

Rapport final

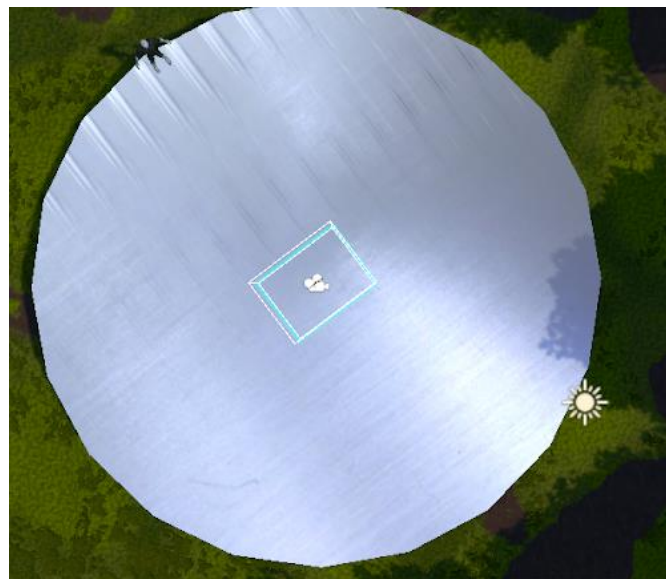
SHIELD



OUT

Shield Out, un jeu où son bouclier devient arme

Shield out est un jeu addictif où il vaut mieux être proactif que défensif. Plus le temps passe, plus il est difficile de rester en vie. Faites attention au monde qui vous entoure car des ennemis rôdent dans les parages, et ils vous en veulent.



SOMMAIRE

Généralités	4
Synopsis	4
Equipe	4
Contexte	4
Ressources	4
Description de Shield Out	6
Les mécaniques et interactions utilisateurs	6
Le code et la modélisation	7
Niveaux de réalisation	8
Niveau 1	8
Niveau 2	8
Niveau 3	8
Problèmes rencontrés	9
Bilan et retour sur le projet	10

Généralités

Synopsis

Shield Out propose une expérience de réalité virtuelle permettant de s'immiscer dans un jeu roleplay. Le joueur est entouré d'ennemi qui lui envoie des projectiles de différentes natures. Le joueur doit renvoyer les projectiles à l'aide d'un bouclier qu'il porte à la main. Le but est de réduire à néant ses adversaires soit en récupérant à l'aide de son autre main un des projectiles lancés, soit en maîtrisant son bouclier et ajuster le renvoi. Il existe plusieurs types de difficultés, et par extension, plusieurs types de projectiles. L'immersion sera visuelle, physique et sonore. Shield Out peut être considéré comme un « *endless game* » avec la possibilité d'obtenir un score toujours plus élevé.

Equipe

Notre équipe est composée de deux étudiants de l'Université de Technologie de Compiègne (UTC). Hugo Paigneau et Guillaume Sabbagh sont étudiants respectivement en 4er et 3e semestre de Génie Informatique à l'UTC. Les deux étudiants n'ont pas d'expérience sur Unity ou autres moteurs 3D mais sont motivés par l'apprentissage des technologies utiles à la réalisation de projets en réalité virtuelle. Ils seront chargés du développement des interactions et des comportements physiques, ainsi que de l'optimisation de l'interfaçage entre Unity et les équipements de réalité virtuelle prévus dans le cadre du projet.

Contexte

Le projet est réalisé dans le cadre de l'unité de valeur RV01 (Réalité Virtuelle) de l'Université de Technologie de Compiègne. Cet enseignement a pour but d'initier les étudiants à la réalisation d'applications liées à la réalité virtuelle, augmentée ou mixte en mettant à leur disposition des moyens d'immersion tels que des lunettes, des capteurs, des salles type CAVE.

Ressources

Dans le cadre de l'UV de Réalité Virtuelle, les outils mis en œuvre sont :

- L'HTC Vive pour l'immersion visuelle.



- Un set de manettes qui serviront à matérialiser le bouclier en main gauche et la main droite
- Le développement de l'application est réalisé sous Unity. Les ressources graphiques seront simples (cubes, sphères, quelques ressources libres de l'Asset Store).

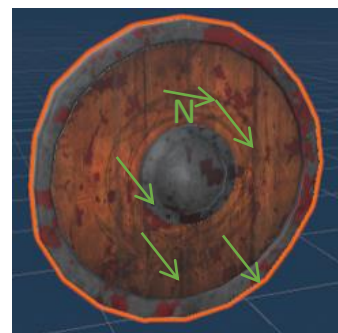


Description de Shield Out

Les mécaniques et interactions utilisateurs

Dans ce jeu, nous nous sommes concentrés sur deux mécaniques principales (le parage au bouclier et le renvoi de projectiles). Il est important de rappeler que nous avons préféré ne pas permettre le déplacement du joueur dans la scène. Ce choix peut se justifier par la complexité d'implémenter un tel jeu en si peu de temps.

