3.5 L’Intelligence Artificielle Premièrement,

Afin de me familiariser avec le moteur de jeu unity, nous avons suivi diffèrent tutoriel de l’application. Cela a été très important pour la création de l’intelligence artificielle, car en plus de nous apprendre les notions élémentaires de unity comme les nœuds et les scènes, cela nous a également permis d’avoir un espace de tests pour implémenter les nouveautés de l’IA à petite échelle et également de nous inspirer sur diffèrent ennemies possible. Cela fut d’une grande utilité étant donné que le projet devint jour après jour de plus en plus lourd et cela nous a en outre permis d’isoler plus facilement les problèmes rencontrés et en mettant en évidence les différents éléments et fonctionnalités de l’espace de travail

Une image contenant capture d’écran, jeux

Description générée automatiquement

En ce qui concerne les avancements techniques de l’intelligence artificielle, nous avons tout d’abord implémenté la composante principale de nos différent IA qui est le pathfinding en effet dès que le joueur entre dans le champs de vision de l’enemie ce dernier se mets a la chasser ou a effectuer son attack tout dépend de l’enemie

En effet il ya a dispotition 5 enemie different

Le volant :qui vole et des qu’il vois le joueur il lance un projetctile qui se case et mets des dégâts a l’enemie avec une fonction take damage

La tourelle : si le joueur est a bonne disctante il tire un prjetctile

Le block tombant : des que le joueur passe en dessous du joueur il tombe sur le joueur

Et l’oonde de chock : lorsque l’enemie entre dans sa zone de vision il lance une onde de chock

Toute c’est fonctionnalité seront permis par le nev mesh surface , qui va agir comme un calque du sol de la map et permettra d’indiquer à l’IA où elle peut se déplacer

Le principe pour l’instant des ia est assez simple en effet ils vont juste poursuivre le joueur et lorsque il le perde de vu et bien le bot reprend son pattern aléatoire cree grâce a la fonction patrol et reprend sa vitesse initiale

(Troisièmement, suite à l’implémentation du multijoueur, il fallait désormais que l’IA puisse suivre les deux joueurs, et il fallait donc que l’IA choisisse une cible à suivre. Nous avons fait le choix que l’IA suive le joueur le plus proche, ce qui nous semblait le plus logique.)

Deuxièmement, nous avons implémenté les interactions entre l’IA et le joueur

Le joueur peut tuer les ennemies en 3 coup et inversement cependant le joueur ne eut pas tuer tous les ennemie mais seulement les ennemie mobile

Cette section comporte trois nouveaux ajouts :

1. l’IA met à jour la vie du joueur en la réduisant quand ils se touchent ou le touche avec une projectile

2. l’IA et le joueur se font téléporter à leurs points d’apparition respectifs quand la partie est terminer cad que le joueur est mort

