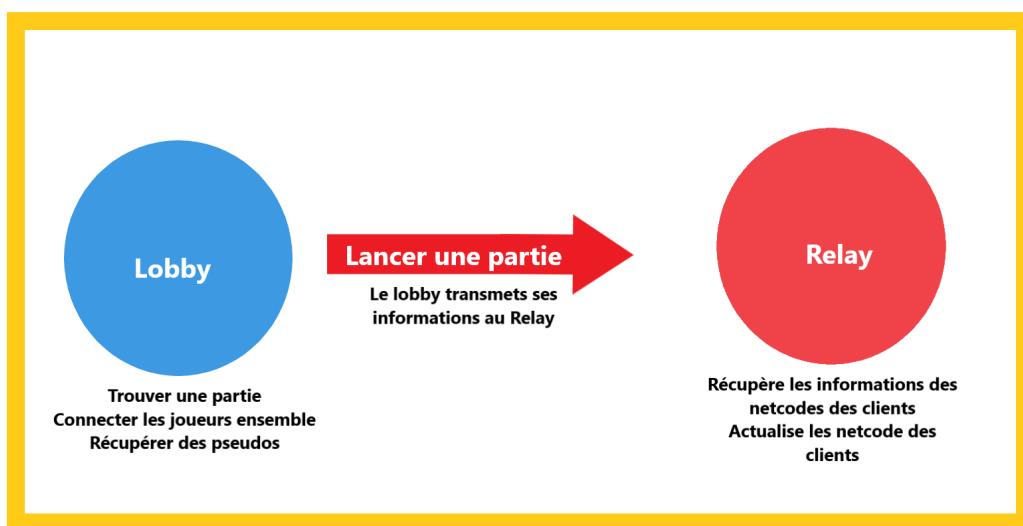


Figure 44: Infrastructure et fonctionnement du serveur

Rapport de projet

2.15.4 Lobby

Comme son nom l'indique, ce service permet de créer des salons où les joueurs peuvent se retrouver. Lorsqu'un joueur se connecte sur le serveur, ce dernier lui donne un ID unique permettant de le différencier avec les autres joueurs. Lorsqu'un joueur a un ID, il peut alors voir tous les lobbies créés et potentiellement les rejoindre. Ce service stockera les informations dites statiques telles que le pseudo, la classe du personnage et l'ID du joueur. Lorsque la partie est lancée, le lobby transmet toutes ces informations au Relay, qui s'occupera de faire apparaître les joueurs dans un environnement 3D.

2.15.5 Relay

Ainsi, c'est le Relay (comme son nom l'indique) qui s'occupera de faire communiquer les joueurs entre eux en réseau, vu alors comme un point de récupération d'informations. Exemple : Le joueur A se déplace en local. Le joueur A communique cette information au Relay qui s'occupera alors de communiquer cette information à tous les autres clients. Ainsi les joueurs B, C et D verront le joueur A se déplacer.

3 Avancement du Projet

3.1 Avancement global / Comparaison avec le cahier des charges

Tableau présentant l'avancée de notre projet et les attentes que nous nous sommes donnés lorsque nous avons commencé sa conception.

	Prévisions	Avancement
Développement I.A.	100%	100%
HUD/Menus	100%	100%
Graphismes/DA	100%	100%
Animation	100%	100%
Multijoueur	100%	100%
Sons/Style des personnages	100%	100%
Développement Général	100%	100%
Site Web	100%	100%
Classes des personnages	100%	100%

	Dans les temps / En avance	En retard
Légende		

Pour résumer, nous avons réalisé tous nos objectifs et sommes satisfaits du résultat final et de l'important travail fourni.

Rapport de projet

3.1.1 Comparaison entre les soutenances

	1 ère	2 ème	3 ème
Développement I.A.	75%	95%	100%
HUD/Menus	10%	75%	100%
Graphismes/DA	20%	60%	100%
Animation	60%	80%	100%
Multijoueur	90%	90%	100%
Sons/Style des personnages	20%	55%	100%
Développement Général	20%	70%	100%
Site Web	15%	90%	100%
Classes des personnages	60%	85%	100%

3.2 Personnages

Les personnages ont tous été implémentés dans le jeu. La création des Hitbox nous a demandé beaucoup de travail, car il a fallu le faire pour chaque personnage. Tout a été réalisé selon cahier des charges.

3.3 Animations

Nous avons réalisé l'ensemble des animations prévues, et nous en avons même ajouté lorsque que cela était nécessaire (e.g., animation de GolluM qui casse une porte)

3.4 Téléphone

Nous sommes très satisfaits de cette fonctionnalité et nous sommes ravis du rendu final. Tout fonctionne comme prévu, et avons même réussi à faire en sorte que GolluM entende lorsqu'un joueur parle dans le chat vocal.

