



## Rapport de Soutenance Finale

---

# ***We Don't See Sharp***

---

**Flush Hand Studio™**

***Membres de l'équipe :***

Adrien GOMES

Laetitia BERKOUNE

Doria BOUZIANE

Emma RUCAY





# Sommaire

1- Introduction .....	1
1.1- Contexte du projet.....	1
1.2- Objectifs pédagogiques et techniques .....	1
1.3- Structure du rapport.....	1
1.4- Outils de gestion principaux .....	1
2- Présentation générale du projet .....	2
2.1- Résumé du concept .....	2
2.2- Scénario résumé .....	3
2.3- Originalité et innovations techniques.....	4
2.4- Intérêt du projet pour les joueurs et l'industrie vidéoludique .....	5
3- Communication et promotion.....	5
3.1 Réseaux Sociaux.....	5
3.2 Site Web.....	5
4- Organisation et méthodologie .....	7
4.1- Répartition des tâches .....	7
4.2 Diagramme de Gantt (barres pour soutenances et taux avancement).....	9
5- Réalisations techniques.....	10
5.1- Mécaniques de gameplay .....	10
5.2- Graphismes et animations.....	12
5.3 Intelligence artificielle.....	21
5.4- Détection des mouvements.....	24
5.5- Multijoueur.....	26
6- Bilan intermédiaire.....	28
6.1- Avancées réalisées depuis la S1.....	28
6.2- Avancées réalisées depuis la soutenance S2 .....	30
6.3- Défis rencontrés et retards, solutions et/ou explications .....	32
6.4- Comparaison des progrès par rapport aux objectifs initiaux Objectifs atteints.....	37
7- Conclusion.....	38
Annexes .....	40



# 1- Introduction

## 1.1- Contexte du projet

L'équipe de Flush Hand StudioTM vous propose, à travers ce Rapport, de découvrir l'organisation, le développement et les idées de conception de son premier jeu vidéo, We Don't See Sharp. Ce jeu d'horreur en vue à la première personne se distingue par ses mécaniques de puzzle et ses interactions environnementales immersives.

L'objectif fondamental de ce projet est de créer une expérience vidéoludique captivante, où chaque choix des joueurs impacte le déroulement de l'histoire. Pour atteindre cet objectif, l'équipe s'appuie sur Unity3D [1] et le langage C#, permettant d'explorer pleinement ses compétences techniques afin d'offrir un gameplay fluide et dynamique.

## 1.2- Objectifs pédagogiques et techniques

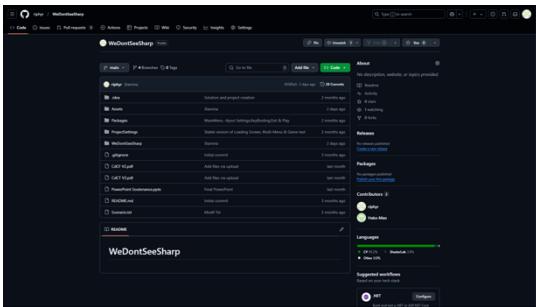
Ce projet valorise le travail collaboratif et l'acquisition de compétences en gestion de projet. Il intègre des concepts originaux comme l'utilisation de cartes de poker pour influencer l'issue du jeu pour ajouter davantage d'interaction entre le joueur et son environnement. En combinant tension psychologique et réflexion stratégique, il offre une narration riche et interactive. Ce projet favorise également l'apprentissage des techniques de programmation en C# et la gestion efficace de ressources communes.

## 1.3- Structure du rapport

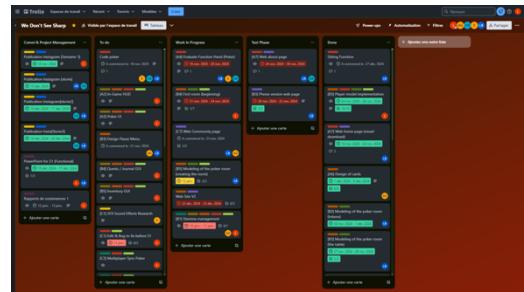
Ce rapport est structuré de manière à présenter successivement le contexte du projet, la présentation générale, l'organisation avec sa méthodologie ainsi que l'avancement, les réalisations techniques (récapitulatives et nouvelles), le bilan pré rendu final à l'état actuel du projet [2], avant de conclure avec une synthèse et des annexes détaillées.

## 1.4- Outils de gestion principaux

Pour mettre en place ce projet nous avons utilisé deux outils majeurs qui permettent à la fois de communiquer des informations mais aussi de gagner énormément en efficacité sur la gestion du travail de groupe : GitGub [4] et Trello [5]



- GitHub Project -



- Méthodologie kankan par Trello -

## 2- Présentation générale du projet

### 2.1- Résumé du concept

Notre projet consiste en la création d'un jeu vidéo jouable à la fois en solo et en multijoueur en vue à la première personne, combinant des éléments narratifs et des mécaniques de puzzle au sein d'un environnement 3D immersif. We Don't See Sharp commence dans un club de poker clandestin avant de basculer dans un univers mystérieux. Le joueur explore des zones thématiques représentant des moments clefs de la vie d'un personnage décédé au cœur du lore du jeu.

⇒ Une fois transporté dans le hub mystérieux, le joueur découvre un espace central entouré de portes, chacune menant à une salle thématique représentant une étape clef de la vie d'un personnage décédé plusieurs années auparavant dans le même club de poker clandestin que celui dans lequel le joueur apparaît au lancement de sa session de jeu. Pour progresser et espérer s'échapper de cet endroit oppressant, le joueur devra explorer ces salles, qui retracent des moments marquants de la vie de cette âme tourmentée. Ces espaces incluent des lieux symboliques, tels que la maison où il a vécu, ou des événements charnières de sa vie d'adulte. Chaque salle est conçue pour dévoiler progressivement les éléments du récit, à travers des énigmes à résoudre, des objets à manipuler, et des indices à collecter. Au fur et à mesure de ces explorations, le joueur reconstitue le contexte tragique de la mort du personnage, tout en étant confronté à des manifestations paranormales et à des choix cruciaux qui influenceront le dénouement de la partie.

⇒ Le mode multijoueur est coopératif et se joue en ligne. Les joueurs travaillent ensemble pour résoudre des puzzles et découvrir des indices qui débloqueront des parties de l'histoire. Ils devront interagir avec l'environnement, utiliser des objets

**trouvés sur place, dont des cartes de poker qui ont un rôle déterminant dans le déroulement du scénario, et affronter des événements paranormaux de plus en plus intenses. L'aventure en mode solo consistera quant à elle à un jeu certes proche dans le fond mais dans la forme, il y aura plus d'intimité, de pression et de frayeurs que le joueur se fera à lui-même.**

## 2.2- Scénario résumé

We Don't See Sharp ! commence dans une ambiance calme et intrigante au sein d'un club de poker clandestin, caché au cœur d'une boîte de nuit, où le joueur prend place à une table avec divers personnages non joueurs (PNJ [6]). Ce cadre tamisé est rapidement assombri par une atmosphère sinistre lorsque la partie commence. Les enjeux deviennent significatifs, et le joueur croit naïvement que la soirée se déroulera sans accroc. Cependant, la tension monte lorsque des événements inexplicables commencent à se produire : des éclats de rire étranges, des murmures inquiétants, ou encore des ombres furtives apparaissant dans la salle.

Soudain, le décor se transforme, la lumière s'éteint, et le joueur chute dans un espace alternatif (style backrooms [7]), un hub mystérieux qui sera le centre névralgique de son aventure. Ce lieu désorientant présente différentes portes avec des décorations qui donnent au joueur des indices sur ce qui l'attend, cela incite le joueur à l'exploration. Chacune des portes de ce hub mène à un niveau distinct, représentant une facette de la vie de l'entité hantée (comme précisé dans la section ci-dessus) qui le suit depuis qu'il est tombé dans cet environnement inconnu. Ces niveaux sont soigneusement conçus pour refléter des moments clefs de son existence, tels que :

- Sa maison d'enfance : Une reconstitution détaillée de son domicile, où le joueur peut interagir avec des objets du passé, revivre des souvenirs et découvrir des indices sur les événements tragiques qui ont conduit à sa mort. Le tout en évoluant progressivement dans cet espace à travers des énigmes.
- Le labyrinthe : Dans cette zone, le joueur se retrouve plongé dans un vaste labyrinthe extérieur, constitué de hauts murs. L'atmosphère est marquée par une obscurité alternée avec des lumières rouges clignotantes, avec une musique angoissante, de la végétation vivante... Alors qu'il explore les recoins inhospitaliers du labyrinthe, le joueur doit naviguer habilement pour éviter l'entité malveillante qui surgis de l'ombre, reflétant les peurs et les regrets de l'entité hantée. Sans oublier les cartes à récupérer pour avancer dans le jeu et pouvoir espérer en sortir vivant.

■ Son appartement **Nous avons modélisé une scène** qui s'inscrit dans l'univers de notre jeu d'horreur. Elle se déroule dans un **studio abandonné**, ancien lieu de torture infligée à John, un personnage central de notre histoire, victime d'une dette de poker qu'il n'a jamais pu rembourser. Le concept de cette scène repose sur une **mécanique de jeu simple mais efficace** : le joueur dispose de **10 minutes** pour explorer le studio et **récupérer un maximum de cartes** disséminées dans l'environnement. En tout, **30 cartes** sont cachées dans la pièce. L'objectif est de réunir suffisamment de cartes pour **former un paquet complet**, indispensable pour accéder à la **partie finale du jeu**. Ce compte à rebours renforce la **tension et l'urgence**, en cohérence avec l'atmosphère oppressante du studio marqué par le **passé tragique de John**.

Au cours de chaque niveau, des apparitions paranormales, telles que des silhouettes floues, des rires ou encore des cris, sont conçues pour susciter la peur et le mystère. Les joueurs devront interagir avec des objets spécifiques, fouiller, et parfois résoudre des énigmes plus ou moins complexes pour progresser. Les choix que le joueur effectue dans ces niveaux peuvent également influer l'histoire, créant des chemins narratifs alternatifs. Par exemple, choisir d'interagir avec un certain objet peut révéler un indice crucial ou déclencher un événement clef.

L'objectif alternatif du joueur est de découvrir la vérité sur la vie tragique de l'entité qui hante le hub, tout en cherchant un moyen de s'échapper de ce monde alternatif. Les joueurs collecteront des cartes de poker tout au long de leur parcours, qui serviront à débloquer un combat final de cartes contre une représentation de l'entité. Ce combat ne sera pas simplement une question de chance, mais exigera une compréhension approfondie des événements et des choix effectués tout au long du jeu.

En intégrant des éléments de gameplay de type "puzzle", chaque pièce présente des défis uniques. Parfois, le joueur devra fouiller et comprendre des éléments de décor, qui serviront à la fois de mécanisme de progression et de révélateurs de l'histoire. Ces rencontres permettent également d'en apprendre davantage sur les PNJ, leurs expériences et leurs liens avec le monde du jeu, enrichissant ainsi le récit.

Le jeu est conçu pour être une expérience immersive où le joueur ressent une véritable pression psychologique, alimentée par l'ambiance sonore, les événements inattendus et la profondeur narrative. Chaque action, choix ou interaction a un impact, rendant l'expérience personnelle et mémorable.

### 2.3- Originalité et innovations techniques

Le jeu introduit des concepts innovants tels que l'utilisation des cartes de poker pour influencer le déroulement de l'histoire et de potentiels mécaniques de reconnaissance vocale permettront des interactions dynamiques. Ces éléments renforcent l'expérience immersive

en ajoutant des aspects stratégiques et interactifs uniques. Il sera en effet possible, au cours de plusieurs moments clés du jeu, de participer à des parties avec des IA [8].