. Nous allons désormais expliquer ci-dessous comment nous avons implémenté ces différents ajouts : Tout d’abord, pour détecter que l’IA a réussi à atteindre le joueur, mécanique utilisée dans le 1er et le 2e ajout, nous avons utilisé un signal lié du nœud NavigationAgent3D de l’IA nommé target\_reached que nous avons lié dans le script du fantôme à la fonction \_TargetReached(). Ce signal se déclenche lorsque l’IA aura atteint certaines distances avec sa cible, dans notre cas 1,5 m. Pour ce premier ajout, il fallait que nous modifions un attribut du joueur. Nous avons donc utilisé un signal envoyé depuis l’IA, dans la fonction GotCaught(), au joueur afin que ce dernier modifie lui-même son attribut dans la fonction \_onEnemyCaught(). page 18 Les maisons aux âmes perdues EPITA 2023 Script de l’IA : p r i v a t e void GotCaught ( ) { Emi tSignal ( SignalName . Caught ) ; } Script du joueur : p r i v a t e void \_onEnemyCaught ( ) { Champsdevision−= 1 ; // ( r e s t e du code ) } Pour le 2e ajout, nous avons simplement réutilisé les fonctions précédentes pour téléporter les deux personnages. Nous avons téléporté l’IA dans \_TargetReached() et le joueur dans \_onEnemyCaught() en changement leurs positions aux coordonnées de leurs points de réapparition respectifs. Script de l’IA : p r i v a t e void \_TargetReached ( ) { P o si ti o n = \_spawnPoint ; //On TP l e fantome a son SpawnPoint GotCaught ( ) ; Rpc (" Synch\_Pos " , \_spawnPoint ) ; } Script du joueur : p r i v a t e void \_onEnemyCaught ( ) { Champsdevision−= 1 ; P o si ti o n = \_respawnPoint ; // ( r e s t e du code ) } Pour le dernier ajout, nous avons ajouté dans la fonction \_onEnemyCaught() du joueur un test pour savoir si son champ de vision est descendu à 0. Si c’est le cas, le joueur va envoyer un signal au script de la scène principale pour que celle-ci change la scène. page 19 Les maisons aux âmes perdues EPITA 2023 Script du joueur : p r i v a t e void \_onEnemyCaught ( ) { Champsdevision−= 1 ; P o si ti o n = \_respawnPoint ; i f ( Champsdevision == 0 ) { Emi tSignal ( SignalName . PlayerDead ) ; } } Enfin, le dernier point a été l’instanciation du fantôme. Pour ce faire, il nous a fallu ajouter au script de la scène principale la ligne suivante : [ Export ] p r i v a t e PackedScene \_phantomScene ; Celle-ci permet de récupérer la scène du fantôme que nous instancions dans la fonction \_Ready(). Ensuite, nous attribuons au fantôme son nom, pour pouvoir récupérer son nœud dans d’autres scripts, ainsi que sa position de départ qui est son point de réapparition.

3.8 Le Site Web La création du site web de nova studio a débuté par une Apprentissage autonome des langages de programmation en effet La création du site web pour Nova studio a nécessité l’apprentissage autonome de langages de programmation tels que HTML, CSS et JavaScript. Comprendre les concepts de base et les bonnes pratiques de programmation a demande du temps et de la persévérance.

De plus un des défis était de trouver un style visuel qui puisse satisfaire l’ensemble de l’équipe. Pour cela nous avons utiliser beaucoup de tuto YouTube afin de nous inspirer mais également regarder diffèrent site de jeux tel que ea, Ubisoft pour voir comment ça se présentait cela nous a permis de créer des betas de notre site et de tester différentes options de design avant de prendre une décision finale

. Pour dynamiser le site et le rendre plus interactif, nous avons intégré des interactions aux boutons et ajouté des images illustrant notre projet, nous avons mis à jour régulièrement les ressources utilisées. Cette démarche vise à garantir la transparence envers les joueurs et à reconnaître le travail des créateurs qui partagent leurs ressources.

De plus le site est optimiser pour petit et grand écran pour avoir un site polyvalent qui peut s’adapter en effet pour l’écran mobile il y a la mise en place d’un affichage burger

