



RAPPORT 2e SOUTENANCE 2023

Nom du Projet S2 :
ATIPE

Language: *C#*

Chef de projet: Robin ROGER
Yassin HADDAD
Mehdi TURKI
Fabien ROUX
Date: April 18, 2023

Contents

1	Introduction	1
2	Organisation	1
3	Découpage du projet	3
3.1	Ancien Planning	3
3.2	Nouveau Planning	4
4	Script	5
5	Modélisation (Fabien)	13
6	Programmation	21
6.1	Squelette du jeu	21
6.2	Avancement sur l'IA: (Yassin)	22
6.3	Multijoueur	24
6.4	Main menu - mise en place	25
6.5	Inventaire (Robin	26
6.6	Mode De jeu du joueur 2 (Robin)	27
6.7	Utilisation du kit d'outil (Mehdi)	29
7	Site Web	30
8	Conclusion	31

1 Introduction

Bienvenue dans ce rapport de soutenance de notre projet de jeu "ATIPE". Notre équipe a choisi de développer ce jeu pour plusieurs raisons. Tout d'abord, nous voulions créer un jeu divertissant qui permettrait aux joueurs de passer un bon moment tout en valorisant notre école, l'EPITA, et en mettant en avant ses différents lieux emblématiques. De plus, nous avons voulu concevoir un jeu qui incite les joueurs à réfléchir et à collaborer, afin de développer leur esprit d'équipe et leur capacité à résoudre des problèmes ensemble. Enfin, l'aspect nocturne du jeu apporte une dimension d'aventure et de mystère, qui renforce l'immersion des joueurs dans l'histoire. Ainsi, notre jeu "ATIPE" propose aux joueurs de s'immerger dans une aventure palpitante, où ils devront collaborer pour dérober des sujets de partiels au sein de l'EPITA, célèbre école d'ingénieurs. L'un des joueurs sera chargé de guider l'autre au sein des locaux de l'école, à l'aide d'un téléphone portable, le tout dans une ambiance nocturne.

2 Organisation

Pour débuter notre projet nous avons mis en place plusieurs outils afin d'être au maximum organiser. En effet, notre

plus grande crainte fut le désordre. Pour pallier cela nous avons ouvert un emploi du temps sur Trello, où nous pouvons suivre l'avancée de chacun. Le point fort de Trello est que nous pouvons assigner plusieurs personnes sur plusieurs tâches divisées en “CheckList” (=).

De plus, nous partageons une PlayList YouTube, où nous regroupons toutes les vidéos utiles à la réalisation du projet. Les youtubeurs Unity fr, La tech avec Bertrand ou encore Alex nous ont fortement accompagné dans la découverte de Unity. En ajoutant à cela, nous avons un serveur discord afin de communiquer efficacement. Nous avions créé un compte Google commun afin de se transmettre les modèles Blender, mais nous avons arrêté d'utiliser car trop de plateforme freinait l'efficacité de notre travail.

3 Découpage du projet

3.1 Ancien Planning

Table 1: Répartition des tâches (%)

Soutenance	Catégorie	RR	FR	MT	YH	Total
1ère	Programmation du jeu	5	5	5	5	20
	Animations 3D	5	5	5	5	20
	Game Design	15	5		5	25
	Squelette du jeu	5	10	10	10	35
	Création du terrain	50	20	10	5	85
	Recherche Multijoueur	50	50			100
	Déroulé des énigmes	25	25	25	25	100
	Recherches sur IA			50	50	100
Soutenance	Catégorie	RR	FR	MT	YH	Total
2ème	Conception de l'IA	10	10	20	10	50
	Implémentation des énigmes	15	35			50
	Audio & Communication	10	15	15	10	50
	Programmation du jeu	10	10	0	10	50
	Animations 3D	15		25		60
	Game Design		10	15	10	60
	Squelette du jeu	20	10	10	10	85
	Multijoueur		15		25	90
	Création du terrain	20	15	15	15	100
	Déroulé des énigmes					100
Soutenance	Catégorie	RR	FR	MT	YH	Total
3ème	Cinématique		20	80		100
	Conception de l'IA	50				100
	Implémentation des énigmes		50			100
	Audio & Communication				50	100
	Programmation du jeu	15	10	10	15	100
	Animations 3D	25		15		100
	Game Design	10	10	10	10	100
	Squelette du jeu	10		5		100
	Multijoueur		5		5	100
	Création du terrain					100
	Déroulé des énigmes					100

RR → *Robin ROGER* **FR** → *Fabien ROUX*

3.2 Nouveau Planning

Table 2: Répartition des tâches (%)

Soutenance	Catégorie	RR	FR	MT	YH	Total
1ère	Programmation du jeu	5	5	5	5	20
	Animations 3D	5	5	5	5	20
	Game Design					
	Squelette du jeu	40	15	30		35
	Création du terrain	50	20	10	5	85
	Recherche Multijoueur	50	50			100
2ème	Déroulé des énigmes	25	25	25	25	100
	Conception de l'IA			25	25	50
	Implémentation des énigmes	15	35			50
	Audio & Communication	10	15	15	10	50
	Programmation du jeu	10	10	0	10	50
	Animations 3D	30		30		60
3ème	Game Design					
	Squelette du jeu	5	10			85
	Multijoueur		15		35	100
	Création du terrain	20	15	15	15	100
	Déroulé des énigmes					100
	Cinématique		20	80		100

Elements ajoutés	///	Elements retirés	///	Elements modifiés
-------------------------	------------	-------------------------	------------	--------------------------

RR → *Robin ROGER*
MT → *Medhi TURKI*

FR → *Fabien ROUX*
YH → *Yassin HADDAD*

4 Script

Sommaire (chronologique)

Extérieur : Récupération des clés

RDC : Lumières

Etage 1 : Caméra

Etage 1 :

Etage 2 :

Etage 3 : PAS D'ÉTAGE 3

Etage 4 : Amphi

Etage 5 :

Sous-sol :

Etage 6 : Récupération des Partiels (fin)

Informations utiles

Personnage 1: personnage qui veut récupérer les infos sur les partiels.