Tout d'abord, le bouclier reste tout le temps collé à la main gauche du joueur. Il suit les mouvements de la manette que le joueur tient dans sa main gauche. Le bouclier suit en fait les mouvements du joueur comme un vrai bouclier le ferait ce qui permet de parer efficacement, et de rendre réel l'interaction. Le renvoie bouclier est une des deux façons pour renvoyer un projectile vers un adversaire et ainsi, augmenter son score et la difficulté par la même occasion.



La deuxième est bien évidemment la main droite. Le joueur a la possibilité de récupérer un projectile qui arrive vers lui directement. Le projectile ne doit cependant pas toucher le joueur, car sinon il est considéré comme « détruit » et compte pour une mort. Pour récupérer un projectile, le joueur doit tout simplement timer l'arrivée du projectile vers sa main

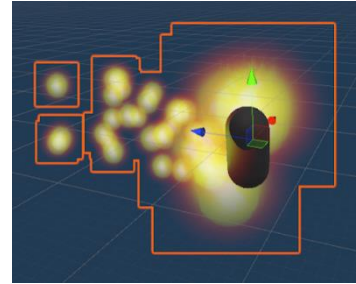
droite et actionner « right on time » le bouton trigger du contrôleur droit. Pour avoir une sensation de prise d'objet réel, il faut rester appuyé sur le trigger pour garder le projectile dans sa main.

Côté sonore cette fois, nous avons mis en avant la visualisation sonore du projectile en ajoutant des sons audios sur les différentes interactions possibles que peut avoir un projectile. C'est-à-dire lorsqu'il est tiré par un ennemi, lorsqu'il se déplace, lorsqu'il touche sa cible ou encore lorsqu'il est renvoyé par le joueur.

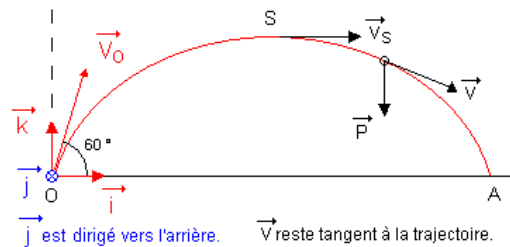
Il est important de noter que différentes stratégies ont été adoptées pour la réalisation du jeu. On peut parler d'optimisation 3D par exemple lorsque nous considérons les collisions entre les différents objets : Calcul intersection entre boîtes englobantes fixes et celle de l'avatar.

Le code et la modélisation

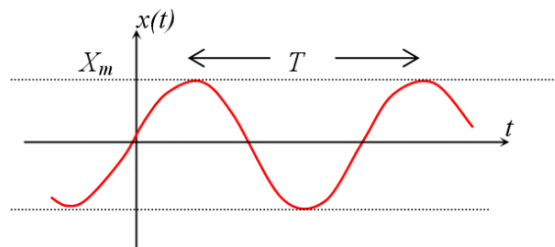
Un point qui nous a vraiment intéressé dans le projet a été de réaliser les objets utilisés dans notre scène principal. Ainsi, nous avons modélisé les différents types de projectile en partant d'une base de capsule, puis créé les textures sur Photoshop ainsi que gérer des « *ParticleSystem* » dans Unity. Le rendu est plutôt optimisé mais n'est pas parfaitement réaliste. Il est aussi à noter que les ennemis sont scénarisés dans le sens où ils répondent à des animations suivant les actions qu'ils font suivant un « *graphe d'animation* ».



Le deuxième point important concerne les mouvements et paramétrisation des projectiles. Nous avons réalisé 3 différents projectiles : on a utilisé pour l'un d'entre eux la physique de Unity et paramétré une équation de courbe pour qu'il atteigne, selon son angle de lancée, sa vitesse et la cible à atteindre (le joueur).



Pour le projectile plante, nous avons écrit un algorithme qui fait suivre le projectile une trajectoire hélicoïdale :



Niveaux de réalisation

Niveau 1

L'objectif de ce niveau minimal était de réaliser un environnement simple où un ennemi envoie un missile de manière rectiligne. Il fallait aussi d'implémenter le bouclier qui part les projectiles et ces derniers immédiatement renvoyés dans la direction de son regard avec une aide à la visée. Nous avons abandonné l'idée d'aide à la visée puisqu'on s'est vite rendu compte que ce n'était pas si dur de viser pour le renvoi. Ce premier niveau a été facile d'implémentation.