4.5 L’Intelligence Artificielle Le premier problème que nous avons rencontré lors de la création de l’IA est le peu de documentation que l’on peut trouver en C#. Cela nous a donc demandé un peu de temps d’adaptation pour pouvoir comprendre le GDScript et le convertir en C#. Nous avons ensuite rencontré l’erreur suivante : Elle se produisait lorsque nous utilisions le nœud NavigationAgent3D du fantôme avant que la carte ne soit synchronisée. Pour résoudre cela, nous avons ajouté dans la fonction \_Ready() de l’IA un CallDefered() qui appelle la fonction WaitNavMapSync() qui va attendre que la première image soit générée pour laisser le temps au NavigationServer de se synchroniser avant d’utiliser le NavigationAgent3D. page 30 Les maisons aux âmes perdues EPITA 2023 C a ll a bl e . From (WaitNavMapSync ) . C all D ef e r r e d ( ) ; p r i v a t e async void WaitNavMapSync ( ) { awai t ToSignal ( GetTree ( ) , SceneTree . SignalName . PhysicsFrame ) ; \_navAgent . T a r g e tP o si ti o n = \_player . P o si ti o n ; } Pour l’implémentation de l’interaction entre IA et joueur, utiliser les signaux ne fut pas quelque chose d’aisé. De plus, pour connecter les signaux, il faut le faire depuis le nœud dont on veut relier au signal en appelant la fonction Connect() ce qui requiert d’obtenir le nœud qui envoie le signal. Et pour ce faire, il faut donc que le nœud qui souhaite récupérer l’autre soit instancié après. Le problème est que dans notre cas l’IA a besoin du nœud du joueur pour le pathfinding \_player = GetNode('/ roo t /Node3D / 1 ') ; \_navAgent . T a r g e tP o si ti o n = \_player . P o si ti o n ; Le joueur a besoin du nœud de l’IA pour se connecter au signal Caught de l’IA indiquant que le joueur s’est fait toucher. Script du joueur : p r i v a t e void ConnectSignalToAI ( ) { // Connect l e s i g n a l Caught de fantome . c s GetNode("/ roo t /Node3D/fantome " ) . Con n e c t (" Caught " , new C al l a bl e ( t hi s , "\_onEnemyCaught " ) ) ; } Cependant, il est impossible d’instancier l’IA et le joueur en même temps. Ainsi, il y aura obligatoirement un nœud qui sera instancié avant l’autre, ce qui pose comme problème que le premier nœud instancié ne pourra pas récupérer l’autre nœud, car cela générera une erreur. Pour remédier à cela, étant donné que c’est le script de la scène principale qui instancie l’IA et le joueur, elle sera instanciée avant eux. Nous avons donc rajouté dans la fonction \_Ready() de ce script un signal envoyé au script du joueur pour lui dire que l’IA a bien été instanciée (car dans notre cas, c’est le joueur qui est instancié en premier) et donc que le script du joueur peut récupérer le nœud de l’IA. page 31 Les maisons aux âmes perdues EPITA 2023 Script de la scène principale : Emi tSignal ( SignalName . AIReady ) ; // Connecte a l a f o n c t i o n ConnectSignalToAI ( ) du jo u e u r Script du joueur : \_p riva te void ConnectSignalToAI ( ) { // Connecte l e s i g n a l Caught de fantome . c s //GD. P ri n t ("Enemy s i g n a l connec ted to Pla y e r " ) ; GetNode("/ roo t /Node3D/fantome " ) . Con n e c t (" Caught " , new C al l a bl e ( t hi s , "\_onEnemyCaught " ) ) ; } 4.6 Le Multijoue

4.5 L’Intelligence Artificielle Le premier problème que nous avons rencontré lors de la création de l’IA est le peu de documentation que l’on peut trouver en C#. Cela nous a donc demandé un peu de temps d’adaptation pour pouvoir comprendre et surtout la visualisation de nombreux tuto pour démarrer

Deuxièmement nous avons eu un problème avec les IA car il ne détectait pas le joueur et tout cela était un problème de tag en effet au début nous n’avions pas créé et mis les tag Player et bot pour que chacun soit bien distinguer donc les ia malgré leurs code ne détectait jamais le joueur la solution a donc été simplement l’ajout de tag qui correspond au code de chacun des enemie pour pouvoir détecter le joueur

Troisièmement un des plus gros problèmes rencontrer a été de faire en sorte d’implanter nos ennemies dans la Map car a la base chacun à gérer sa partie dans son coter c’est-à-dire que la Map a été fait à part le joueur aussi et également les ennemies

Donc lors de la mise en commun on a eu énormément de problème vis-à-vis de la cohérence de nos codes de chacun et surtout des diffèrent tag qui était mal assigné

Donc nous avons du corriger la plus part des codes et mettre en commun afin avoir la cohérence des codes et ainsi faire fonctioneer le jeux

D’autre part il y a eu notamment des erreurs de code qui ont provoqu´e des bugs et des probl`emes de fonctionnalit´es. La r´esolution de ces erreurs a n´ecessit´e du temps et des comp´etences en d´ebogage.