Personnage 2: personnage qui est chargé de guider son coéquipier.

ZQSD pour se déplacer, “e” pour interagir avec un objet, “i” pour avoir accès à son inventaire. Les deux personnages sont en communication à partir du moment

où le vocal s'active et ils le resteront toute la partie. (Vocal) Lorsque le joueur à EPITA se fera toucher par le gardien, il reviendra à l'entrée mais NE PERDRA PAS SA PROGRESSION. (ceci nous évite les conflits de sauvegarde trop complexes.)

Environnement :

Scène 1 : Dans la rue devant EPITA

Scène 2 : La chambre du perso 1.

Scène 3 : Interieur d'EPITA

Scène 4 : Cour (à la sortie du réfectoire)

Extérieur :

Perso dans la rue. Ambiance angoissante, avec du brouillard, la nuit des lampadaires avec peu de lumière.

POV Perso 1 :

Enigme :

Personnage explore, il va vers l'entrée d'Epita. Il trouve un papier qui dit : “ Le gardien est ici toute la nuit, il paraît qu'il détient un double des clés du Campus dans sa voiture”. Il doit trouver la voiture du gardien. Un message en bas de l'écran s'affiche “Mmmh... la voiture est fermée il me semble avoir vu une caisse à outil traîné devant, je pourrai jeter un coup d'oeil ” Le personnage récupère un pied de biche. Il s'en sert pour ou-

vrir la voiture et récupérer les clés
Il peut ainsi s'introduire dans Epita.

POV Perso 2

Enigme :

Personnage se trouve dans sa chambre. Il reçoit un message de son coéquipier. Il explore sa chambre et trouve son ordinateur. Il doit le déverrouiller. Lorsque le personnage accède à son ordinateur, il est envoyé sur une nouvelle scène un écran d'ordinateur

Exemple :

Le joueur devra avant tout déverrouiller l'ordinateur. Il aura ainsi accès à des caméras de surveillance d'Epita. Le joueur a trop fait la fête la veille, il doit se souvenir de son mot de passe. Pour cela, il doit trouver des indices dans sa chambre. Un rébus pour trouver :

RDC :

Le joueur arrive dans Epita. Le gardien dort l'IA n'est donc pas activé. Porte entrouverte avec gardien qui dort.

POV Perso 1 :

Enigme :

Le personnage est dans l'obscurité. Il trouve une

lampe torche pour l'aider à voir plus clair. Son but est d'aller dans le réfectoire. Il doit trouver un disjoncteur. Lorsqu'il rentre un message sera affiché : "Il fait bien sombre ici, il me semble avoir déjà vu un disjoncteur près d'une machine à café" Il trouve le disjoncteur. Le disjoncteur aura un modèle particulier. Le joueur devra renseigner le modèle à son coéquipier. Son ami aura accès au mode d'emploi du disjoncteur. Ce mode d'emploi contiendra un code couleur à taper. Ce qui permettra d'actionner les lumières.

POV Perso 2 :

Enigme :

Le perso a accès au PC.

Il devra fouiller dans les fichiers afin de trouver le mode d'emploi du bon modèle de disjoncteur.

Il renseignera le code à son coéquipier.

1er étage :

Description :

Le joueur arrive dans Epita. Le gardien dort l'IA n'est donc pas activé.

POV Perso 1 :

Enigme :

Le personnage devra suivre les indications de son coéquipier.

Il trouvera au 1er étage une affiche avec des informations sur des changements d'Epita. “GRAND CHANGEMENT À EPITA Rénovation des toilettes Atelier Création de jeu de Palet Ouverture d'un nouveau club de Polos Nouveaux Casiers situé au 1er étage Soirée au Perroquet Bourrée le 31 février “ Cette affiche lui donnera l'indication de trouver les casiers. Il aura le code grâce à son coéquipier. Le joueur détient la clé usb. Avec l'aide de son coéquipier, le joueur devra trouver les serveurs d'Epita. Les serveurs se trouveront dans une pièce secrète. Cette pièce est située dans les escaliers reliant le rez-de chaussé et le premier étage. Le logo IONIS est une porte.

Pour ouvrir cette pièce le joueur passera devant la machine à café et verra affiché “Tiens, la machine à café à l'air fonctionnelle, pour une fois...” Un bouton sur cette même machine permettra d'ouvrir la porte.

Il introduira la clé USB à l'intérieur. Ceci débloque l'accès au caméra de l'étage 1, au joueur n2. Mais active aussi l'arrivée du gardien.

POV Perso 2 :

Enigme :

Le joueur aura accès à une application Caméra. L’application sera bloquée au départ. Il devra fouiller dans ses fichiers et trouver un dossier Projet Hacking Caméra.

Ce dossier contiendra un README avec dedans “Code casier : 5944” (style emacs CamL)

Dans ce même dossier il y aura un fichier qui renseignera que la clé usb devra être introduite dans les serveurs de Epita.

2eme Etage :

Description :

Le joueur arrive au second étage, les caméra sont activé a l’étage n1 POV Perso 1 :

Enigme :

Afin d'accéder à l'étage supérieur, le joueur devra se rendre ensuite dans les bureaux des professeurs afin de trouver une clé qui permettra d'accéder au prochain étage.

Les portes seront déverrouillées lorsque les caméras seront activées.

Le joueur en arrivant devant les bureaux verra affiché

: “Il me semble que je me trouve dans les bureaux des professeurs, j’ai entendu dire que M.Saber travaillait très tard, il faut que je sois bien vigilant.”

Le joueur aura donc deux choix d’ouverture de porte. Il pourra ouvrir le bureau intitulé Professeur, ou il y aura un professeur en train de travailler. Si le joueur rentre dans la pièce, il perd la partie. Le bon choix sera la salle de la secrétaire. A l’intérieur il sera guidé par son coéquipier afin de fouiller son bureau.