## 2.4- Intérêt du projet pour les joueurs et l'industrie vidéoludique

En s'inspirant de jeux comme Phasmophobia, Visage ou encore Granny, mais en y intégrant des mécaniques originales et remixées, We Don't See Sharp offre une expérience interactive et dans laquelle le joueur a une réelle incidence sur les événements. Le projet met l'accent sur la narration interactive où chaque choix des joueurs influence directement le scénario, renforçant l'idée que "vos choix ont des conséquences".



- Jeu d'horreur "Phasmophobia" -



- Jeu d'horreur "Visage" -

## 3- Communication et promotion

### 3.1 Réseaux Sociaux

Pour assurer la promotion de *We Don't See Sharp*, nous avons mis en place une **stratégie de communication digitale** en créant un compte Instagram dédié : **@flush\_hand\_studio**. Ce compte nous a servi de **plateforme centrale** pour partager régulièrement les avancées du projet, les coulisses du développement et les éléments clés du jeu.

À travers des **publications soignées**, nous avons exposé différentes étapes de la production, allant des premières maquettes aux mécaniques de jeu finalisées. Des **teasers visuels**, des courts extraits de gameplay et des annonces ont permis d'attiser la curiosité de notre audience tout en fidélisant une communauté autour du projet.

### 3.2 Site Web

Dans le cadre de ce projet, nous avons développé **un site web sur mesure** qui sert de **plateforme officielle** pour notre jeu. Ce site a été conçu comme un **espace complet et interactif**, rassemblant toutes les informations essentielles sur le projet :

- **Présentation détaillée du jeu** (concept, gameplay, univers)
- **Actualités et mises à jour** régulières
- **Une section "A propos"** mettant en valeur chaque membre et son rôle

- **Un blog de développement** partageant les coulisses de la création
- **Et surtout, une page de téléchargement** permettant d'accéder directement au jeu

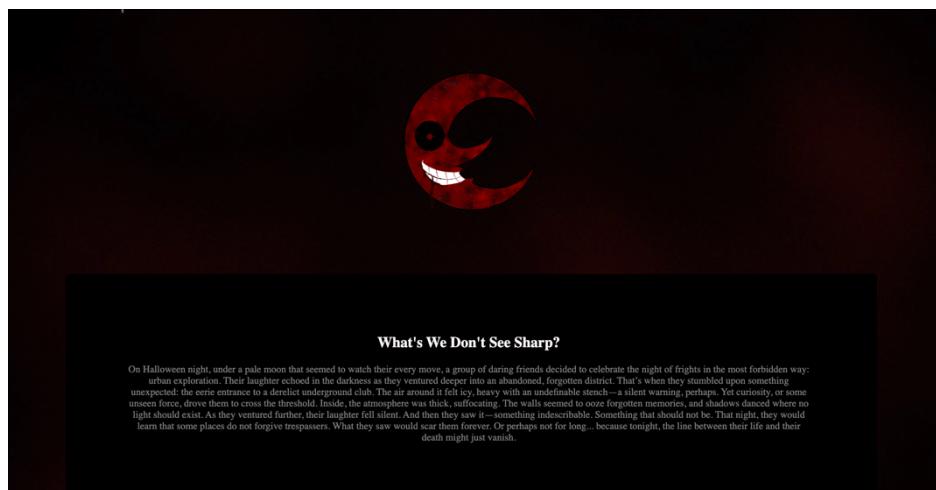
Ce dernier point est particulièrement important : nous avons voulu faciliter au maximum l'accès à notre création. **Le site propose ainsi le téléchargement du jeu** dans sa version finale.

La conception responsive garantit une **expérience utilisateur optimale**, que les visiteurs accèdent au site depuis ordinateur, tablette ou smartphone. Des liens vers nos réseaux sociaux et un formulaire de contact complètent l'ensemble, **favorisant l'interaction et les retours** de notre audience.

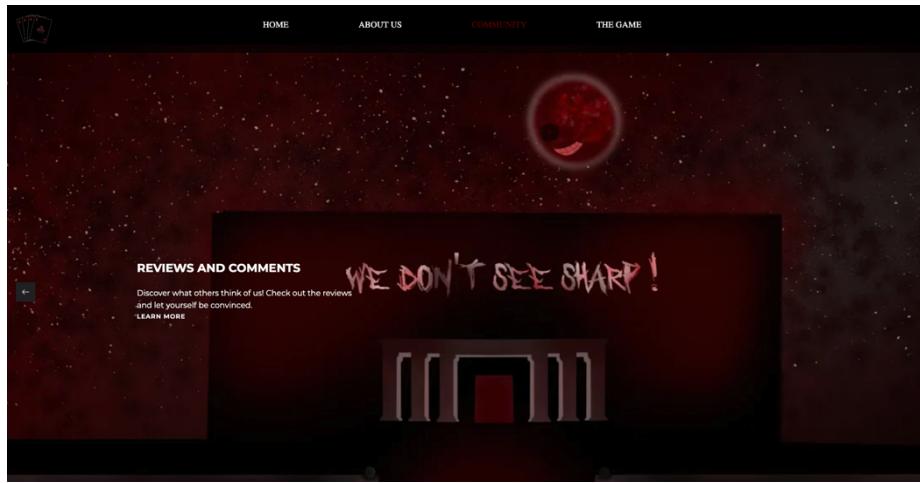
Cette initiative nous a permis de :

- **Maîtriser notre image** en ligne
- **Simplifier l'accès** à notre jeu

Le site reste régulièrement mis à jour pour refléter l'évolution du projet et maintenir l'intérêt des joueurs, faisant de lui bien plus qu'un simple outil promotionnel : **un véritable prolongement numérique de notre jeu.**



- Extrait de la page « Our Game » -



- Extrait de la page « Community » -

## 4- Organisation et méthodologie

### 4.1- Répartition des tâches

- Adrien GOMES : Principalement en charge des graphismes [12], du level design [13], de la conception des espaces, et des interactions entre joueur(s) et environnement. Il joue également un rôle dans l'assistance auprès de LAETITIA vis-à-vis des déplacements du joueur ainsi que de certains aspects esthétiques liés à la promotion du jeu.
- Laetitia BERKOUNE : Responsable spécialisée dans le développement web (création des sections du site, de son design et de l'ensemble de ses fonctions), elle gère également les assets du jeu permettant de faciliter le travail de conception des espaces et d'optimisation de l'espace mémoire du jeu. Elle apporte en même temps son soutien aux interactions environnementales à ADRIEN, ainsi qu'à DORIA et EMMA sur les aspects promotionnels du projet ou sur la conception des fonctionnalités relatives aux multiples IA [8][14].
- Doria BOUZIANE : Responsable de la conception des IA de poker [8] grâce notamment à son intérêt initial pour ce jeu de bluff, elle gère aussi en parallèle la promotion sur les réseaux sociaux et assistera dans la reconnaissance vocale ainsi que dans la mise en réseau finale de l'ensemble des composants du jeu (en particulier lors des phases de merging de scène entre membres). Elle intervient également dans le support des SFX[15] divers.
- Emma RUCAY : Membre particulièrement plongée dans le développement de l'intelligence artificielle, en particulier responsable pour les ennemis [14] et assistante sur les jeux de cartes [8]. Responsable des effets sonores et visuels, elle intervient également en très grande

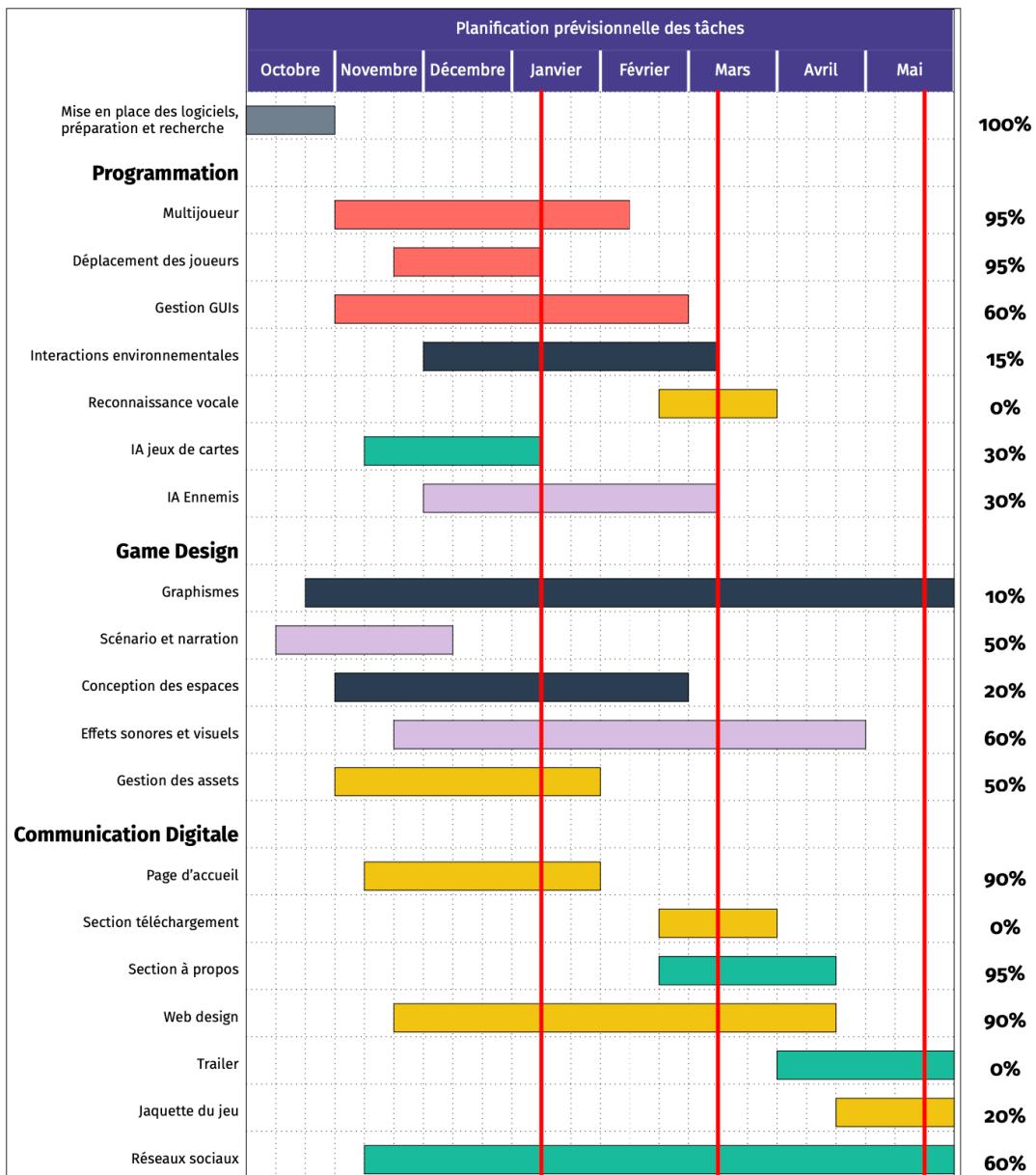
partie dans le scénario [16].

Tâches	Léo	Adrien	Laetitia	Doria	Emma
<b>Programmation</b>					
Multijoueur	R			S	
<b>Mécaniques de jeu</b>					
Déplacement des joueurs	R	S			
Gestion GUIs	R	S			
Interactions avec l'environnement	S	R	S		
Reconnaissance vocale			R	S	
<b>Intelligence artificielle</b>					
Jeux de cartes			S	R	S
Ennemis			S		R
<b>Game Design</b>					
<b>Style</b>					
Graphismes		R	S		
Scénario et narration	S	S			R
<b>Environnement</b>					
Conception des espaces		R	S		
Effets sonores et visuels				S	R
Gestion des assets	S		R		
<b>Communication Digitale</b>					
<b>Web Development</b>					
Page d'accueil			R	S	
Section téléchargement			R	S	
Section à propos			S	R	
Web design			R	S	
<b>Promotion</b>					
Trailer				R	S
Jaquette du jeu		S	R		
Réseaux sociaux (instagram)			S	R	

Répartition des tâches par membre de l'équipe.

