

PROJET INFORMATIQUE S2
JEU VIDÉO : POLY

Rapport de Projet

Équipe EpiCATT :

Charles DUCHAMP
Tugdual ECOFFET
Tony GEORGE
Antoine LÉA

2017 - 2018

1 Avant-propos

Le but de ce rapport est de présenter notre projet dans son ensemble, les outils et méthodes utilisées pour le mettre en œuvre, ainsi qu'une comparaison des résultats escomptés et obtenus. Ce document se veut à la fois exhaustif et synthétique.

Cette tâche est cependant rendu complexe par un hasard de calendrier. En effet, au jour d'écriture de ce rapport, la dernière semaine et demi de "rush" précédent chaque soutenance n'a pas encore été effectuée. De par l'importance de ces semaines en terme de productivité et de volume horaire, les parties concernant troisième la soutenance contiennent uniquement des prévisions pour la semaine à venir, par opposition ou autre soutenance qui contiennent à la fois prévision et réalisation.

Comme dit précédemment, ce rapport reprend tout les éléments relatifs à notre projet, et ce de manière chronologique.

Vous découvrez ainsi en premier lieu une reprise du cahier des charges, avec une présentation du projet tel que nous l'avons conçu, à la fois mécaniquement et artistiquement.

Dans un second temps, vous trouverez les tâches que nous avons déléguées, ainsi que les méthodes que nous avons utilisées pour les implémenter (avec au passage, les problèmes que nous avons rencontrés).

Enfin, nous ferons un point sur l'avancement global du projet, les réussites et les défaites du point de vue du cahier des charges dressé en début d'année, puis d'un point de vue plus humain, le ressenti de chacun face au projet.

Ce document représente donc l'aboutissement de six mois de travail intensif, de recherche et de mise en pratique. Projet pour lequel nous attendions beaucoup, et qui a été à la hauteur de nos espérances.

Table des matières

1	Avant-propos	1
2	Présentation du projet	3
2.1	EpiCATT, notre entreprise fictive	3
2.2	PoLy : Concept & Description	3
2.2.1	L’Histoire	3
2.2.2	Le Gameplay	3
2.3	L’intérêt et le but du projet	3
3	Information contextuelles (Benchmarking)	4
3.1	Origine et nature	4
3.2	État de l’Art	4
4	Mécaniques de jeu	6
4.1	Déroulement d’une partie	6
4.2	Les bâtiments classiques	6
4.2.1	Le Générateur	6
4.2.2	Les Labos	6
4.2.3	Le Pare-Feu	6
4.3	Les bâtiments spéciaux	7
4.3.1	The Hill	7
4.3.2	Les Serveurs	7
4.4	Spécificités techniques	7
5	Direction Artistique	9
5.1	Le Contexte (Lore)	9
5.2	La Palette graphique	9
5.3	La Musique	10
6	Explication des tâches	11
7	Projet et avancement réel	13
8	Chronologies des implémentations	14
8.1	Première soutenance (Vendredi 16 Mars)	14
8.2	Seconde soutenance (Vendredi 27 Avril)	17
8.3	Troisième soutenance (Lundi 11 Juin)	22
9	Conclusion	23
9.1	Résultats du projet	23
9.2	Les joies et les peines	23
9.3	Derniers mots	27

2 Présentation du projet

2.1 EpiCATT, notre entreprise fictive

EpiCATT est l'organisation à l'origine de poLy. C'est une fausse entreprise qui a été créée afin d'organiser tous les outils nécessaires à la création du projet. Elle dispose ainsi d'une adresse E-mail, d'un Github (plateforme pour la mise en commun de code), de réseaux sociaux (Facebook, Twitter, YouTube), d'un site internet, etc. Elle a servi de support pour la structure même du projet.

2.2 PoLy : Concept & Description

2.2.1 L'Histoire

Dans notre monde, beaucoup de batailles se gagnent en réalité sur le papier. La tactique est nécessaire aussi bien dans la diplomatie que dans la guerre. Dans le monde de poLy, l'information est le pouvoir. Les gens l'ont bien compris. Vous, en tant que joueur, vous battez pour le monopole de réseaux, afin d'étendre votre contrôle sur l'information. Que vous comptiez utiliser ces informations pour votre intérêt personnel ou le bien du monde importe peu. Vous n'êtes pas seul sur le coup. Maintenant, il va falloir vous battre.

2.2.2 Le Gameplay

PoLy est un *RTS Arena*, un jeu de stratégie en temps réel (Real Time Strategy). Les joueurs sont placés dans des arènes, dont ils doivent acquérir le monopole en triomphant de leur(s) ennemi(s). Pour ce faire, ils prennent le contrôle de bâtiments aux fonctionnalités diverses (aussi appelés *Tours*). Ces derniers contiennent des unités, que les joueurs déplacent librement afin de prendre le contrôle d'autres bâtiments, et ainsi étendent leur contrôle jusqu'à la victoire. Un joueur est déclaré vainqueur lorsqu'il a vaincu toutes les unités ennemis.