POV Perso 2 :

Enigme :

Le joueur aura accès à un dossier d’archive avec des anciennes vidéos des caméras L’une d’elle indique l’emplacement de la clé. (vidéo de la secrétaire qui pose sa clé dans un certain endroit.)

3eme Etage

Description :

Pas d’étage 3 à Epita, Ce sera précisé dans l’interface des caméras du joueur 2.

4eme Etage

Description :

Découverte de l’amphithéâtre, les deux joueurs seront

coupés en arrivant à ce niveau.

POV Perso 1 :

Enigme :

Le joueur devra essayer d'allumer le projecteur, il devra aller chercher un ordinateur situé dans les rangs de l'amphithéâtre, dans un sac oublié par un élève.

POV Perso 2 :

Enigme :

Le PC affichera un problème réseau, le joueur sera contraint à quitter son PC et devra résoudre le problème. Il sortira de sa chambre

5eme Etage (final)

Description :

Etage final, le joueur trouvera une salle secrète où repose les partiels.

POV Perso 1 :

Enigme :

Le joueur devra trouver une salle avec un tableau où il y aura marqué ACDC. Les pc seront allumées et sur chaque pc il y aura marqué la commande linux “man

ascii”

POV Perso 2 :

Enigme :

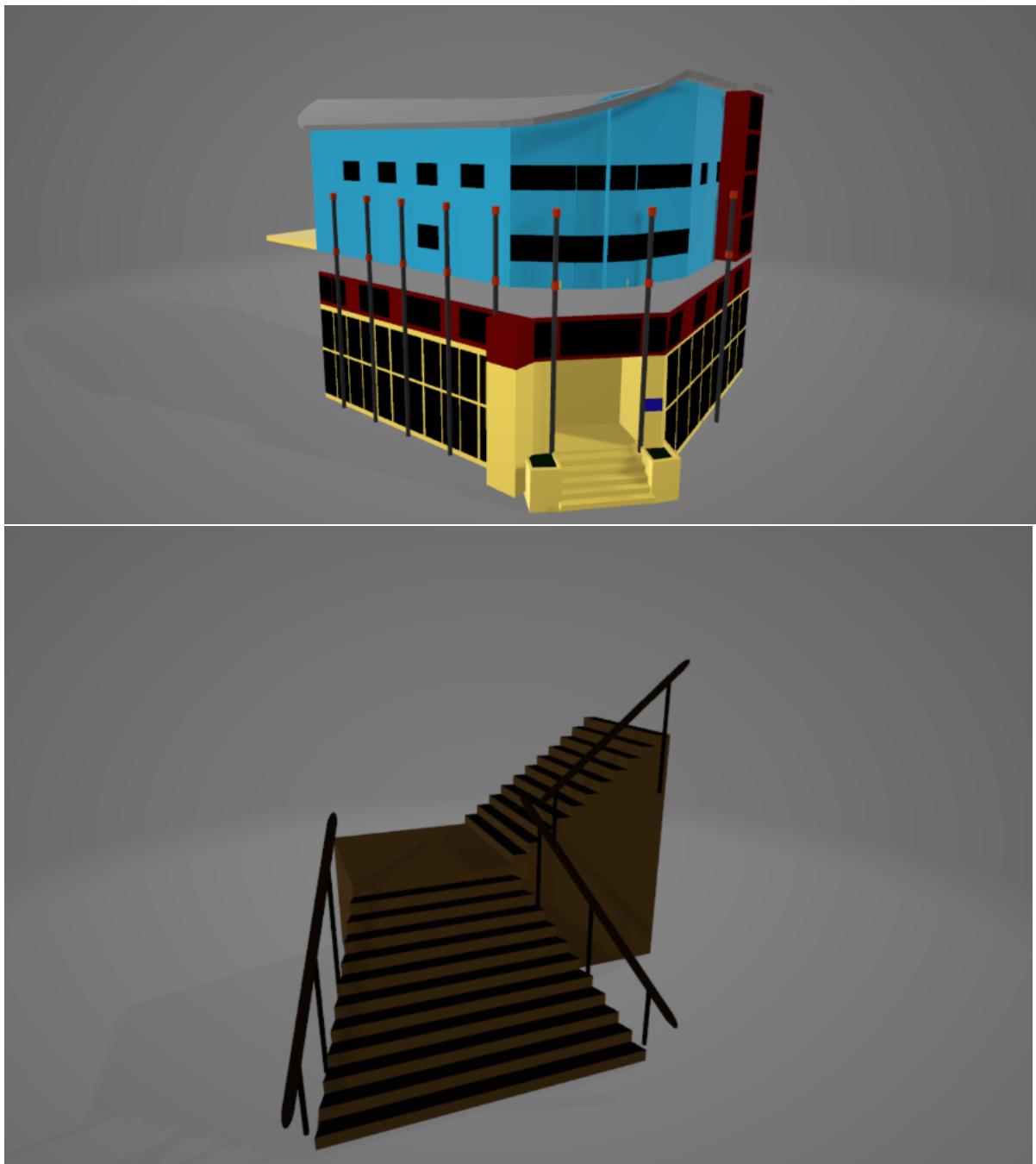
Le joueur aura accès à un fichier contenant la table ascii et devra guider son ami pour trouver le code.

Fin :

Le joueur récupérant les partiels débloque la sortie et peut s'échapper tout en fuyant le vigile.