**R** : responsable - **S** : suppléant(e)

## 4.2 Diagramme de Gantt (barres pour soutenances et taux avancement)



Couleur	Tâche	Responsable
Vert	IA jeux de cartes, Section à propos, Trailer, Réseaux sociaux	Doria
Orange	Multijoueur, Déplacement des joueurs, Gestion GUIs	Léo
Noir	Interactions environnementales, Graphismes, Conception des espaces	Adrien
Grise	IA Ennemis, Effets sonores et visuels, Scénario et narration	Emma
Jaune	Reco. vocale, Gestion assets, Page d'accueil - téléchargement, Jaquette du jeu	Laetitia

## 5- Réalisations techniques

### 5.1- Mécaniques de gameplay

#### Présenté en S1

Les mécaniques de gameplay déjà mises en œuvre intégraient plusieurs fonctionnalités essentielles pour enrichir l'expérience du joueur. Le système de mouvements des personnages, par exemple, offre des contrôles fluides et personnalisables, avec la possibilité d'ajuster la sensibilité des axes horizontaux et verticaux directement depuis les paramètres du jeu. Ce système assure une navigation intuitive dans l'univers du jeu tout en respectant les préférences du joueur.

#### Présenté en S2

Depuis la dernière soutenance, plusieurs nouvelles mécaniques de gameplay ont été implémentées afin d'enrichir l'expérience du joueur et d'améliorer l'interaction avec l'environnement principalement.

##### *Script lumière*

Pour accentuer l'atmosphère du jeu, un **script de gestion de l'éclairage** a été mis en place. Celui-ci permet de faire clignoter certaines lumières de manière dynamique, en contrôlant à la fois leur **intensité** et leur **fréquence**. Cette variation lumineuse apporte un effet visuel marquant, idéal pour instaurer un climat de suspense, voire d'angoisse, dans certaines zones du niveau. Ce système de lumière participe activement à l'ambiance générale du jeu, en jouant sur l'immersion du joueur et en renforçant les éléments de mise en scène.

#### Nouveautés de la soutenance finale :

##### *L'IA pathfinding*

Une intelligence artificielle de navigation a été mise en place dans le jeu, reposant sur un système de **pathfinding** (recherche de chemin) via le composant **NavMesh** de Unity. Ce système permet à l'ennemi de se déplacer de manière autonome dans l'environnement, en contournant les obstacles et en suivant le tracé le plus efficace pour atteindre sa cible. Le labyrinthe, construit manuellement, a été spécialement conçu pour offrir un parcours complexe et stratégique, renforçant ainsi l'intérêt du joueur à observer et anticiper les mouvements de l'ennemi. Cela ajoute une dimension tactique au gameplay, où chaque couloir ou détour peut devenir un élément clé de survie ou de confrontation.

### ***Mouvement de l'ennemi***

Afin de rendre le comportement de l'ennemi plus réaliste et mieux intégré à la progression du joueur, un script a été développé pour contrôler le déclenchement de ses déplacements. L'ennemi reste inactif tant que le joueur n'est pas présent dans la scène. Dès que sa présence est détectée, l'IA est activée, et l'ennemi commence à patrouiller ou à poursuivre le joueur selon les conditions définies. Cette approche permet non seulement d'optimiser les performances du jeu (en évitant les calculs inutiles en l'absence du joueur), mais elle contribue aussi à renforcer l'impact dramatique de son apparition, créant une montée en tension au bon moment.

### ***Touches de triches***

Dans le but de fluidifier la démonstration du jeu lors de nos présentations, nous avons décidé d'intégrer un système de **touches de triche**. Cette idée a d'abord suscité des débats au sein de l'équipe, car elle soulevait des questions d'équilibre, d'intégrité du gameplay et de faisabilité technique. Néanmoins, nous avons fini par convenir que, dans un contexte de test ou de démonstration, ces raccourcis seraient utiles pour mettre en avant les fonctionnalités essentielles du jeu, sans que le joueur ait à tout parcourir manuellement.

La mise en place de ces touches n'a pas été immédiate : elle a nécessité plusieurs jours de réflexion pour définir précisément leur utilité, leur fonctionnement et leur comportement dans différentes situations. Il nous a fallu près de deux semaines pour concevoir, tester et stabiliser chaque raccourci clavier, en nous assurant qu'ils ne créaient pas de bugs ou de conflits dans les scènes concernées.

Parmi les principales touches de triche, nous avons mis en place :

**Une touche spéciale dans le labyrinthe** permettant de téléporter le joueur directement derrière le monstre. Ce raccourci ne repose pas simplement sur des coordonnées fixes : il utilise une détection en temps réel de la position de l'ennemi, garantissant ainsi une cohérence avec le système d'IA. L'objectif est de montrer visuellement et rapidement que le système de pathfinding fonctionne correctement, sans attendre que l'ennemi atteigne une position spécifique.

**Une touche de téléportation simple** vers la fin du labyrinthe, permettant de passer directement à l'étape suivante pour gagner du temps en démonstration.

**Dans la scène de la maison**, une touche a été ajoutée pour avancer directement le décompte à zéro. Cela permet de simuler rapidement la fin d'un événement temporel sans avoir à attendre la durée complète prévue initialement.

Ces raccourcis ont été conçus pour être activables à tout moment du jeu, ce qui nous a obligés à anticiper leur interaction avec les différents états du joueur, de l'IA, et des scripts en cours d'exécution. Ils se sont révélés extrêmement pratiques pour tester des scènes spécifiques et pour rendre nos présentations plus dynamiques et efficaces.

Touche	Utilité
O	Téléportation vers la fin du labyrinthe
L	Téléportation vers le montre du labyrinthe
T	Incrémantion du temps dans collecte à 5s pour le maison du spectre

## 5.2- Graphismes et animations

### Présenté en S1

Les premiers jalons de notre identité visuelle ont été posés avec la création des deux logos fondamentaux : celui incarnant notre studio et celui spécifiquement dédié au jeu. Ces éléments graphiques fondateurs ont été imaginés comme les pierres angulaires de l'univers que nous souhaitions bâtir, établissant les codes visuels qui allaient ensuite se décliner à travers l'ensemble du projet.

Conçus dans un souci d'harmonie globale, ces visuels intègrent des subtilités graphiques qui évoquent l'atmosphère singulière de notre jeu.

Ces logos ne se contentent pas de remplir une fonction illustrative - ils incarnent l'essence même de notre projet. Celui du studio reflète notre philosophie créative, tandis que celui du jeu en condense l'esprit et l'univers. Leur conception rigoureuse a permis d'établir une base solide pour le développement ultérieur de tous les éléments visuels, garantissant une cohérence à chaque étape de la communication.



- Logo de We Don't See Sharp -



- Logo de Flush Hand Studio -

Dans le cadre de notre jeu, nous avons créé une collection exclusive de cartes de poker personnalisées, soigneusement pensées pour s'intégrer à l'expérience globale. Ces cartes ne se contentent pas d'afficher notre logo de manière discrète mais efficace - elles incarnent véritablement l'esprit de notre jeu à travers chaque détail de leur conception.

Leur design, à la fois épuré et sophistiqué, transpose avec élégance les principaux codes visuels de notre univers. Les motifs, les couleurs et les finitions ont été méticuleusement sélectionnés pour évoquer l'atmosphère particulière qui caractérise notre projet, tout en conservant la lisibilité essentielle à des cartes fonctionnelles.

Bien au-delà de leur rôle esthétique, ces cartes jouent un rôle narratif et mécanique central dans l'expérience de jeu. Elles servent de véritables pivots interactifs :

- Déterminant des tournants clés dans l'histoire
- Modulant les dynamiques entre joueurs selon les différentes phases de jeu

Chaque détail de ces cartes - de leur texture au choix des symboles - a été conçu pour renforcer l'immersion tout en servant la mécanique ludique. Elles représentent ainsi une synthèse parfaite entre forme et fonction, où le design rejoint la narration pour créer une expérience cohérente et mémorable.



- Roi de cœur -

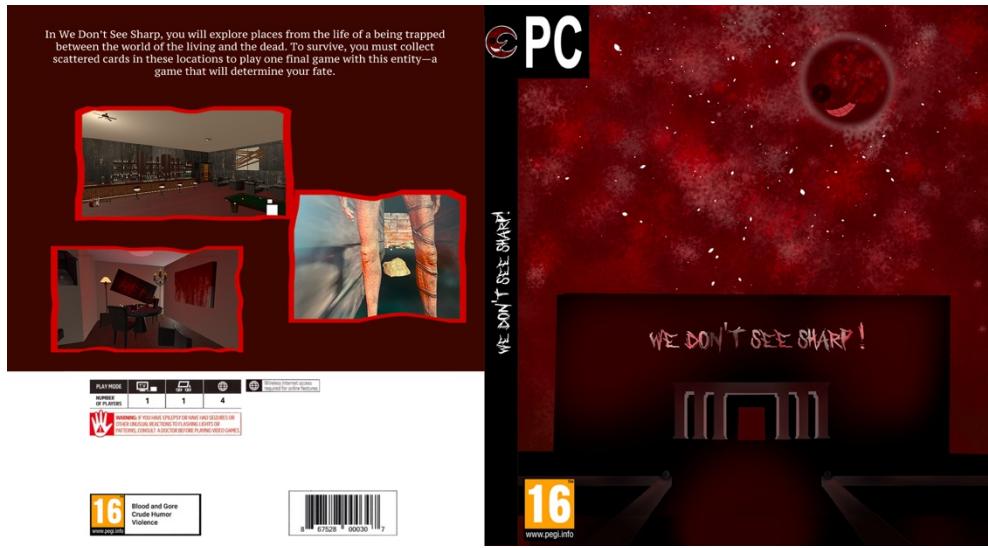


- As de trèfle -

Nous avons conçu une jaquette qui prolonge et renforce l'identité visuelle de notre jeu. En harmonie parfaite avec la direction artistique, elle reprend les couleurs emblématiques et les éléments graphiques clés qui caractérisent notre univers. La typographie soigneusement sélectionnée vient compléter ce tableau visuel cohérent.

Ce support physique a été pensé pour dépasser sa simple fonction d'emballage. Il sert d'ambassadeur à notre univers, offrant aux joueurs une première immersion tangible avant même le démarrage du jeu. Sa conception a nécessité un travail méticuleux pour garantir son impact visuel.

La jaquette devient ainsi un objet à part entière, un véritable pont entre notre communication externe et l'expérience de jeu. Chaque détail a été optimisé pour créer une continuité visuelle parfaite, témoignant de notre exigence de cohérence à chaque étape du projet. Elle incarne notre volonté d'offrir une expérience artistique unifiée, où même le support physique participe à l'immersion dans l'univers que nous avons créé.



- Jaquette de We Don't See Sharp -

Dans la continuité de cette identité visuelle soigneusement élaborée, nous avons également développé une charte graphique complète pour le site web, conçue spécifiquement pour prolonger et enrichir l'expérience du jeu bien au-delà du simple écran. L'univers visuel du site a été méticuleusement travaillé pour reprendre et adapter les principales teintes, les textures caractéristiques et les atmosphères distinctives qui font l'identité unique de notre jeu. Cette transposition créative permet d'assurer une parfaite cohérence esthétique à travers tous les supports de communication.

Ce travail graphique approfondi a joué un rôle central dans l'unification de l'ensemble de la communication autour de notre projet. Qu'il s'agisse du site web lui-même, des divers supports promotionnels ou même de l'interface du jeu, la même signature visuelle se déploie harmonieusement, créant ainsi une expérience cohérente et immersive pour les joueurs comme pour les visiteurs. La continuité ainsi établie entre ces différents éléments renforce considérablement l'identité globale de notre création.



- Page d'accueil du site de promotion de We Don't See Sharp -

Enfin, dès les débuts du développement, **les réseaux sociaux** ont été intégrés à notre stratégie de communication. Des visuels en lien avec le jeu, ses mécaniques ou ses personnages y ont été régulièrement publiés, dans le respect de la direction artistique que nous avions définie. Cela a permis de créer une communauté, de susciter de l'intérêt autour du projet, tout en renforçant la cohérence graphique de l'ensemble.