Rapport de projet

3.5 Quêtes

La conception des quêtes a été réalisée selon cahier des charges, et nous a demandé du travail créatif.

3.6 Intelligence Artificielle

Nous sommes très satisfaits du résultat de l'intelligence artificielle. En effet, nous sommes arrivés à développer l'ensemble de nos objectifs afin d'intégrer une intelligence artificielle autonome. De plus, son design étant terrifiant, nous avons vraiment réussi à créer l'effet escompté. Le plus difficile a été la partie concernant l'absorption du son des murs, nous avons dû réfléchir à de nombreuses méthodes afin d'y parvenir. Mais, en appliquant un simple raisonnement mathématique sur les vecteurs, nous y sommes parvenus.

3.7 Réseau

Cette partie de notre projet a été le plus difficile à faire. C'est pour cette raison que nous nous sommes chargés de l'implémenter en premier afin de n'avoir qu'à régler des problèmes mineurs à l'avenir. Pour le rendu final du projet, nous avons complètement changé l'architecture réseau du projet afin de mieux gérer les requêtes réseau.

3.8 HUD / Menus

Les HUD et Menus sont complets, et sommes vraiment satisfaits d'avoir réussi à leur donner un beau design pour le rendu final.

Rapport de projet

3.9 Character Designs

Nous avons avancé comme souhaité, et nous avons même personnalisé nos personnages afin de respecter notre direction artistique.

3.10 Téléphone

Le téléphone est totalement fonctionnel, et pensons que c'est l'une des meilleures fonctionnalités offrant une nouvelle approche des interfaces.

3.11 Carte

La carte est terminée, incluant même les textures extérieures, telles que l'herbe et des arbres.

3.12 Objets

L'ensemble des objets ont été implémentés, notamment ceux servant aux quêtes.

3.13 Graphismes/DA

Tout est conforme au cahier des charges. Avec une satisfaction particulière pour le design des poubelles.

Rapport de projet

4 Notre avis sur le projet

Nous avons travaillé avec engagement et détermination pour créer ce jeu d'horreur captivant et immersif. Chaque membre de l'équipe a apporté sa contribution unique et nous avons su collaborer efficacement pour atteindre nos objectifs. Nous sommes fiers des fonctionnalités que nous avons intégrées et de l'expérience de jeu que nous avons créée. Ce projet "Gollum's Lair" sera pour nous une expérience mémorable et espérons qu'il en soit de même pour les joueurs. Nous sommes ainsi impatients de présenter le projet et de partager aux autres étudiants.

5 Tableau d'avancement du projet pour le rendu final

	Etat actuel
Développement I.A.	100%
HUD/Menus	100%
Graphismes/DA	100%
Animation	100%
Multijoueur	100%
Sons/Style des personnages	100%
Développement Général	100%
Site Web	100%
Classes des personnages	100%

6 Conclusion

En conclusion, nous sommes extrêmement fiers de notre projet, "Gollum's Lair". Nous avons consacré un temps considérable à sa réalisation et nous sommes heureux des résultats obtenus.

Tout au long du développement, nous avons mis en place un certain nombre de fonctionnalités essentielles qui ont grandement

Rapport de projet

amélioré l'expérience de jeu. Parmi celles-ci, nous avons réussi à intégrer avec succès le réseau utilisant Vivox, offrant ainsi une communication fluide et immersive entre les joueurs. De plus, nous avons apporté une attention minutieuse aux détails, allant jusqu'à modéliser les poubelles de l'EPITA, afin de créer une expérience authentique pour les joueurs. La réalisation de la map sur Blender a également été un succès, permettant une exploration riche et diversifiée de l'environnement de jeu.

Enfin, l'ajout d'un chat positionnel a été une réussite majeure, permettant aux joueurs de communiquer de manière réaliste et stratégique en fonction de leur position dans le jeu.

Dans l'ensemble, notre équipe a travaillé avec dévouement et a relevé de nombreux défis tout au long du processus de développement. Nous sommes fiers du résultat final et espérons que "Gollum's Lair" suscitera une expérience captivante et mémorable aux joueurs.

Nous tenons à remercier tous ceux qui ont contribué à la réalisation de ce projet, ainsi que notre école d'ingénieur pour nous avoir donné l'opportunité de mettre en pratique nos connaissances et compétences en informatique.

En conclusion, nous sommes fiers d'avoir mené ce projet à bien et sommes impatients de partager avec notre entourage.

7 Ressources

Logiciel	Fonctionnalité
Unity	Rendu 3D et code
Rider	Code
Blender	3D
Polycam	Scan 3D
Mixamo	Animation
Overleaf	Documents