5 Modélisation (Fabien)

Pour modéliser le jeu nous avons tout réalisé sur Blender. Nous avons créé la plupart des objets. Néanmoins nous avons fait appel à des sites gratuits qui proposent des modèles 3D. Par exemple, TurboSquid, Open 3D Model, Free 3D ont servi pour les objets les plus complexes, comme les plantes par exemple. Voici deux exemples de modèles que nous avons réalisée



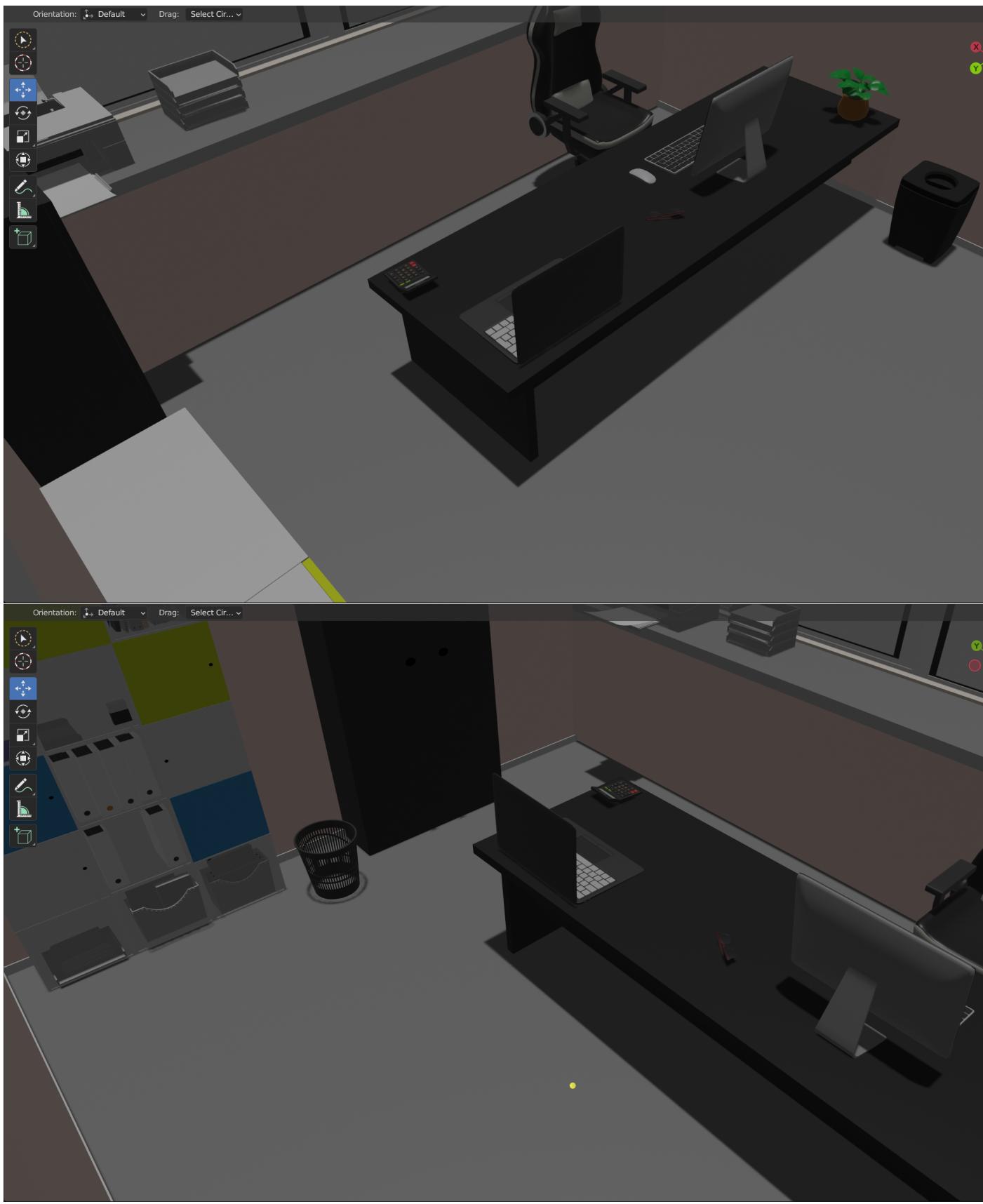
Nous avons beaucoup avancé sur l'intérieur, en effet le bâtiment entier est opérationnel et détaillé, à l'exception du bureau de M. Saber, qui n'est pas encore opérationnel, mais ça n'est pas un frein dans l'absolu, étant donné que nous avons juste besoin d'une vidéo de lui qui se lève de son bureau dans ladite salle pour s'approcher

de la porte. Egalement, il reste la chambre du second joueur qui reste à finir, car certains éléments clés sont à poursuivre, comme le couloir qui mène à une autre salle. Cette autre salle permettra d'accéder à un indice pour le bien du game design qui a été pensé en amont.

Pour l'instant l'avancement de cette ébauche essentielle à terminer reste en suspens
Pour ce qui est du travail achevé, voici un plan vu de haut pour avoir une idée globale