## Présenté en S2

Depuis la dernière soutenance, quelques avancées ont été réalisées dans la direction artistique du projet. L'accent a été mis sur le développement des environnements, l'amélioration des effets visuels et l'intégration d'éléments immersifs permettant de se projeter un peu plus au sein du jeu.

### **Développement des premiers environnements**

L'aspect visuel du labyrinthe a été soigneusement travaillé afin d'offrir une expérience immersive et cohérente avec l'ambiance générale du jeu. Les murs, sols et autres éléments décoratifs ont été créés ou sélectionnés pour renforcer le caractère mystérieux et parfois oppressant de l'environnement. Une attention particulière a été portée à la répétition des textures pour éviter toute monotonie visuelle. Des variations dans les matériaux et l'ajout de détails comme des fissures, des traces ou des éléments de végétation ont permis d'enrichir visuellement l'espace. L'éclairage a également été pensé pour guider subtilement le joueur tout en conservant une part d'exploration et de tension.

## Soutenance Finale

### *Club abandonné*

Cette zone ne se résume pas à un simple espace de transition ou de repos : elle constitue l'un des éléments structurants du jeu, à la fois point d'ancrage, carrefour décisionnel et repère émotionnel. Elle joue un rôle central dans l'expérience du joueur en assumant simultanément plusieurs fonctions essentielles.

Avant tout, elle agit comme un véritable **nexus de l'aventure**. Conçue comme un hub, elle relie entre elles toutes les régions explorables du jeu. Grâce à son organisation spatiale pensée avec précision — notamment un design circulaire et des embranchements multiples — cette zone permet une orientation claire et une continuité dans la progression. Elle devient ainsi une plaque tournante où convergent les différentes phases de gameplay, tout en matérialisant visuellement l'interconnexion entre les niveaux.

Mais son importance ne s'arrête pas là. Cette zone occupe également une place clé sur le plan narratif, jouant le double rôle de **point de départ et de destination finale**. Le joueur y commence son aventure, puis y revient à la toute fin du jeu, bouclant ainsi son périple de manière symbolique. Ce retour final, dans un lieu désormais chargé d'histoire et de souvenirs, confère une forte valeur émotionnelle à l'expérience. Il s'agit d'un choix de conception délibéré, visant à renforcer le sentiment d'accomplissement tout en créant une forme de boucle narrative cohérente et marquante.

Enfin, cette zone remplit une **fonction ludique et émotionnelle** importante. Elle offre au joueur des moments de répit bienvenus entre les séquences de tension. C'est un espace de respiration, de préparation et d'anticipation, où l'on peut faire le point avant de s'engager à nouveau dans l'inconnu. Sa familiarité progressive contribue à rassurer le joueur et à renforcer son attachement au lieu. De plus, en étant le théâtre de l'ultime confrontation, cette zone devient le vecteur d'un **climax émotionnel fort**, amplifié par tout ce qu'elle incarne au fil du jeu.



- Extrait du casino -

### ***Le labyrinthe***

Ce niveau a fait l'objet d'une attention particulière sur le plan visuel, afin de renforcer l'immersion du joueur et de créer un environnement cohérent, vivant et crédible. Chaque élément décoratif a été soigneusement sélectionné et placé pour servir l'atmosphère générale du jeu. Les murs, par exemple, ont été modélisés et texturés avec une volonté de réalisme, dans un style qui reflète l'identité artistique globale du projet. Des détails comme les fissures, l'usure ou la saleté ont été intégrés pour renforcer l'impression d'un lieu ancien, marqué par le temps.

En complément, de la végétation animée a été ajoutée à certains endroits du niveau. Bien que discrète, cette animation – simulant un léger souffle de vent sur des plantes ou des feuilles – insuffle une dynamique subtile à l'environnement. Elle permet de briser la rigidité d'un décor statique, en donnant au joueur l'impression que le monde qui l'entoure respire, évolue et existe indépendamment de ses actions. Ce type de détail, souvent imperceptible de manière consciente, joue pourtant un rôle important dans la sensation d'immersion.

Le travail ne s'est pas limité à l'environnement ; l'ennemi principal du niveau a également bénéficié d'un traitement visuel poussé. Sa modélisation, ses textures et sa posture ont été pensées en cohérence avec les lieux qu'il parcourt. Le design de cet antagoniste a été conçu pour s'intégrer naturellement dans le décor, sans rupture visuelle, afin de conserver une immersion constante. Son apparence participe à l'ambiance angoissante du niveau et complète efficacement le décor sans attirer une attention incohérente ou artificielle.

Dans l'ensemble, ce soin apporté à la cohérence visuelle et à l'animation de l'environnement contribue fortement à l'identité du niveau. Il favorise l'engagement du joueur, renforce la tension narrative et enrichit l'expérience de jeu en rendant le monde plus tangible et crédible.



- Extrait du labyrinthe -

### ***Studio d'enfance***

Pour ce niveau, l'intention principale était de créer un espace abandonné, sale et marqué par le temps, dans lequel toute trace de vie semble avoir disparu. L'ambiance visuelle repose sur des murs décrépis, des sols poussiéreux, et des objets disposés de manière chaotique, comme s'ils avaient été laissés en place à la hâte, ou déplacés dans un moment de panique. Ce cadre visuel contribue à installer une atmosphère pesante, dominée par la solitude, l'abandon et l'oubli.

Malgré sa superficie relativement restreinte, l'espace de jeu a été conçu avec une grande richesse de recoins et de cachettes, créant un terrain de jeu idéal pour la mécanique de recherche de la poupée, l'ennemi principal de ce niveau. Des zones d'ombre, des meubles renversés et des espaces partiellement dissimulés renforcent le sentiment de danger potentiel à chaque détour. Le joueur est ainsi confronté à une tension constante, ne sachant jamais réellement où la menace peut apparaître.

Chaque détail du décor a été pensé pour encourager l'exploration attentive, tout en maintenant un climat d'incertitude. Cette conception volontairement oppressante participe activement à l'expérience de jeu, en renforçant la peur de l'inconnu et la nécessité d'avancer avec prudence. L'environnement devient alors un acteur à part entière du gameplay, autant qu'un outil de narration visuelle.



- Extrait du studio d'enfance -

### ***Maison de John***

L'esthétique de la scène du studio a été pensée pour transmettre une atmosphère sombre, oppressante et marquée par un passé violent. Le choix s'est porté sur un environnement délabré, avec des sources de lumière faibles et dirigées, créant de forts contrastes et des zones d'ombre propices au suspense. Le mobilier a été disposé de manière volontairement chaotique : chaises renversées, étagères à moitié détruites, objets abandonnés, renforçant le sentiment de désordre et de tension.

Des éléments interactifs ont été placés de façon stratégique afin d'attirer l'attention du joueur sur des zones importantes, notamment celles où sont dissimulées les cartes à récupérer. Cette mise en scène visuelle guide subtilement le joueur sans recourir à des marqueurs trop explicites, renforçant ainsi l'immersion.

Par ailleurs, l'environnement du studio intègre des détails visuels liés à la scène de torture censée s'être déroulée dans ce lieu. Des traces de sang séché, des outils rouillés et du mobilier souillé ou brisé viennent enrichir la narration environnementale. Chaque élément a été pensé pour évoquer l'histoire du lieu sans dialogue, uniquement par l'observation, plongeant le joueur dans une ambiance lourde et angoissante, où chaque recoin peut porter les marques d'un événement passé.



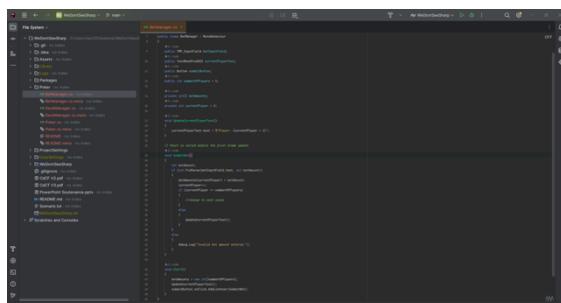
- Extrait de la maison de Jonh -

## 5.3 Intelligence artificielle

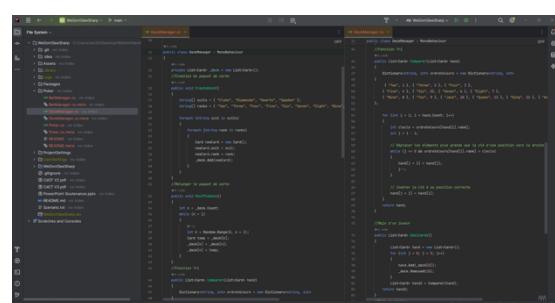
Présenté en S1

Le développement de l'intelligence artificielle avait avancé lors de la dernière étape de manière progressive, mais il avait déjà à l'époque rencontré certains retards en raison de la complexité technique et de la nécessité de coordonner cette partie avec d'autres aspects du projet. Ces retards étaient dus à des imprévus dans l'intégration des mécaniques. Cependant, quelques avancées avaient été réalisées sur des fonctionnalités spécifiques et essentielles (en majorité relatives au jeu de poker).

Pour cet aspect de l'IA donc, de nombreuses fonctions individuelles avaient été codées mais non implémentée en jeu. Cela incluait notamment la gestion du mélange des cartes, leur distribution équitable mais aléatoire aux joueurs, ainsi que la détection des mains de chaque joueur ou PNJ [6]. Ces fonctionnalités ont posé les bases du développement ultérieur, en veillant à rendre les adversaires réalistes et compétitifs, tout en offrant des interactions engageantes pour le joueur.



- Script de gestion des mises -



- Script de gestion du Deck -

Bien que l'IA pour les interactions avec les ennemis ou les événements paranormaux étaient encore en phase de réflexion et de prototypage, les avancées dans le domaine du poker avaient démontré une compréhension initiale des mécaniques nécessaires pour rendre le gameplay riche et interactif. Ces efforts permettront, à terme, de garantir une expérience immersive et cohérente dans l'ensemble des niveaux du jeu.

Présenté en S2

L'intelligence artificielle a connu plusieurs avancées, notamment sur le développement du pathfinding des ennemis. Ce travail est essentiel pour un niveau en cours de conception, où le joueur devra fuir une entité dans un labyrinthe.

## Développement du pathfinding et conception du labyrinthe

Un prototype de pathfinding a été mis en place en utilisant le NavMesh d'Unity, permettant aux entités ennemis de se déplacer dynamiquement dans un environnement complexe. Ce système est destiné à gérer les mouvements d'un ennemi poursuivant le joueur dans un niveau de type labyrinthe.

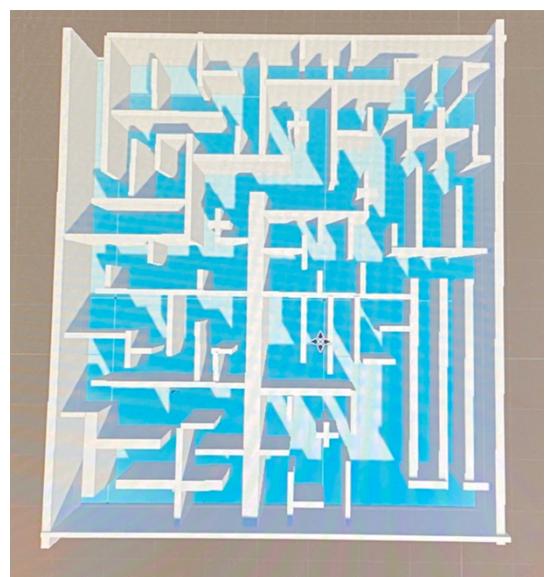
La première phase a consisté à modéliser le terrain, en positionnant les murs et les pièges du labyrinthe. Avant cette étape, plusieurs plans ont été dessinés afin d'obtenir un tracé équilibré en difficulté et challenge pour le joueur.