2.3 L'intérêt et le but du projet

Le but premier de ce projet est évidemment d'acquérir de l'expérience technique, avec la manipulation d'outils tel que *Unity*, et le codage en *CSharp*. C'est aussi un nouveau défi à relever en terme de complexité et de volume horaire.

A cela s'ajoute l'aspect de la gestion de projet. L'organisation du temps de travail, la répartition des tâches, le conflit entre qualité du produit et respect des violentes deadlines, sont autant de problématiques auxquelles il nous faut répondre.

3 Information contextuelles (Benchmarking)

3.1 Origine et nature

"PoLy ou les nouveaux échecs" : Telle était notre ambition pour ce projet. Étant des individus aimant la compétition, il était évidant que nous créions un jeu en accord avec cet état d'esprit.

Les RTS sont connus pour être des jeux très complexes, peu ouverts aux non-initiés. PoLy casse cet aspect élitiste en adoptant des mécaniques de jeu simples, et ce en gardant une certaine profondeur de gameplay. Le jeu s'inscrit dans la politique *"Simple to Learn, Hard to Master"* (*S2L/H2M*). Les règles du jeu sont simples à apprendre, mais les subtilités de stratégie rendent le jeu très complet.

Le jeu récompense la capacité de contrôle du joueur : Sa capacité à prendre les bonnes décisions rapidement, à lire le jeu de son adversaire et à anticiper ses mouvements, afin de le contrer avec la stratégie adaptée.

3.2 État de l'Art

Lorsque l'on demande à un joueur aguerri de citer un jeu de type RTS, celui-ci a de grandes chances de répondre *StarCraft*. Et de fait, *StarCraft* est le géant qui a défini les RTS et dont on ressent encore l'influence aujourd'hui. Ce jeu est aux RTS ce que *Mario* est aux Plateformeurs, et ce que *Magic* est aux jeux de cartes. Le jeu est sorti le 31 mars 1998, et est toujours actif aujourd'hui. La quantité d'informations et de tâches que doivent gérer les joueurs force l'admiration des néophytes, mais crée aussi une forme de détachement pour les non-initiés. C'est là qu'intervient notre source première d'inspiration : *Mushroom Wars*.

Mushroom Wars (deuxième du nom), est un jeu indépendant sorti en octobre 2016 sur la plateforme *Steam*. Ce dernier prend un parti intéressant sur la formule du RTS. Il simplifie le genre à l'excès, rendant le jeu bien plus attrayant au grand public. En effet, là où la plupart des RTS jugent le joueur sur sa connaissance des mécaniques de jeu et récompense le joueur vétéran, *Mushroom Wars* jugent le joueur dans sa capacité à prévoir, anticiper, et contrer ses adversaires. Le jeu acquiert une toute autre dimension, qui se repose exclusivement sur le *Mind Game*, le fait de lire le jeu de son adversaire tout en le trompant sur les objectifs que l'on vise réellement.



FIGURE 1 – Arène du jeu *Mushroom Wars*

PolY est notre interprétation du concept initié par *Mushroom Wars*. Il reprend la plupart des mécaniques de ce dernier. Nous enlevons et rajoutons des éléments de gameplay afin de donner à *polY* une atmosphère de jeu de plateau (toujours dans l'objectif de créer "*les nouveaux échecs*").

Ceci permet une transition vers notre dernière principale source d'inspiration : les Échecs eux-mêmes. Dans ces derniers vous retrouvez tout l'aspect Mind Game décrit précédemment : imaginer et anticiper ce que l'adversaire jouera en réponse à votre coup, et ce sur plusieurs tour. La grandeur d'un joueur d'échecs est aussi mesurée à sa capacité d'imaginer le plateau dans X coups.

Cet hommage à ce jeu de plateau mythique se fait surtout par l'aspect graphique de notre jeu. En effet, savoir jouer les pièces de bois ne donnera pas d'avantages directs sur *polY*, les mécaniques sont trop différentes. Cependant, l'esprit tactique nécessaire pour jouer aux deux jeux permet un apprentissage plus rapide, ainsi que la création de stratégies plus complexes menant à la victoire.
Échec, et mat.

4 Mécaniques de jeu

4.1 Déroulement d'une partie

Une fois leur(s) adversaire(s) trouvé(s), les joueurs sont propulsés sur la carte. Sur cette dernière, joueur contrôle au minimum une tour chacun, sachant que les autres tours peuvent être neutre au début de la partie.

Après un décompte aux joueurs permettant de prendre conscience de l'arène dans laquelle ils vont se battre, les joueurs sont totalement libre de leur mouvement. Ils préféreront dans un premier temps prendre le contrôle des bâtiments neutres afin de monter rapidement en force, avant d'attaquer les bâtiments adverses.

Pour ce faire, il peut déplacer ses unités vers d'autres tours, changer le type de ses bâtiments (parmi les 3 type décrit ci dessous) ou augmenter le niveau de ceux-ci. Lorsque le nombre d'unités que contrôle un joueur atteint zéro, il perd la partie. Le joueur gagnant est celui ayant éliminé tous les autres.