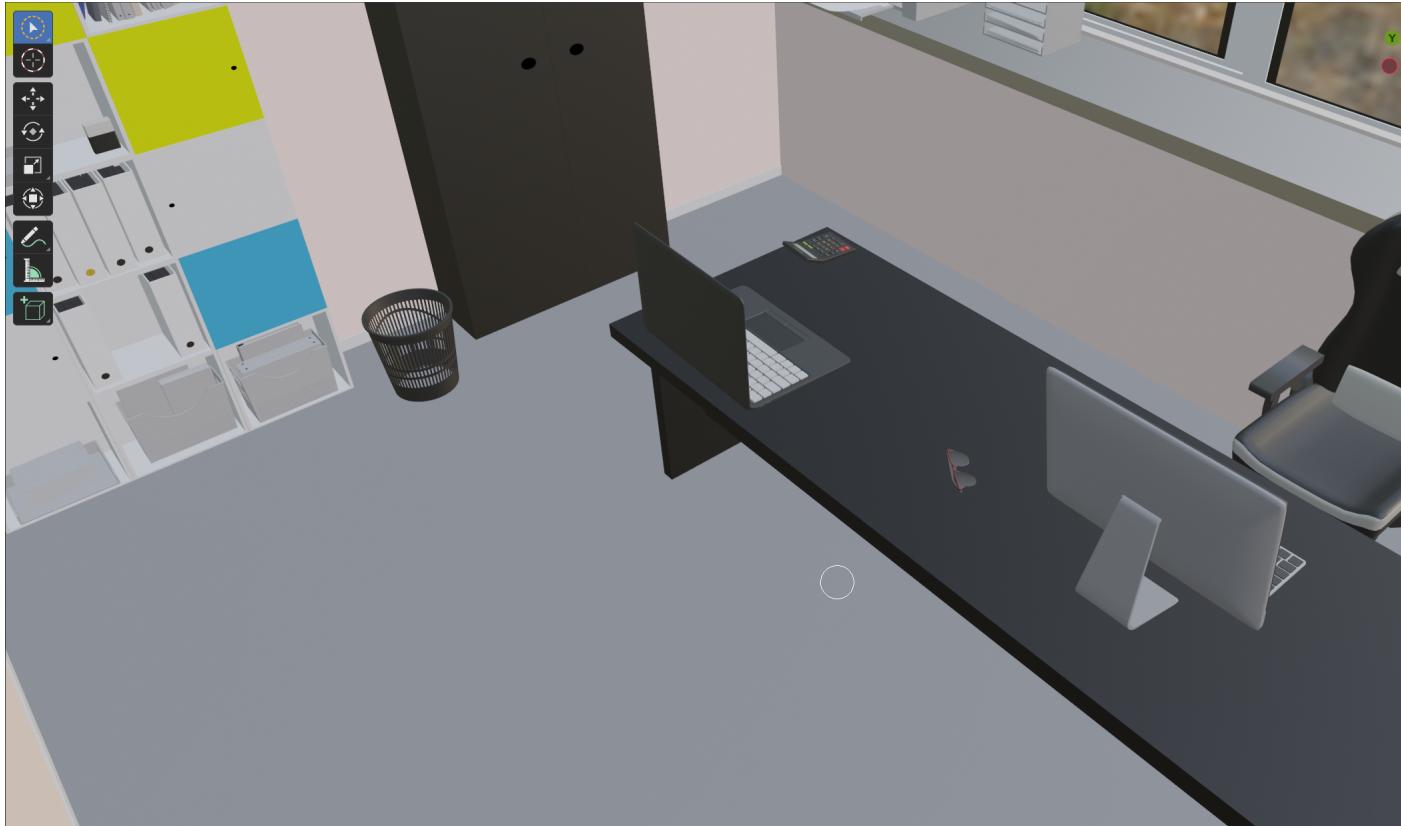
Et voici une vue en 3D pour avoir une vision plus précise de l'avancée du bureau :



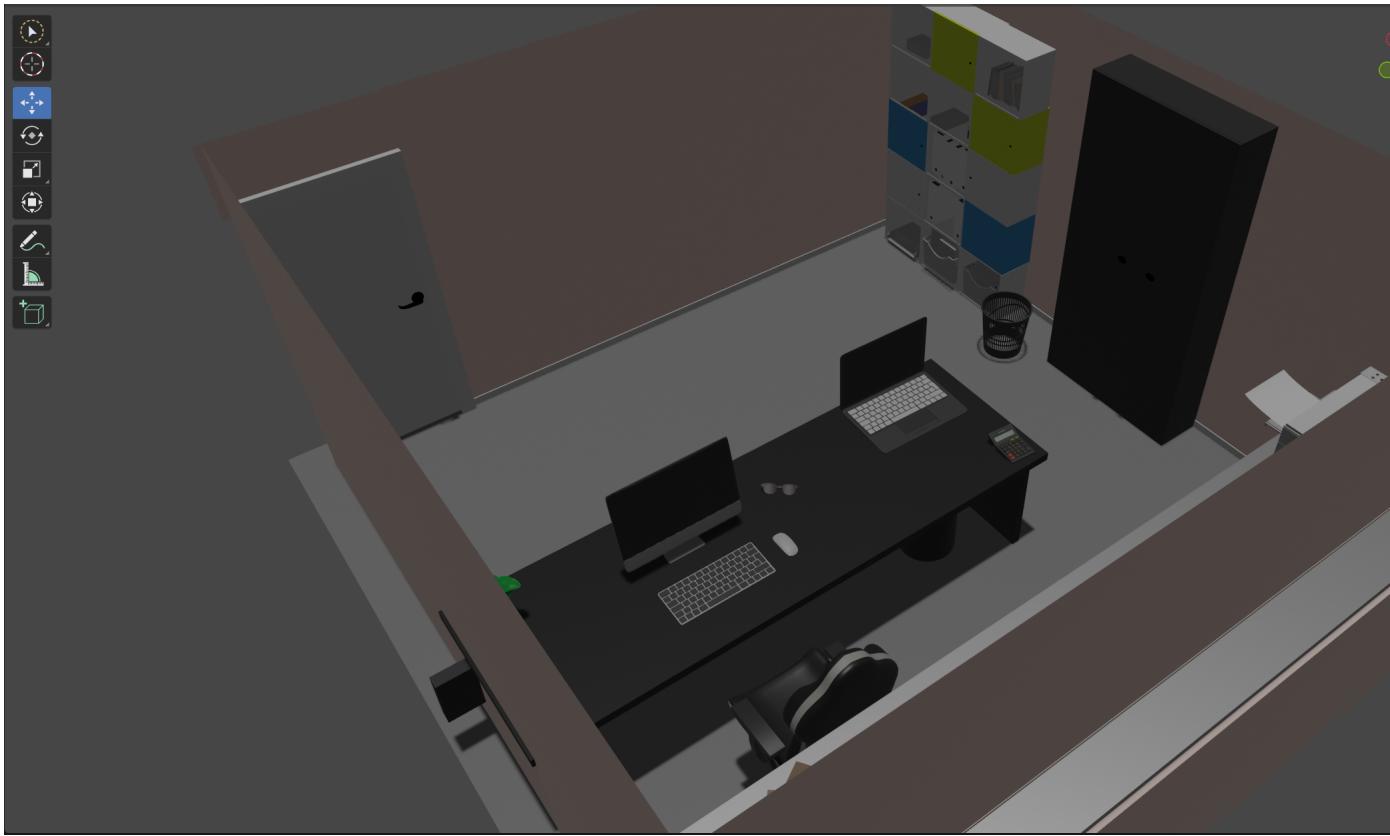
A noter que les lumières permettent juste d'avoir un