De plus on a eu un problème de map en effet dans la première version de la map il n’y avait pas les navmesh surface doonc les enemy avec un navmesh agent ne pouivais pas bouger dans la map librement car il n’yavais pas la zone pour se dealcer c’est donc purc e que nous donc du ajouetr un navmesh surface sur toute la map afin que les enemy cncerner ppuisse focntoner

Nous avons remarqué par la suite que le bot avait un nouveau problème après l’avoir mlenter dans la map qui est que dans le code initiale leneme ne prenais pas en compte les obtacle donc lorsque le bot commençais a aller dans le vide et bien il était stopper par le mur insible et continuer a avancer de dans

Donc la solution a été de faire en sorte que des que le bot se rapproche d’un obstable et bien il changais de direction grâce un code fixer

Le site web les problèmes :

Le premier problème rencontré lors de la création du site a été notre manque de connaissance sur ce sujet en effet nous avons donc dû apprendre en automne les diffèrent langage de css html et java script chacun avec une utilité totalement différente

En effet un de nos premier problème a été de faire en sorte de crée le style de site avec le code html

Ce qui a été une erreur de députant typiquement en effet c’est grâce a un tuto YouTube que nous avons pu comprendre que le html était utilisé pour construire l’architecture du site

Le css quant a lui le design les couleur ext… entre autres le design

Et le java script permet de crée les animations telle que les carrousels et les boutons interactif après avoir compris tout cela

Le 2eme problème apparu a été un problème avec l’architecture en effet inutilement nous avions mal compris le principe de section et de class donc les éléments du site ont été implanter n’importe comment c’est donc après des bonne heures de débogage et de remise à l’ordre que le site a été corriger

A eu une architecture correcte

Un des derniers gros problème du site a été une erreur assez basique

Qui a été de commencer a develoer le site pour pc avant de le développer pour téléphone en effet

Il est toujours plus simple de d’optimiser un site de mobile vers pc que l’inverse ainsi

Lorsque le site a été terminer pour les grand ecran nouus avions remarquer que pour les petit le site avait donc un gros problème tel que les boutons se rentrat de dedans la taille des carrer était soit tro grand soit trop petit et c’était ilisisble donc après de nombreux tuto et l’aide également d’ia nous avons pu régler tout cela et même réussir a mettre en place le fameux affichage burger pour mobile

Afin de dynamiser l’expérience utilisateur et d’accroître la difficulté du jeu, les modifications suivantes seront apportées au jeux pour la prochaine soutenance :

Intégration des Boss : nous allons essayer de crée diffèrent boss qui auront a disposions des pattern d’attaque diffèrent

Par exemple un boss qui lorsque le joueur est trop loin envoie une boulle de feu

S’il est trop proche le boss donne des coups de poing et a mid distance saute et envoie une onde de choc

Ensuite l’intégration des animations : pour les ennemis nous allons essayer de faire en sorte que les ennemies lance une animation a chaque attaque una animation lorsqu’il patrouille et une animation lorsqu’il poursuit le joueur

Création des designs pour les ennemie qui en ont toujours pas afin davoir un jeux plus immersif

Mise en place d’une difficulté plus élevé et prise en compte de plusieurs personage par les ennemie . si jeux se joue a plusieurs : le but vas être que lorsque le jeux se joura a 2 les enemie auront plus de vie feront plus de dégâts mais aussi d’ajouter le code pour les ennemi pour qu’il chasse le plus proche

Test et validation : Tester les modifications apportées et recueillir les avis des utilisateurs pour identifier d’éventuels points d’amélioration. Les modifications prévues pour la prochaine soutenance visent à enrichir l’expérience de jeu et à renforcer l’interaction entre le joueur et l’IA. Le développement de l’IA est en bonne voie et sa finalisation est imminente.