#### ***Correction des problèmes du NavMesh***

Lors de l'implémentation du NavMesh, plusieurs problèmes ont été rencontrés, notamment une mauvaise attribution des surfaces, ce qui empêchait l'IA de se déplacer correctement. Après plusieurs ajustements, le terrain fonctionne désormais correctement et est prêt à être utilisé pour les tests de gameplay.



***- NavMesh mal configuré -***



***- NavMesh bien configuré -***

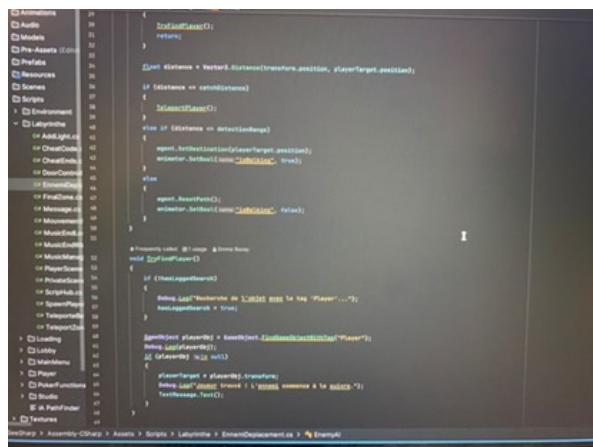
Ces avancées posent les bases du système de poursuite des ennemis, qui sera affiné au fil des prochaines semaines de développement. L'objectif final est en effet d'implémenter un comportement de traque dynamique, où l'ennemi pourra adapter ses mouvements et stratégies de poursuite en fonction des actions du joueur. Toutefois, ce développement est encore en phase d'expérimentation et nécessite plusieurs itérations avant son intégration définitive.

## Soutenance finale

### NavMesh

La version finale du système de navigation basé sur le NavMesh a été intégrée dans le jeu après plusieurs phases de tests et d'ajustements. L'une des principales améliorations apportées concerne son activation conditionnelle : un système de détection a été conçu pour s'assurer que le NavMesh ne se déclenche qu'à partir du moment où le joueur entre réellement dans la zone concernée. Cette approche permet d'éviter les erreurs d'activation prématuée, notamment pendant les phases de chargement de la scène, qui pouvaient entraîner des comportements inattendus de l'intelligence artificielle.

Ce système garantit ainsi une utilisation plus stable et cohérente des déplacements ennemis. L'IA ne s'active que lorsque les conditions de jeu sont réunies, assurant une meilleure synchronisation entre les événements et le positionnement réel du joueur dans l'espace de jeu. Cela contribue à renforcer l'immersion, tout en limitant les problèmes techniques liés à la navigation automatique. Grâce à cette optimisation, les séquences de poursuite ou d'interaction avec l'ennemi sont devenues plus fluides, naturelles et dynamiques.



The screenshot shows the Unity interface with the Hierarchy window on the left and the Script Editor on the right. The Hierarchy window lists various assets including Animations, Audio, Models, Pre-Assets, Prefabs, Resources, Scripts, and several folders like Environment, Labirinth, and Player. The Script Editor contains the following C# code:

```
private void Update()
{
    if (target != null)
    {
        float distance = Vector3.Distance(transform.position, playerTarget.position);
        if (distance <= detectionRange)
        {
            agent.SetDestination(playerTarget.position);
            animator.SetBool("InAttackZone", true);
        }
        else
        {
            agent.ResumePath();
            animator.SetBool("InAttackZone", false);
        }
    }
}

private void OnDrawGizmos()
{
    Debug.DrawLine(transform.position, playerTarget.position);
    Debug.Log("Distance de " + distance + " entre le tag " + tag + " et le tag " + target.tag);
    Debug.DrawLine(transform.position, target.position);
}
```

- Extrait du code de l'intelligence artificielle –

### IA Poker

Dans une partie de poker traditionnelle, la multiplicité des joueurs est essentielle : elle garantit à la fois la diversité des stratégies, le suspense des tours et l'équilibre global du jeu. Toutefois, dans le cadre de *We Don't See Sharp*, il nous fallait également penser aux cas de figure où le joueur se retrouve seul, ou dans une session multijoueur incomplète. Pour répondre à ce besoin, nous avons développé une intelligence artificielle dédiée – un joueur virtuel, surnommé « l'ordi de poker » – capable d'occuper l'une des places autour de la table.

Ce joueur automatisé est intégré de manière fluide dans le système. Il apparaît automatiquement lorsqu'il manque des participants pour démarrer une partie. Une fois en place, il se comporte exactement comme un joueur humain : il possède son propre deck,

reçoit ses cartes via le système de distribution, et agit durant les tours selon des règles logiques préprogrammées. Cela permet de maintenir un rythme de jeu constant, sans interruption ni déséquilibre.

#### **Caractéristiques techniques :**

- **Comportement défini par script** : L'IA suit des appels prédéfinis dans les scripts du jeu, ce qui lui permet de prendre des décisions en temps réel selon la situation (miser, suivre, se coucher).
- **Intégration complète avec Photon** : Le joueur IA est enregistré comme un véritable participant au sein d'une room Photon. Toutes ses actions sont synchronisées avec les autres clients, garantissant une cohérence réseau.
- **Interaction équitable** : Le bot respecte les règles du jeu, participe activement, et peut remporter une partie comme n'importe quel joueur humain, en fonction de ses cartes et des actions adverses.
- **Représentation visuelle à l'écran** : Pour renforcer l'immersion, l'ordi de poker est visible autour de la table avec un avatar et ses cartes (dans certaines configurations, comme le mode solo ou les tests internes).

#### **Objectif :**

L'objectif principal de cette IA est d'assurer une continuité de jeu en toute circonstance. Que le joueur soit seul, qu'il souhaite s'entraîner, ou simplement découvrir les mécaniques du jeu avant de rejoindre une session multijoueur, cette intelligence artificielle garantit une expérience fluide, accessible et sans temps d'attente. Elle renforce aussi l'accessibilité du mode poker, en le rendant jouable à tout moment, quels que soient les effectifs disponibles.



- Extrait d'une partie de Poker en jeu -

### 5.4- Détection des mouvements

#### **Présenté en S2**

Lors de la première soutenance, une version préliminaire d'un script de détection de mouvement a été présentée. Ce prototype constituait la première étape technique visant à capturer et analyser les gestes des participants durant la séquence interactive inspirée du jeu « 1, 2, 3, Soleil », mettant en scène une poupée inquiétante.

### **Le script avait pour vocation de :**

1. **Détecter en temps réel** les mouvements des joueurs.
2. Réagir lorsque les joueurs bougent pendant que la poupée les regarde ou qu'un joueur l'atteint.

### **Soutenance finale**

Pour le rendu final, le script initial a été entièrement repensé dès sa conception afin de satisfaire non seulement aux exigences spécifiques du niveau de difficulté, mais aussi pour pallier les lacunes du système de détection de mouvement qui s'était avéré peu fiable en pratique.

Une refonte complète du script a donc été implémentée. Cette nouvelle version permet désormais de capturer avec précision les déplacements des joueurs dès qu'un signal lumineux illumine l'intégralité de la zone de jeu.

Cependant, cette amélioration a mis en lumière plusieurs défis techniques inattendus. Les mécanismes de détection, bien que théoriquement solides, ont révélé des failles au cours des phases de développement avancé. Ces problèmes ont nécessité l'exploration et l'évaluation minutieuse de multiples approches alternatives. Parmi les méthodes testées, on peut citer :

- **L'analyse de la vitesse** du modèle 3D du joueur,
- **La comparaison différentielle** entre deux mesures successives de sa position dans l'espace virtuel.

Après une série de tests rigoureux, la seconde solution a été retenue pour son équilibre entre performance et précision. Néanmoins, cette méthode comporte des spécificités notables : la détection d'un mouvement pendant l'activation de la lumière intègre une **tolérance temporelle délibérée** de quelques millisecondes. Cette marge technique, bien que minime, a été conçue pour offrir une expérience plus équitable aux joueurs, évitant les faux positifs liés aux latences ou aux micro-mouvements involontaires.

```

119     }
120 
121     private void CheckPlayerDistance()
122     {
123         if (PhotonNetwork.IsMasterClient) { UpdateObject(); return; }
124 
125         foreach (var player in PhotonNetwork.PlayerList)
126         {
127             GameObject playerObj = PhotonNetwork.GetPlayerObject(player);
128             if (playerObj != null)
129             {
130                 UpdateMovementTracker(playerObj);
131 
132                 // Verification de la distance
133                 if (!IsScreenLight) { ShouldUpdatePlayer(playerObj); }
134                 else
135                 photonView.Awake("playerList", player);
136             }
137         }
138     }
139 
140     // Avertissement
141     private Dictionary<int, Vector3> lastPositions = new Dictionary<int, Vector3>();
142     private Dictionary<int, float> lastCheckTimes = new Dictionary<int, float>();
143 
144     private const float MOVEMENT_THRESHOLD = 0.0f; // Minimum movement distance to consider
145     private const float MIN_TIME_BETWEEN_CHECKS = 0.0f; // Prevent too frequent checks
146 
147     private Dictionary<int, MovementTracker> movementTrackers = new Dictionary<int, MovementTracker>();
148 
149     // Avertissement
150     private class MovementTracker
151     {
152         public Vector3 previousStartPosition;
153         public float movementDuration;
154         public float lastCheckTime;
155     }
156 
157     private void UpdateMovementTracker(GameObject player)
158     {
159         if (PhotonNetwork.View != player.GetComponent())
160             return;
161 
162         int playerID = view.GetInstanceID();
163 
164         if (movementTrackers.ContainsKey(playerID))
165         {
166             movementTrackers[playerID] = new MovementTracker();
167         }
168         else
169         {
170             var tracker = movementTrackers[playerID];
171 
172             if (IsScreenLight)
173             {
174                 if (view == null)
175                     return;
176 
177                 int player = view.GetInstanceID();
178                 if (movementTrackers.ContainsKey(player))
179                 {
180                     movementTrackers[playerID] = new MovementTracker();
181                 }
182             }
183         }
184 
185         // Mise à jour de l'ancienne position dans le cas où
186         // la distance > Vector3.Distance(tracker.previousStartPosition, player.transform.position)
187         tracker.previousStartPosition = player.transform.position;
188         tracker.movementDuration += value;
189     }
190 
191     private void ShouldUpdatePlayer(GameObject player)
192     {
193         if (IsScreenLight)
194         {
195             if (Vector3.Distance(lastPositions[player.ID], player.transform.position) > MOVEMENT_THRESHOLD || (lastCheckTimes[player.ID] + MIN_TIME_BETWEEN_CHECKS) < System.currentTimeMillis())
196                 lastCheckTimes[player.ID] = System.currentTimeMillis();
197         }
198     }

```

-Extrait du code de détection des déplacements -

## 5.5- Multijoueur

### Présenté en S1

Le mode multijoueur de We Don't See Sharp avait dès lors de la S1 atteint un stade pleinement opérationnel, offrant une synchronisation en ligne efficace grâce à l'intégration de Photon (21), une solution AP| (2) spécialisée pour les jeux multijoueur. Contrairement à une configuration LAN (23) traditionnelle, le jeu permet aux utilisateurs de se connecter facilement via Internet, garantissant une accessibilité accrue et une expérience fluide pour les joueurs, quel que soit leur emplacement. Cette mise en œuvre a marqué une étape clef du développement.

Le processus de connexion est géré par un script dédié, assurant la liaison initiale au serveur Photon (21) et dirige les joueurs vers un lobby commun une fois connectés. Ce système garantit une transition fluide et élimine les interruptions potentielles. D'autres scripts facilitent quant à eux la création et la gestion des salles de jeu, permettant aux joueurs de rejoindre ou de créer leurs propres sessions. Chaque salle offre un environnement cohérent pour les interactions multijoueur.