rendu plus visuel de la modélisation, qui n'est pas entièrement représentatif du bureau sur Unity. En effet la lumière simulée sur les photos ci-dessus sont celles d'un Soleil, et par conséquent, les effets de lumières seront dans le jeu plus précises.

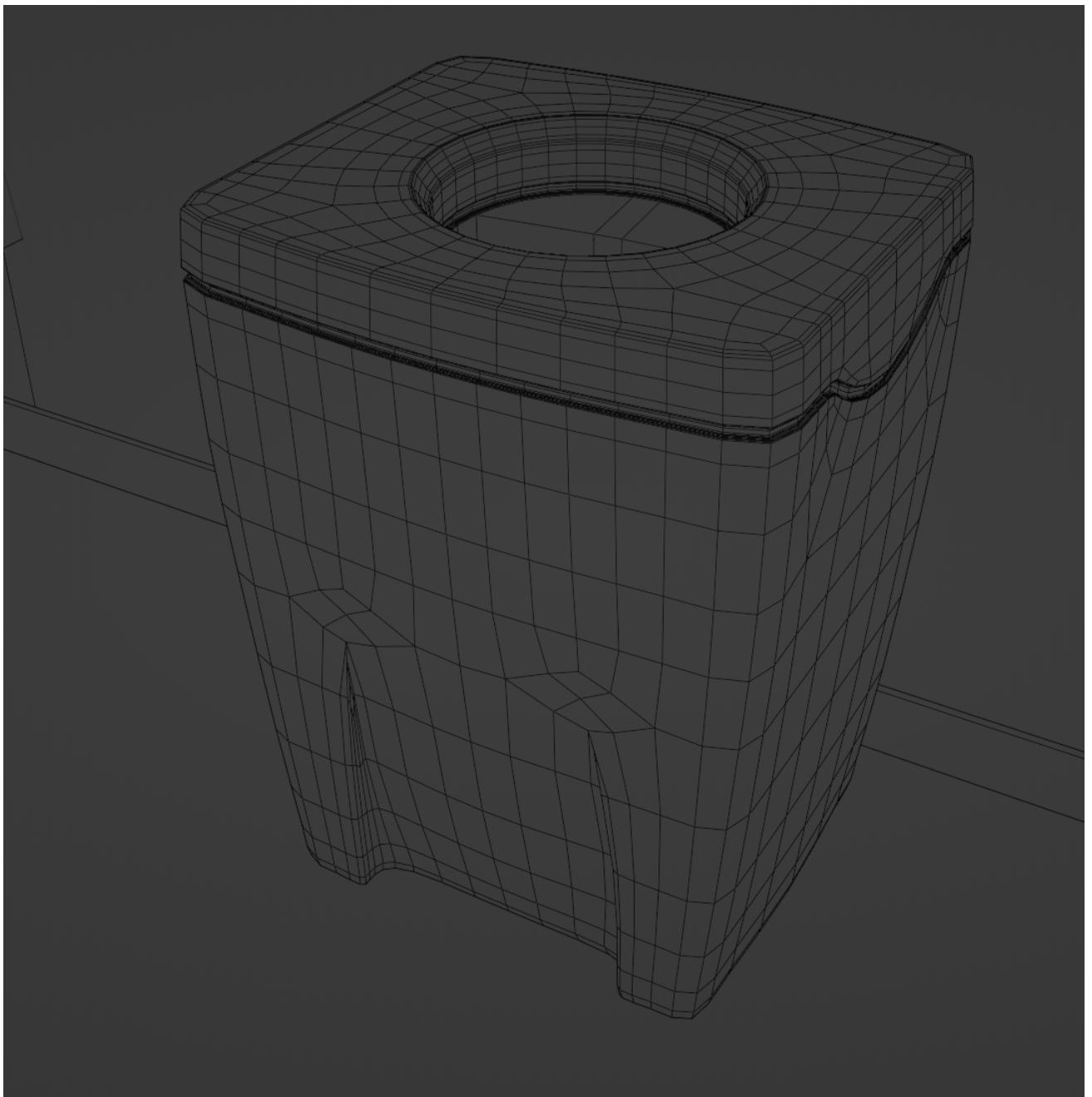
Par ailleurs vous pouvez voir que si l'on retire les effets de lumière, on obtient un rendu bien moins agréable, surtout en termes de couleurs, étant donné que les objets ne sont pas texturés, je le rappelle, pour optimiser les performances graphiques



Donc, par convénience, nous utiliserons ce type de rendu pour les prochaines captures d'écran à venir



Nous avons continué de modéliser les objets dans un style cartoon, toujours dans l'optique de gagner des performances, puisque les textures sont gourmandes en puissance de carte graphique. Exemple en image :



L'objet ci-dessus est en mode Wireframe. C'est un Viewport qui permet de voir à travers les objets, en l'occurrence ça nous permet de voir que c'est dans un style très épuré, sans tomber dans l'excès pour différencier ce broyeur d'une poubelle.

Puisqu'on est pas en monde ouvert, pour gagner en

performance encore une fois, ça explique le fait qu'il n'y ait pas de plafond et rien autour des murs.