Niveau 2

Ici, Le joueur pouvait recevoir des projectiles de la tête aux pieds et doit donc parfois se baisser pour arrêter les projectiles. Les projectiles devaient aussi être améliorés et donc diversifiés. Nous avons aussi abandonnée l'idée de s'abaisser car cela compliquait les mouvements circulaires avec le casque VR et son fil. Cependant, nous avons bien réussi le reste et même ajouté des effets sonores.

Niveau 3

Le niveau de réalisation optimal devait permettre de rendre résistant certains ennemis selon le type de projectile qui le touchait. Il devait aussi permettre au joueur de ramasser ou récupérer un projectile avec son autre main. Nous avons préféré passer la première étape au profit du design et de l'animation des ennemis pour rendre le jeu plus ludique. Cependant, la récupération et le rejet des projectile a été un vrai succès !

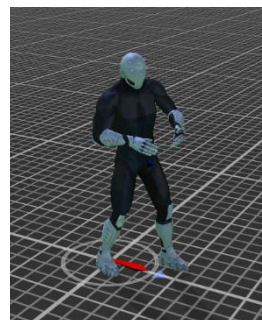
Problèmes rencontrés

La gestion du projet s'est plutôt bien passée. Nous étions deux à travailler sur le projet et le développement du projet a été réalisé ensemble. Autrement dit, nous avons travaillé sur le projet ensemble afin de pouvoir couvrir tous les aspects du jeu car nous étions intéressés dans tous les facettes du projet. La première difficulté que nous avons surmontée a été l'adaptation de SteamVR à la dernière version et la prise en main de son API car elle a définitivement changé : la gestion de l'équipement suit désormais une logique User Action.

Nous nous sommes acharnés à l'étude en profondeur de la documentation légère de SteamVR mais sommes parvenu à nos fins. Après ce cap passé, nous étions en bonne voie pour réaliser un projet qui nous a extrêmement intéressé.

Survient alors un deuxième problème : le design et modeling sur Unity. Cette passe a été décisive dans l'avancée du projet. Nous avons pris énormément de temps pour designer des projectiles correct avec des particules après avoir parlé avec des personnes compétentes.

Nous pouvons aussi parler de l'animation des ennemis : après avoir récupéré un modèle sur le SteamVR Community Forum, nous avons étudié de fond en comble l'animation d'ennemis ; il a été instructif d'apprendre à gérer des animations aussi bien dans le code comme dans l'interface Unity. Il aurait été intéressant de voir cette partie en TD guidé, même si la documentation sur le site RV01 nous a été d'une grande aide.



Ensuite, nous voulons aborder notre plus gros problème : le développement. Plus précisément, nous avons eu beaucoup de mal à développer la partie projectile : prendre en compte la gravité Unity et réaliser des équations vues dans des UV mécaniques, gérer l'hélicoïde, etc...

Pour terminer, il n'est sans nul doute que notre dernier problème rencontré a été le manque de temps. Le projet que nous voulions réaliser au début était assez idéaliste et nécessiterait 6 mois de plus pour être abouti et proposé au grand public, notamment pour rendre le jeu plus interactif avec la possibilité de bouger, de jouer en ligne ou encore de créer des ennemis intelligents.

Bilan et retour sur le projet

Pour évaluer le résultat par rapport à l'objectif de base, nous sommes plutôt heureux du résultat (notamment des outils théorique et technique vis-à-vis du code), même s'il y a des points négatifs qu'il est important de noter : le manque de temps nous a obligé de nous concentrer sur des réalisations bien précises et abouties plutôt que de tout réaliser vaguement. En effet, il manque l'implémentation des ennemis résistants et autonomes qui apporterait un gros plus à ShieldOut mais le manque de connaissance et de temps nous ont fait prendre conscience qu'il était impossible d'implémenter ces deux fonctionnalités en 3 mois (après TD).

Globalement, ce projet a été très enrichissant et productif. Nous étions tous deux motivés par ce projet et a fait naître une belle rencontre de compétences. Nous aurions aimé cependant avoir plus d'interactions entre le cours et le projet.

Nous avons vu des choses marquantes en cours que nous ne voyons pas en pratique (on peut par exemple parler des modèles de lumières qui aurait été intéressant d'implémenter pour ajouter un point positif à l'immersion RV, ou encore de l'optimisation 3D et le Level Of Detail qui est obligatoire désormais dans un jeu développer sous moteur 3D pour alléger les calculs, mais nous n'avons pas eu la possibilité d'essayer. Ceci reste compréhensible vu que l'UV est englobée dans une perspective semestrielle à l'UTC et que nous n'avons pas assez de temps pour couvrir toutes les notions théoriques en TD.

Finalement, Shield Out restera un projet qui se démarque des autres à l'UTC puisque réellement prenant pour nous.