Une fois dans une partie, le programme assure automatiquement l'instanciation des avatars des joueurs sur le réseau. Chaque joueur étant équipé de contrôles locaux, avec une caméra dédiée et des menus spécifiques, tout en étant synchronisé avec les autres participants. Cette configuration garantit que les mouvements, interactions et actions locales soient visibles pour tous les joueurs en temps réel, renforçant la coopération et l'immersion dans le jeu.

De plus, des éléments comme la gestion des entrées des joueurs ou la synchronisation des interactions environnementales et des animations ont été soigneusement intégrés. Ces fonctionnalités assurent une expérience stable et immersive, avec un équilibrage optimal entre performance et précision réseau. Malgré les défis techniques, l'équipe a réussi à créer une base multijoueur solide, qui servira de socle pour les futures améliorations et ajouts qui demeureront sans aucun doute plus complexes et nombreux.

Ces avancées ont démontré l'engagement déjà présent fin 2024 pour fournir une expérience multijoueur de haute qualité, essentielle pour captiver les joueurs et garantir le succès du jeu.

Le multijoueur est non seulement une fonctionnalité opérationnelle, mais aussi un pilier central du gameplay collaboratif de We Don't See Sharp.

## Présenté en S2

Le mode multijoueur a bénéficié de plusieurs améliorations visant à optimiser la gestion des animations, la synchronisation des objets interactifs et la cohérence de l'environnement en réseau. Ces mises à jour garantissent une meilleure stabilité et fluidité du jeu en ligne.

Optimisation de la gestion des animations sur réseau.

Les animations des joueurs sont désormais synchronisées via un système simplifié de transmission de données en réseau. Plutôt que d'envoyer l'état de chaque animation en continu, elles sont gérées par un flux de requêtes en écriture/lecture, permettant de limiter la charge réseau et d'optimiser la réactivité des mouvements.

### ***Intégration et synchronisation des objets interactifs***

Tous les objets interactifs mentionnés dans la partie Mécaniques de Gameplay ont été intégrés dans l'environnement multijoueur. Ils sont instanciés dynamiquement et synchronisés entre les joueurs, garantissant une interaction fluide et cohérente, quel que soit le nombre de participants.

Gestion des états d'environnement et persistance des objets

Un système de mémoire des états a été mis en place afin d'assurer une cohérence globale des éléments du jeu en multijoueur. Chaque objet interactif essentiel dispose désormais d'un état enregistré en mémoire, permettant :

- De garantir la continuité des interactions, même en cas de déconnexion/reconnexion d'un joueur.
- D'assurer qu'un joueur rejoignant une partie en cours récupère immédiatement l'état actuel des objets et de l'environnement.

Ce fonctionnement repose sur l'utilisation de requêtes Remote Procedure Call (RPC), qui servent de mémoire tampon et assurent la transmission des états en temps réel dès qu'un nouvel utilisateur rejoint la session.

Ces mises à jour permettent d'améliorer significativement l'expérience multijoueur, en rendant les interactions plus naturelles et en assurant une gestion robuste des objets et de l'environnement en réseau.

## 6- Bilan intermédiaire

### 6.1- Avancées réalisées depuis la S1

Depuis la dernière soutenance, de nombreuses avancées ont été réalisées dans différents aspects du projet. Ces évolutions ont permis de consolider les fondations posées lors des premières étapes de développement. Le travail accompli a servi de lien entre les premières intentions de conception et la version finale présentée en mai. Ces progrès concernent aussi bien les mécaniques de jeu, l'optimisation des scènes, que l'enrichissement visuel et sonore, contribuant ainsi à une expérience plus complète et aboutie.

#### **Laetitia BERKOUNE**

- **Conception de la scène d'introduction**

Un travail a été réalisé sur la scène d'introduction du jeu, incluant les premières interactions du joueur ainsi que l'application de modèles 3D et de textures. Cette phase a posé les bases de l'univers visuel et interactif dès le lancement du jeu.

#### **Emma RUCAY**

- **Système de navigation IA avec le NavMesh**

Un système de navigation pour l'intelligence artificielle a été développé en utilisant le NavMesh d'Unity. Il permet à l'ennemi de se déplacer de manière autonome dans un environnement défini, en suivant une logique de pathfinding fluide et fonctionnelle.

- **Conception d'un niveau labyrinthe pour une séquence de poursuite**

Un niveau de type labyrinthe a été modélisé et intégré, spécialement pensé pour une

phase de poursuite avec l'IA. Sa structure favorise les détours, les points de tension, et soutient l'intensité du gameplay.

- **Correction de problèmes liés à l'application du NavMesh**

Des erreurs d'application du NavMesh ont été corrigées, notamment le mauvais calage sur certaines surfaces. Ces ajustements ont permis d'éviter des déplacements incohérents de l'IA, comme marcher sur les murs ou sortir des zones prévues.

### Doria BOUZIANE

- **Développement des cartes à récupérer et intégration dans le hub**

Un système a été mis en place pour permettre la détection et la récupération des cartes dans une scène, ainsi que leur transfert et leur affichage dans la scène du hub. Cela inclut le script de collecte et la gestion de leur persistance entre les scènes.

- **Contribution aux scripts liés au poker**

Une aide a été apportée sur la mise en place des scripts gérant la partie de poker, notamment pour le fonctionnement des mécaniques de jeu et la logique des interactions.

### Adrien GOMES

- **Conception d'un niveau inspiré de « 1 2 3 Soleil »**

Un niveau basé sur le principe du jeu « 1 2 3 Soleil » a été imaginé et développé. Il repose sur des mécaniques de mouvement contrôlé et d'observation, apportant une tension particulière et un rythme différent dans le déroulement du jeu.

- **Création d'éléments visuels en 2D**

Plusieurs illustrations et éléments graphiques en 2D ont été réalisés afin d'enrichir l'univers visuel du jeu. Ils ont servi à renforcer l'identité artistique et à appuyer certains aspects narratifs.

- **Développement d'un système de gestion de l'endurance**

Un système permettant de gérer l'endurance du joueur a été conçu et intégré. Il ajoute une contrainte supplémentaire de gameplay, obligeant le joueur à adapter ses actions en fonction de ses capacités physiques limitées.

## 6.2- Avancées réalisées depuis la soutenance S2

Laetitia BERKOUNE

- **Une mini-map contextuelle et immersive**

Une mini-map a été conçue pour guider les joueurs jusqu'au club *We Don't See Sharp*, point central de l'aventure. Intégrée de manière discrète, elle accompagne la progression dans les environnements complexes, tout en préservant la lisibilité de l'interface. Une fois la destination atteinte, la mini-map disparaît automatiquement, marquant symboliquement l'entrée dans une nouvelle phase du jeu et renforçant l'immersion dans la narration.

- **Ouverture de trappe synchronisée en multijoueur**

Le système permettant d'accéder à la salle secrète de poker via une trappe a fait l'objet d'une refonte complète. L'ouverture de cette trappe est désormais parfaitement synchronisée entre tous les joueurs d'une session multijoueur. Cela permet d'éviter toute désynchronisation de scène ou de gameplay, et favorise une transition fluide, cohérente et collective vers l'étape suivante.

- **Intégration du jeu de poker avec intelligence artificielle adaptative**

Le jeu de poker constitue un moment clé du scénario. Pour qu'il reste jouable en toutes circonstances, même avec un nombre réduit de joueurs, un joueur virtuel a été intégré sous forme de bot. Ce dernier possède son propre jeu de cartes, suit les mêmes règles que les autres participants, et intervient automatiquement dès qu'un joueur manque. Son comportement est entièrement géré par script, assurant ainsi une expérience fluide, équilibrée et sans interruption.

- **Système de gestion centralisée des scènes**

Un mécanisme global de contrôle de la progression a été mis en place pour organiser les transitions entre les niveaux et le hub central. Ce système détermine quelles portes sont accessibles en fonction de l'état d'avancement du joueur. Par exemple, une porte peut rester verrouillée si le niveau qui y est associé n'a pas encore été terminé, ou si la zone correspondante a déjà été complétée. Cette logique permet de maintenir un fil conducteur narratif et d'éviter les retours intempestifs ou incohérents.

- **Table de dépôt des cartes : un enjeu narratif et mécanique**

Une table spécifique a été installée dans le hub principal du jeu. Elle permet aux joueurs de déposer les cartes collectées tout au long des différents niveaux. Cette table ne se limite pas à un simple élément interactif : elle constitue un pivot scénaristique majeur. En effet, le nombre total de cartes déposées influencera

directement la fin du jeu, instaurant une logique de récompense fondée sur l'exploration et la coopération.

- **Trois fins distinctes selon les cartes récupérées**

Le scénario du jeu propose trois conclusions différentes, selon le nombre de cartes déposées dans le hub :

- **0 à 20 cartes** : L'échec collectif est acté, les joueurs n'ont pas réussi à remplir les conditions minimales.
- **21 à 45 cartes** : Une partie de poker est lancée, mais seul le joueur gagnant est considéré comme vainqueur.
- **46 à 52 cartes** : Tous les joueurs remportent la partie ensemble, incarnant une coopération réussie jusqu'au bout.

- **Ajout d'interactions et animations dynamiques**

Enfin, de nombreuses interactions ont été ajoutées avec l'environnement : manipulation d'objets (clés, leviers, portes), déclenchements de mécanismes, etc. Chacune de ces interactions est accompagnée d'une animation spécifique sur le joueur, afin de renforcer le réalisme et l'immersion. Ces ajouts participent à rendre le monde de *We Don't See Sharp* plus crédible, vivant et engageant.

## EMMA RUCAY

- **Finalisation de la scène du labyrinthe**

La scène du labyrinthe a été finalisée avec l'intégration de l'intelligence artificielle de l'ennemi, les scripts de déplacement, ainsi que l'ajout d'un script spécifique au joueur. Cette phase a permis de renforcer l'interactivité et la tension du niveau.

- **Réalisation des deux fins du jeu**

Deux fins distinctes ont été mises en place, chacune accompagnée d'une ambiance sonore adaptée. Elles apportent une conclusion narrative au jeu tout en renforçant l'impact émotionnel de l'expérience.

## Doria BOUZIANE

- **Création de la scène du studio**

La scène du studio a été conçue et finalisée avec l'intégration d'un système de décompte, de cartes à récupérer, d'un décor complet et d'une ambiance sonore. Cette phase a demandé une attention particulière à la cohérence visuelle et à la dynamique de jeu.

- **Implémentation des scripts liés à la scène du studio**

Tous les scripts nécessaires au bon fonctionnement de la scène du studio ont été développés et intégrés. Ils gèrent les interactions, les événements, ainsi que la logique du jeu dans cet environnement.

- **Modélisation de la scène finale**

La scène finale, dans laquelle les joueurs peuvent participer à la partie de poker décisive, a été modélisée et intégrée. Elle représente le point culminant du jeu, avec une mise en scène soignée pour marquer la fin de l'expérience.

### **Adrien GOMES**

- **Refonte du niveau de la poupée en 3D**

Le niveau de la poupée a été entièrement repensé et reconstruit dans un environnement 3D. Cette refonte a permis d'améliorer l'ambiance visuelle, la lisibilité de l'espace et l'intégration des mécaniques de gameplay.

- **Création graphique (jaquette et illustrations)**

Plusieurs éléments visuels ont été réalisés, dont la jaquette du jeu et certaines illustrations clés. Ces créations ont servi à renforcer l'identité graphique du projet et à le valoriser visuellement, notamment pour sa présentation.

### **Toute l'équipe**

- **Bonne cohésion de groupe**

Tout au long du projet, la cohésion du groupe s'est installée naturellement. La communication était claire, les échanges constructifs, et chacun savait se rendre disponible en cas de besoin. Cette entraide constante a permis de surmonter plus facilement les difficultés rencontrées, tout en maintenant une ambiance de travail agréable et motivante.