6 Programmation

6.1 Squelette du jeu

Nous avons débuté la programmation avec la création du squelette du joueur. Voici comment nous avons divisé la programmation du personnage : (Robin)

Player Controller :

Ici on gère les touches pressées par le joueur afin de bouger avec l'aide des commandes GetAxisRaw. De

plus, nous avons fait appel à la structure Vector3 déjà implémentée dans Unity. Cette structure est utilisée dans Unity pour transmettre des positions et des directions 3D. Il contient également des fonctions pour effectuer des opérations vectorielles courantes. Dans cette classe on gère aussi la rotation du joueur, en gérant les mouvement de la souris

Player Motor

Le script gère le mouvement du joueur. Il est relié à Player Controller qui lui s'occupe comme son nom l'indique des contrôles du joueur. Il utilise le Rigid-Body de notre personnage et lui appliquera des changements de positions. La méthode PerformMovement est chargée de faire le mouvement.

Player interact

Ce script est chargé de l'interaction du joueur avec les objets. Nous utilisons un Raycast qui est une ligne invisible constamment présente dans la caméra du joueur. Ainsi lorsque le joueur aura au centre de son point de vue un objet il sera en capacité d'interagir avec.

6.2 Avancement sur l'IA: (Yassin)

Lors de la dernière soutenance, nous avons rapidement présenté un algorithme d'IA simple, composé de 3 fonctions, Patrolling, Chase et Update. Depuis, nous avons

perfectionné ce qui va être l'ennemi principal de notre jeu de plusieurs façons

Déplacements

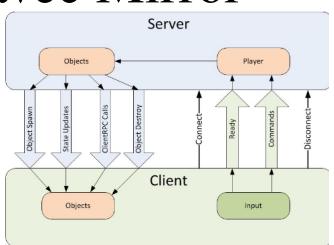
Nous avons apporté des modifications sur le déplacement du gardien, celui-ci fait désormais la différence entre les endroits où il peut se déplacer (sol, escalier...) et ceux où il ne peut pas (traverser un mur par exemple). L'IA est désormais consciente des obstacles, ce qui n'était pas le cas lors de la dernière soutenance.

Ajout d'un état d'alerte

NB : Cet état est encore discuté et peut être retiré avant la finalisation du jeu Pour rendre le jeu plus réaliste, nous avons ajouté un nouvel état au gardien, l'état d'alerte. Quand le joueur a été repéré et réussi à s'enfuir du champ de vision du gardien, celui-ci ne revient pas directement à l'état de patrouille, mais reste pendant un certain temps en état d'alerte, cet état augmente légèrement sa vitesse initiale, et fait se déplacer le gardien dans les étages où se trouvent le joueur (EX : si le joueur est à l'étage 5, le gardien se déplacera uniquement dans les étages 4 à 6).

6.3 Multijoueur

Pour synchroniser nos joueurs nous allons utiliser l'asset gratuite proposé par Unity, Mirror. Mirror est un outil efficace afin de générer la mise en réseau du projet. Deux composants de Mirror : le NetworkManager (permet de gérer le réseau) et le NetworkHUD (permet de gérer les personnage accueillie sur le réseau). Mirror utilise un système de communication entre chaque objet qui interagit dans le jeu ainsi nous devons ajouter un NetworkIdentity à chaque objet utile de notre jeu. (ex : joueur / item / etc ...) Transmission des informations avec Mirror



Un problème que nous avons rencontré lors de la réalisation du multijoueur est que les joueurs sont connectés les uns avec les autres et dès lors qu'un personnage bouge l'autre aussi. Ainsi nous avons réalisé un Script nommé PlayerSetup, qui sera implémenté sur chacun des joueurs. Le script précisera que les actions réalisées seront faites uniquement sur le joueur sélectionné. En d'autres termes nous préciserons les composants à désactiver pour le joueur.

Du côté de la synchronisation, nous avons choisis d'optimiser les packages envoyer au serveur en les limitant à 100 millisecondes entre chaque envoie. En effet, si nous baissions au maximum ce temps, le serveurs seraient surchargés et cela ferait l'effet inverse les packages seraient triés en file d'attente et le jeu serait saccadé.

6.4 Main menu - mise en place

Pour la mise en place du menu principal, nous avons décidé d'utiliser les méthodes déjà pré-faites par Mirror. Tout d'abord nous avons mis en place un Canevas afin de pouvoir adapter les objets “UI” à chaque PC. Les objets “UI” utilisés sont les suivants : bouton, image, texte (package Text Mesh Pro). Ainsi nous avons assigné à chaque bouton une méthode bien précise. Par exemple, le bouton ”créer un serveur” est assigné à la méthode StartServer().

Ici on a rencontré plusieurs problèmes:

Les RoomPlayer sont des objets qui permettent de synchroniser les personnages en multijoueur, ils se multiplient au fur à mesure que les joueurs rejoignent le serveur. Or pour réaliser un système de Prêt / Pas prêt cela nécessite de récupérer le RoomPlayer du joueur et seulement le sien. Pour ce faire , nous avons limité le

nombre de joueurs à 2 et avons ajouté un tag à chaque joueur.

Les UI. Pour rendre notre jeu plus esthétique il m'a fallu changer les UI ce qui ne fut pas de tout repos. Dans unity on utilise un système de Sprite (qui est un format nécessaire pour ajouter des texture au boutons), il m'a été difficile de comprendre comment créer ma propre UI et ai donc fait appel à un paquet gratuit sur Unity Hub.

6.5 Inventaire (Robin)

Tout d'abord, pour créer un système d'inventaire dans Unity, nous avons créé un objet vide dans la hiérarchie de Unity, que nous avons nommé "Inventaire". C'est le conteneur pour tous les éléments de l'inventaire.

Ensuite, nous avons ajouté un script à cet objet, qui nous permet de gérer l'inventaire. Ce script contient une liste d'objets (les éléments de l'inventaire) et des fonctions pour ajouter, supprimer ou déplacer ces éléments.