## **6.3- Défis rencontrés et retards, solutions et/ou explications**

### **Laetitia BERKOUNE**

- **Obstacles techniques et solutions mises en place : une courbe d'apprentissage formatrice**

Comme dans tout projet complexe, le développement de *We Don't See Sharp* a été jalonné de défis techniques, de problématiques de coordination et de zones d'inconfort liées à la prise en main d'outils professionnels. Ces difficultés, bien qu'ayant parfois ralenti notre progression, se sont révélées être des opportunités

précieuses de montée en compétence. Voici les principaux obstacles rencontrés, ainsi que les solutions que nous avons su mettre en place pour les surmonter collectivement.

- **Prise en main difficile de Git et gestion des versions**

L'un des premiers défis fut la gestion du versioning via Git. Si l'outil est incontournable dans le cadre du travail collaboratif, sa mise en œuvre s'est avérée plus complexe que prévu, notamment lors des premières tentatives de push depuis plusieurs ordinateurs. Des erreurs de synchronisation, des conflits de merge et des problèmes liés aux historiques divergents ont temporairement rendu le dépôt instable.

***Solution adoptée :***

Pour limiter ces blocages, nous avons opté pour une approche pragmatique : centralisation temporaire de certains fichiers sur un disque dur externe, partage local de matériel entre membres du groupe, et intensification des sessions de travail en présentiel. Cela a permis de limiter les conflits de version et de fluidifier les échanges de données. Ces ajustements nous ont permis de maintenir une progression constante, tout en continuant à utiliser Git à des fins de sauvegarde.

- **Découverte progressive de l'environnement Unity**

L'environnement Unity, bien que puissant, ne se laisse pas apprivoiser facilement par les néophytes. Les premières semaines ont été marquées par une incompréhension des logiques internes du moteur : hiérarchisation des objets, navigation dans la scène, utilisation des composants, gestion des animations... autant d'aspects qui demandaient un temps d'adaptation.

***Solution progressive :***

Un travail régulier, couplé à des recherches ciblées et des tests répétés, a permis une amélioration nette de notre maîtrise du moteur. Bien que certaines notions avancées restent complexes, l'apprentissage par la pratique nous a progressivement rendus autonomes sur les tâches essentielles au bon déroulement du projet.

- **Intégration complexe du multijoueur via Photon**

L'intégration du multijoueur grâce à Photon a constitué l'un des plus gros chantiers techniques du projet, notamment pour la partie poker. Nous avons été confrontés à un manque de documentation claire, des messages d'erreurs souvent peu explicites, et à la difficulté de déboguer des scripts complexes liés à la synchronisation des actions entre les joueurs.

***Conséquence directe :***

Certaines fonctionnalités prévues – comme la synchronisation parfaite des animations autour de la table de poker – n'ont pas pu être finalisées dans les temps. Néanmoins, le cœur du gameplay multijoueur a été conservé, et les éléments essentiels ont été intégrés de manière stable.

**• Défis liés au travail en équipe et à l'absence d'assistance technique ponctuelle**

À certains moments clés, des difficultés sont également survenues du fait de l'indisponibilité technique d'un membre du groupe. Cette absence de relais ou d'assistance dans l'intégration du multijoueur a temporairement ralenti l'avancement de certaines fonctionnalités.

***Résolution apportée :***

Face à ces blocages, nous avons choisi de ne pas abandonner ces fonctionnalités mais de redoubler d'efforts dans l'analyse des scripts et des erreurs rencontrées. Cette démarche, bien que chronophage, nous a permis d'acquérir une autonomie nouvelle et de débloquer des éléments critiques, assurant une version multijoueur fonctionnelle malgré les contraintes.

**Emma RUCAY**

**• Problèmes avec le NavMesh**

L'implémentation du système de pathfinding s'est révélée plus complexe que prévu, en grande partie à cause d'une mauvaise attribution des calques de navigation dans Unity. Concrètement, l'ennemi contrôlé par l'intelligence artificielle ne reconnaissait pas toujours la bonne surface sur laquelle il était censé se déplacer. Cela entraînait des comportements imprévisibles ou des blocages complets. Pour corriger ces erreurs, plusieurs ajustements ont été nécessaires dans les paramètres du NavMesh, ainsi qu'une phase d'expérimentation soutenue pour tester différentes configurations jusqu'à parvenir à un fonctionnement stable et réaliste du déplacement de l'IA.

**• Retards liés au dépôt Git**

Des difficultés liées à l'utilisation de Git ont également eu un impact négatif sur le développement, en particulier sur l'intégration du travail réalisé autour de l'intelligence artificielle. Certains fichiers, pourtant modifiés en local, ne parvenaient pas à être poussés correctement sur le dépôt partagé, ce qui a entraîné des désynchronisations entre les différentes versions du projet. Il a fallu réaliser plusieurs vérifications et synchronisations manuelles afin de rétablir un état cohérent du projet pour tous les membres de l'équipe, ce qui a engendré des pertes de temps non négligeables.

- **Prise en main complexe de Unity**

L'environnement de développement Unity a représenté un autre défi important, notamment pour les membres du groupe qui le découvraient. La logique de travail propre à ce moteur, qu'il s'agisse de l'organisation des scènes, de la configuration des objets ou de la gestion des animations, ne s'est pas révélée intuitive au départ. Cette courbe d'apprentissage initialement abrupte a ralenti la progression lors des premières phases de production. Cependant, à force de pratique et de recherche, une meilleure compréhension des outils et de leurs interactions a été acquise, permettant une montée en compétence progressive au fil du projet.

## Doria BOUZIANE

- **Difficultés techniques avec les préfabs dans Unity**

L'utilisation des préfabs dans Unity, bien que très utile pour structurer et réutiliser des éléments de jeu, a généré plusieurs complications. Un des problèmes récurrents concernait le fait que les modifications apportées à un préfab ne se répercutaient pas toujours correctement dans toutes les scènes où celui-ci était utilisé. Cette incohérence a parfois mené à des bugs visuels ou fonctionnels dans le jeu, rendant certains objets partiellement opérationnels ou invisibles. Pour corriger ces dysfonctionnements, il a fallu reprendre manuellement plusieurs objets dans différentes scènes, afin de vérifier leur bon affichage, leur comportement et leur bon lien avec la version source du préfab. Ce travail, chronophage, a mis en lumière la rigueur nécessaire dans la gestion des assets partagés et l'importance de tester systématiquement les modifications sur l'ensemble des scènes concernées.

- **Problèmes avec Git (commits et merges)**

La gestion de version avec Git a également soulevé des difficultés, particulièrement lors des étapes de merge entre différentes branches. Les conflits ont été fréquents, notamment sur les fichiers de scène qui, du fait de leur structure propre à Unity, sont particulièrement sensibles aux modifications simultanées. Dans plusieurs cas, des parties du projet ont été accidentellement écrasées ou perdues à cause de mauvais merges ou de résolutions de conflit incomplètes. Il a parfois été nécessaire de restaurer manuellement certaines versions antérieures, voire de recommencer certaines tâches, ce qui a généré des pertes de temps importantes. Ces problèmes ont souligné la nécessité de communiquer efficacement avant chaque push et merge,

ainsi que l'importance de bonnes pratiques en matière de gestion collaborative sur Git.

## Adrien GOMES

### • Difficultés lors des tests du niveau du studio d'enfance

Le développement du niveau centré sur la poupée s'est heurté à de nombreux problèmes techniques lors des phases de test. Plusieurs fonctionnalités prévues ne réagissaient pas comme attendu, ce qui a nécessité une réécriture complète de certaines fonctions. La stabilité de l'ensemble était difficile à garantir, avec des erreurs parfois difficiles à reproduire ou à localiser. Chaque modification en entraînait de nouvelles, ce qui a engendré un important travail de débogage. Cette instabilité a retardé la validation du niveau, rendant les tests plus longs et plus complexes que prévu.

### • Manque d'inspiration pour les illustrations

La phase de création artistique a été particulièrement délicate. Il a été difficile de trouver une direction visuelle convaincante : les idées proposées initialement manquaient de cohérence ou ne parvenaient pas à générer l'ambiance souhaitée. Malgré plusieurs tentatives, le rendu ne correspondait pas aux attentes artistiques du projet. Ce manque d'inspiration a freiné la progression et entraîné des remises en question sur la direction artistique à adopter.

### • Aspect chronophage de la création d'illustrations

La production des illustrations destinées à enrichir cette scène a demandé un temps conséquent. Chaque élément visuel nécessitait des ajustements précis et de nombreuses retouches pour s'intégrer harmonieusement au reste de l'environnement. Cette exigence qualitative a rendu le processus particulièrement long, d'autant plus que chaque changement artistique pouvait remettre en cause des choix antérieurs, prolongeant encore les délais.

### • Difficultés liées à la prise en main de Unity

L'utilisation de Unity a représenté un défi supplémentaire. La prise en main du moteur s'est avérée complexe, notamment en ce qui concerne l'organisation des assets, la gestion des scripts et la manipulation des animations. Plusieurs erreurs ont été commises à cause d'une compréhension incomplète des fonctionnalités du moteur, ce qui a obligé à revenir fréquemment sur certaines parties du niveau. Cette courbe d'apprentissage, bien qu'enrichissante à long terme, a fortement ralenti l'avancée du projet au départ.

## 6.4- Comparaison des progrès par rapport aux objectifs initiaux Objectifs atteints

À l'échelle du groupe, le bilan que nous tirons de cette phase de production est résolument positif. Les avancées réalisées ont été globalement conformes aux objectifs que nous nous étions fixés dès le départ, tant en termes de contenu que de qualité. Si quelques retards ont pu être constatés à l'occasion de la deuxième soutenance, ils n'ont pas compromis la dynamique du projet. Nous avons su faire preuve de réactivité en réévaluant nos priorités et en réorganisant notre planning de travail de manière efficace. Cette capacité d'adaptation a permis de rattraper rapidement le temps perdu, sans sacrifier la qualité des livrables.

La gestion du projet s'est appuyée sur une répartition claire des responsabilités, une communication continue entre les membres de l'équipe, et un suivi rigoureux de l'avancement. Ces éléments ont joué un rôle déterminant dans notre capacité à finaliser le jeu dans les délais impartis. Chaque étape du développement, qu'elle soit technique, artistique ou conceptuelle, a été menée avec le souci de produire un résultat cohérent, stable et fidèle à notre vision initiale.

La version finale du jeu que nous présentons aujourd'hui est pleinement fonctionnelle et reflète l'ensemble du travail accompli. Elle illustre les nombreux aspects du projet : mécaniques de gameplay, environnement graphique, ambiance sonore, narration, et ergonomie de l'expérience utilisateur. Plus encore, elle témoigne de l'engagement de chacun tout au long de cette aventure collective. Le résultat obtenu est le fruit d'une attention constante aux détails, de nombreuses itérations et d'une volonté partagée d'aller au bout de ce que nous avions entrepris.

Ce projet nous a permis non seulement de mettre à l'épreuve nos compétences, mais aussi de mesurer l'importance de la rigueur, de l'écoute et de la collaboration dans un cadre de production complexe. Il constitue, à nos yeux, une réussite à la fois sur le plan technique et humain.

## 7- Conclusion

La rédaction de ce rapport marque bien plus qu'une simple formalité académique : elle symbolise une étape charnière dans l'évolution du projet *We Don't See Sharp*, une synthèse concrète de tous les efforts investis, des difficultés surmontées et des connaissances acquises au fil des mois. Ce document témoigne à la fois du chemin parcouru et de l'ampleur du travail accompli, que ce soit sur le plan créatif, technique ou humain.

Le projet tel qu'il se présente aujourd'hui reste fidèle à l'ambition que nous avions formulée dès les premières réflexions. Il traduit avec justesse notre vision initiale, tout en s'étant enrichi d'idées nouvelles, d'améliorations continues et d'une meilleure compréhension des enjeux liés à la conception d'un jeu vidéo complet. L'équilibre atteint entre l'aspect artistique, le gameplay et la cohérence narrative est le fruit d'un travail rigoureux, d'un investissement constant et d'une capacité à s'adapter face aux imprévus.

Au-delà du résultat visible dans le jeu final, c'est aussi tout le processus qui nous rend fiers : les choix de direction, les corrections apportées en cours de route, les remises en question constructives et les nombreuses itérations qui ont permis à l'expérience de s'affiner peu à peu. Cette aventure de développement a été l'occasion non seulement de mettre en œuvre des compétences déjà acquises, mais aussi d'en acquérir de nouvelles – parfois dans l'urgence, parfois dans l'exploration – et de progresser ensemble, dans un cadre de collaboration exigeant mais formateur.

Chaque difficulté rencontrée a été perçue comme une opportunité d'apprentissage, chaque défi relevé comme une preuve de notre capacité à avancer collectivement vers un objectif commun. Ce projet nous a appris à structurer notre travail, à gérer nos ressources, à communiquer efficacement, et surtout à donner corps à une idée jusqu'à sa concrétisation. Ce que nous en retirons va bien au-delà du jeu lui-même : c'est une expérience complète, riche en enseignements, et formatrice à tous les niveaux.

En refermant cette phase du projet, nous conservons le sentiment fort d'avoir mené à bien une aventure complexe et ambitieuse, avec pour moteur principal la passion du jeu vidéo et le plaisir de créer ensemble un univers cohérent, marquant et engageant.

### ***Site Web***

Le projet possède également son propre site web accessible avec l'URL suivant :

<https://bit.ly/WeDontSeeSharp->

Le site propose des sections de références au projet, de téléchargement du jeu ou encore des sections de documentations.

### ***Leçons apprises***

- Vue de l'interface du site web -

Le développement de We Don't See Sharp a permis à l'équipe d'acquérir de nombreuses compétences et de tirer des enseignements précieux :

- Gestion de projet : L'importance d'une méthodologie agile pour coordonner les efforts de l'équipe, prioriser les tâches et s'adapter aux imprévus a été confirmée. La plateforme Trello [5] s'est révélée efficace pour attribuer les tâches et suivre leur avancement.
- Collaboration : Les défis techniques et créatifs ont mis en lumière la nécessité d'une communication constante et claire entre les membres de l'équipe, favorisant un partage d'idées constructif. Ce qui a parfois du mal à être intégré réellement au sein du groupe.
- Technologies : L'apprentissage et la maîtrise d'outils tels que Unity3D [1], Blender [29], et Photon [21] ont renforcé les compétences techniques des membres s'étant le plus investis.
- Résolution de problèmes : Les défis rencontrés, comme les bugs liés au multijoueur ou les difficultés dans la modélisation 3D, ont permis de développer des compétences en recherche de solutions et en itération rapide.

### ***Appports pédagogiques***

Ce projet a offert une opportunité exceptionnelle de mettre en pratique les compétences techniques et organisationnelles acquises au cours de notre formation :

**⇒ Programmation avancée : L'utilisation de C# pour les scripts et le développement des IA a permis de consolider nos bases en programmation tout en explorant des concepts avancés comme le pathfinding [30] ou encore la synchronisation et les requêtes réseau.**

**⇒ Design de jeu : La conception des niveaux et des mécaniques de gameplay a offert une perspective enrichissante sur l'importance de l'expérience utilisateur et de l'esthétique dans le développement vidéoludique.**

**⇒ Innovation : La réflexion visant à l'intégration de nos idées en jeu, comme les interactions avec l'environnement, des interfaces ou sur le fonctionnement pur de système intégré comme l'IA, a stimulé notre créativité et notre capacité à réfléchir efficacement.**

## Annexes

Cahier des charges fonctionnel

Contraintes techniques imposées et/ou retenues :

- Plateforme cible : PC sous Windows.
- Moteur de jeu : Unity3D [1] avec programmation en C#.
- API multijoueur : Photon [21].
- Optimisation des performances : gestion de la mémoire, utilisation de LODs [31] pour les modèles 3D.

Ressources utilisées Outils et Logiciels

- \* Unity3D : Utilisé pour le développement global (jeu, IA, multijoueur).
- \* Blender : Modélisation et animation 3D pour les environnements et objets interactifs.
- \* Photon : Gestion du multijoueur en ligne.
- \* Photoshop : Retouche graphique pour les textures et les interfaces.
- \* GitHub Desktop : Outils de mise en réseaux des fichiers pour programmer de manière collaborative.
- \* Make Human : Logiciel de création de personnage 3D avec des modèles définis.
- \* Mixamo : Ressource utilisé afin de concevoir des animations applicables aux modèles de joueur.

Licences

- \* Unity3D : Version étudiante gratuite.
- \* Photoshop : Licence annuelle à 232,84 €.
- \* Jetbrains Rider / VS Code : Version étudiante gratuite.
- \* Photon : Licence gratuite pour les petites équipes.
- \* Chat-GPT 4-o : Licence Plus mensuelle à 20 \$.
- \* GitHub : Version étendue en mémoire pour étudiants gratuite.

Bibliographie

- \* Tutoriels Unity : Documentations officielles et vidéos en ligne (YouTube).
- \* Blender : Guides en ligne pour modélisation 3D (forum, Chat-GPT, YouTube...). \*
- Photon API : Documentation pour développement multijoueur sur le site même.

## Références

- [1] Moteur de jeu polyvalent permettant de créer des jeux en 2D et 3D. Utilisé pour le développement de We Don't See Sharp, il offre des outils pour le rendu graphique, les animations, la physique, et les interactions en temps réel.
- [2] Date actuelle du projet : 26 Mai 2025.
- [4] Plateforme de gestion de versions et de collaboration pour le développement de logiciels. Permet de suivre les modifications du code, de collaborer en équipe et de gérer les dépôts (repositories). Utilisé pour centraliser les fichiers et les scripts du projet.
- [5] Trello est un outil de gestion de projet basé sur la méthode Kanban, qui repose sur l'utilisation de colonnes et de cartes pour visualiser et organiser les tâches à réaliser, leur progression et leur achèvement.
- [6] Accronyme : Personnage Non-Joueur (personnage du jeu ayant une apparence semblable à celle du véritable joueur ou bien simplement une entité indépendante mais qui n'est pas directement contrôlable par le joueur, donc soit par une IA soit par un comportement alternatif et/ou régulier).
- [7] Univers fictif d'horreur basé sur un concept d'espaces interminables, labyrinthiques, souvent associés à une ambiance oppressante et des dangers inconnus. Inspiré des creepypastas et utilisé comme cadre dans certains jeux d'horreur.
- [8] Accronyme : Intelligence artificielle (ayant la capacité de participer à une partie de poker).
- [9] Environnements fonctionnels mis en place à ce jour : GitHub (service de mise en réseau de fichiers partagés) et Trello (KANBAN : système de gestion méthodologique agile par attribution de tâches spécifiques).
- [10] Documents réalisés à ce jour : CDCT (cahier des charges technique), CDCF (cahier des charges fonctionnel), PDS (plans de soutenance), et RDS1-2 (rapport de soutenance S1 et S2).
- [11] Accronyme : Graphical User Interface (éléments visuels interactifs) ≈ UI (User Interface).
- [12] Ensemble des éléments relatifs : Menus (fonds, boutons, icônes...), Interfaces (Inventaires, Barre de vie, Barre de stamina...), et Textures (dessins, cartes, éléments de décor...)
- [13] Art de concevoir et d'adapter les espaces du jeu afin de refléter la vision des concepteurs, tout en préservant l'autonomie et la liberté du joueur, subtilement guidé par des détails et des mécanismes ingénieux qui orientent son exploration vers les objectifs souhaités.

[14] Acronyme : Intelligence artificielle (ayant la capacité de participer de se déplacer en autonomie (pathfinding) et d'interagir avec le joueur (dégâts, conversations...))

[15] Acronyme : Sound Effects (effets sonores utilisés dans un jeu vidéo pour renforcer l'immersion, comme des bruits d'environnement, des interactions, ou des sons liés aux actions du joueur ou des entités du jeu).

[16] Élaboration d'une vision globale de la trame narrative et des éléments de gameplay à développer, en harmonie avec le Lore, qui enrichira et complétera l'histoire pour renforcer son caractère immersif.

[17] Sauvegarde de données intrinsèque à l'ordinateur sur lequel le jeu est sauvegardé, même après suppression du jeu. Accessible depuis l'éditeur de registre et modifiable par valeurs binaires à l'emplacement :

HKEY\_CURRENT\_USER\Software\Unity\UnityEditor\FlushHandStudio\WeDontSeeSharp.

24

[18] Composant de Unity permettant de simuler la physique d'un objet 3D ou 2D. Il permet à un objet d'être affecté par la gravité, les forces et les collisions avec d'autres objets physiques. Utilisé notamment pour les déplacements réalistes des entités dans un jeu.

[19] Composant Unity permettant de gérer le déplacement d'un personnage sans utiliser directement la physique du moteur. Contrairement au Rigidbody, il ne subit pas les forces comme la gravité par défaut et offre un contrôle plus précis sur les mouvements du joueur.

[20] Technique utilisée en programmation de jeux pour envoyer un rayon invisible depuis un point donné dans une direction spécifique. Permet de détecter les objets qu'il rencontre, souvent utilisé pour les interactions avec l'environnement ou les mécaniques de tir.

[21] API de mise en réseau spécialisée pour les jeux multijoueur. Elle permet de gérer la communication en temps réel entre les joueurs, la création de salles (rooms), et la synchronisation des interactions dans des environnements en ligne.

[22] Acronyme : Application Programming Interface (ensemble de règles, de protocoles et d'outils qui permet à différentes applications ou logiciels de communiquer entre eux en définissant les moyens par lesquels les composants logiciels interagissent et offre des fonctionnalités préétablies pour simplifier le développement et l'intégration d'applications, comme la mise en réseau dans un jeu vidéo par exemple).

[23] Acronyme : Local Area Network (réseau local permettant à des ordinateurs ou appareils situés à proximité de communiquer entre eux. Dans le contexte des jeux vidéo, un LAN est souvent utilisé pour des parties multijoueur locales, ne nécessitant pas d'accès à Internet).

[24] Acronyme : Remote Procedure Call (appel de procédure à distance). Utilisé dans les jeux multijoueur pour exécuter une fonction sur plusieurs clients ou serveurs, permettant de synchroniser des actions entre joueurs via le réseau.

[25] Acronyme : Heads-Up Display (affichage tête haute). Interface graphique visible en permanence sur l'écran, affichant des informations essentielles au joueur comme la barre de vie, la jauge d'endurance, la carte, ou les objectifs.

[26] Zone invisible associée à un objet ou un personnage dans un jeu vidéo, utilisée pour détecter les collisions. Les hitboxes servent à déterminer si un projectile touche un ennemi, si un joueur subit des dégâts, ou si une interaction est possible avec une interface par exemple.

[27] Terme utilisé en programmation et dans Unity pour créer dynamiquement des objets pendant l'exécution du jeu. La fonction Instantiate() permet de générer des copies d'objets ou de préfabriqués (Prefabs) dans une scène.

[28] Processus par lequel plusieurs éléments d'un jeu (joueurs, objets, animations, variables, etc.) sont maintenus à jour et cohérents entre différents clients en multijoueur. Dans Unity, cela se fait via Photon, des RPC, ou des scripts de mise à jour réseau.

[29] Logiciel open-source de modélisation, d'animation et de rendu 3D. Utilisé pour créer des assets graphiques, des environnements et des animations pour les jeux vidéo. Il offre une flexibilité et une puissance adaptées aux projets professionnels comme amateurs.

[30] Technique utilisée en informatique et en développement de jeux vidéo pour trouver le chemin optimal entre deux points dans un environnement, souvent utilisé pour les déplacements des IA dans un jeu.

[31] Acronyme : Level of Detail (Technique d'optimisation graphique qui adapte dynamiquement la qualité des modèles 3D en fonction de la distance par rapport à la caméra, afin d'améliorer les performances sans compromettre la qualité visuelle perçue).