Pour représenter les différents éléments de l'inventaire,

nous avons créé des objets tels que des potions, des armes et des clés. Ces objets peuvent être des préfabriqués ou des objets que nous avons créés nous-mêmes.

Nous avons également ajouté des scripts à ces objets,

qui leur permettent d’interagir avec le script d’inventaire. Par exemple, un objet ”Lampe” peut avoir un script qui ajoute cette lampe à l’inventaire lorsque le joueur clique dessus.

Ensuite, nous avons placé les objets dans le monde. Lorsque le joueur interagit avec eux (en cliquant sur ”e”), il doit ajouter l’objet à l’inventaire.

Pour afficher l’inventaire, nous avons ajouté une interface utilisateur (UI), qui inclut une grille de cases, dans laquelle chaque case représente un emplacement dans l’inventaire. Lorsqu’un objet est ajouté à l’inventaire, il apparaît dans l’une des cases de l’interface.

6.6 Mode De jeu du joueur 2 (Robin)

Nous nous sommes renseignés sur la façon de modéliser l’interface du joueur 2 et nous avons décidé de créer une interface utilisateur PC simpliste dans Unity. Pour cela, nous allons suivre les étapes suivantes :

Tout d’abord, nous allons créer un objet vide dans la hiérarchie de Unity et le nommer ”Bureau”. Ce sera le conteneur de notre interface utilisateur.

Ensuite, nous ajouterons un composant Canevas à l’objet Bureau. Cela créera un nouveau Canevas pour

l’interface utilisateur. Un fond d’écran d’ordinateur sera ajouté afin de rendre l’expérience le plus réaliste possible.

Nous ajouterons également un composant CanvasScaler à l’objet Bureau. Cela garantira que l’interface utilisateur est mise à l’échelle correctement pour différentes résolutions d’écran.

Nous ajouterons également un composant ”Graphic Raycaster” à l’objet Bureau. Cela permettra aux utilisateurs d’interagir avec les éléments de l’interface utilisateur.

Pour l’arrière-plan de notre interface utilisateur, nous ajouterons un nouveau jeu d’images de fond. Nous créerons une image de bureau que nous souhaitons utiliser et nous l’ajouterons à l’objet Bureau.

Pour les dossiers et applications à ouvrir dans le bureau **nous utiliserons des boutons** qui permettent d’afficher des images. Nous ajouterons un script à ce bouton pour lui permettre d’ouvrir l’image lorsqu’il est cliqué. Pour cela, nous créerons un nouveau script. Dans le script, nous écrirons un code qui chargera et affichera l’image souhaitée lorsqu’on clique sur le bouton.

Pour afficher l’image, nous créerons un objet vide, le nommerons ”Image”, y ajouterons un SpriteRenderer et lui assignerons la texture de l’image que nous souhaitons

afficher.

Finalement, nous pourrons ajouter des éléments supplémentaires à l'interface utilisateur tels que des icônes, des boutons, ou tout ce que nous voulons, en utilisant les outils UI de Unity.

En suivant ces étapes, nous pourrons créer une interface utilisateur PC simpliste dans Unity pour le joueur 2 de notre jeu. En expérimentant avec les différentes options et fonctions de l'interface utilisateur de Unity, nous pourrons créer une interface utilisateur unique pour notre jeu.

6.7 Utilisation du kit d'outil (Mehdi)

Comme nous l'avons stipulé dans la partie précédente, le logiciel Unity propose un kit d'outil issu d'une élaboration très poussée nous permettant de compléter différents types de création selon notre volonté. Le kit d'outil prend en charge la création et le débogage d'interfaces utilisateur d'exécution pour les jeux et les applications, avec un flux de travail intuitif qui aide les artistes et les concepteurs à démarrer plus rapidement. Les outils de création dédiés du kit d'outils IU incluent UI Builder et UI Debugger, inspirés des technologies Web pour offrir une expérience familière. Utilisez UI Builder pour

créer et modifier visuellement l’interface utilisateur des jeux et des applications directement dans Unity, tandis que UI Debugger vous aide à résoudre rapidement les problèmes.

Le kit d’outils IU est livré avec un système d’animation d’interface utilisateur dédié qui simplifie la production de transitions d’interface utilisateur dans vos projets. Ce nouveau système permet même de créer efficacement des transitions réutilisables, qui constituent la majorité des animations dans la plupart des projets d’interface utilisateur, afin de nous faire gagner un temps précieux de conception.

7 Site Web

Notre site web est enfin en ligne. Vous pouvez y accéder ici <https://atipe.fr>

A l’heure actuelle, il n’est pas encore terminé, de nombreuses modifications vont être déployées d’ici la soutenance, rendant le site plus ergonomique, et plus fonctionnel. Néanmoins, malgré l’absence de ces modifications pour le moment, le site répond à tous les critères demandés. La page d’accueil résume le projet, présente et donne le contact des différents membres de l’équipe,

donne une chronologie de notre travail, explique les problèmes que nous avons rencontrés et les solutions que nous avons trouvés.

Un site de documentation (docs.atipe.fr) sera également prochainement disponible, d'ici la soutenance, et contiendra tous nos rapports, cahier des charges et documents du projet.

Enfin, une page contact est disponible et permet d'envoyer un mail à l'adresse contact@atipe.fr.

8 Conclusion

Les objectifs remplis pour la première soutenance :

Nous avons terminée la modélisation d'Epita.

Nos objectifs pour la prochaine soutenance:

Nous voulons avoir intégralement fini le développement des joueurs (en particulier le joueur 1, et si possible le joueur 2), avoir intégré l'IA dans le jeu (afin qu'elle puisse se déplacer sans problème, avec la vitesse adéquate), avancer voire finir les énigmes du script et enfin terminer le multijoueur.

Par ailleurs dans l'idéal Mme Tal et M. Saber seront réalisés d'ici là avec PifuHD.