4.2 Les bâtiments classiques

4.2.1 Le Générateur

Le rôle du Générateur est de créer des entités. La vitesse de production augmente avec le niveau du bâtiment. Cette dernière tombe cependant à 0 si le nombre total d'entités contrôlées par un joueur dépasse une certaine valeur.

4.2.2 Les Labos

Le Labo augmente la force de combat des entités. Par défaut, une unité alliée vainc une seule unité ennemi, soit un ratio de [1 :1]. Les Labos font monter ce ratio et permet d'attaquer des bâtiments plus peuples avec moins d'unités. C'est le seul bâtiment qui ne peut pas être amélioré. Ceci permet d'éviter la surprotection de ce dernier et le rend à la fois plus vulnérable et coûteux (par le manque en production d'entités).

Le nombre de Labo contrôlés par un joueur est illimité, cependant le bonus de combat n'est cumulable que trois fois.

4.2.3 Le Pare-Feu

Le Pare-Feu empêche les unités ennemis de progresser dans son champ d'action. Ce dernier agit comme une tourelle, attaquant les unités. Un tir est suffisant pour détruire une entité, quelque soit le niveau du bâtiment. Cependant, la vitesse d'attaque et la portée du Pare-Feu sont augmentés selon ce dernier.

4.3 Les bâtiments spéciaux

4.3.1 The Hill

Ce bâtiment est exclusif au mode *King of the Hill* (KotH). Les joueurs s'affrontent pour le contrôle de ce bâtiment. Contrôler ce dernier permet de faire monter une valeur propre au joueur, qui une fois arrivé à 100% donne la victoire à celui-ci. Cet intervalle de temps est cumulé (Pas de remise à zéro du compteur lors de la perte de contrôle du bâtiment). De même, il peut y avoir plusieurs bâtiment de ce type sur une même carte. Un bonus est alors accordé pour le contrôle de multiple serveur (Score calculé de manière non linéaire).

4.3.2 Les Serveurs

Ces bâtiments sont exclusifs au mode *Capture the Point* (CTP). Pour gagner dans ce mode, un joueur doit contrôler tous les Serveurs présent sur la carte simultanément. Afin d'assurer un temps de jeu convenable, chaque joueur dispose d'au moins un serveur en début de partie (Contient les "*Rush*"). De même, si la partie dure trop longtemps, un timer y met fin. Le joueur contrôlant alors le plus de serveur gagne alors la partie (une partie peut se finir sur une égalité).

4.4 Spécificités techniques

Les bâtiments sont générés selon le réseau (arène) choisi, il est impossible pour les joueurs d'en créer. Chaque bâtiment est soit neutre, soit contrôlé par un unique joueur.

La valeur d'un bâtiment dépend uniquement du nombre d'entités qu'il abrite. Les joueurs peuvent payer un certain nombre d'entités afin d'augmenter son niveau, améliorant ainsi ses fonctionnalités. Ils peuvent modifier le type du bâtiment de la même manière (*Générateur* / *Pare-Feu* / *Labo*).

Chaque joueur connaît le nombre d'entités totale de ses adversaires, mais ne peut savoir pas le nombre d'entités présentes dans un bâtiment spécifique.

Les joueurs peuvent donner l'ordre de déplacer tout ou partie des entités vers n'importe quel autre bâtiment. Ces dernières sortiront successivement une à une. Si les entités se déplacent vers un bâtiment contrôlé par le joueur qui les a envoyé, elles sont cumulées. Dans le cas contraire, elles rentrent en conflit avec les entités sur place.

En cas de conflit, les entités s'échangent selon un certain ratio. Par défaut, celui-ci est de 1 pour 1, mais certains facteurs peuvent le modifier (tel le nombre de Labos contrôlés).

Le déplacement d'entités entre deux bâtiments crée un lien temporaire entre ces derniers, rendant les futurs déplacements plus rapides. Ceci se fait pour tous les bâtiments indépendamment des joueurs qui les contrôlent.

Les cartes sont déterminées et créées à l'avance. Elles se veulent équitables (symétriques), et ne doivent apporter aucun élément lié au hasard.

Chaque carte est unique, et peut contenir des zones aux effets divers :

- *Champ Électrique* : Vitesse de déplacement des entités réduite.
- *Supraconducteur* : Vitesse de déplacement des entités augmentée.
- *Brouilleur* : Défense des bâtiments légèrement réduite.
- *Nébuleuse* : Défense des bâtiments légèrement augmentée.

D'autres zones sont en cours de réflexion.

ATTENTION : Les mécaniques décrites ci-dessus sont soumises à changement. La phase d'équilibrage décidera du maintien ou non de certaines règles. De même, il est possible que d'autres mécaniques soient ajoutées pour parfaire l'expérience du joueur.

5 Direction Artistique

5.1 Le Contexte (Lore)

Les joueurs sont mis dans la peau du pseudo-hackers futuristes, se battant pour avoir le monopole de réseaux (cartes). Les entités qu'ils contrôlent sont des incarnations d'énergie pure, semblable à des virus.

Ils disposent pour cela de Générateurs d'unités, de Pare-Feu pour se protéger, et de Labos afin de rechercher des unités plus fortes.

Pour avoir une chance de gagner, il leur faudra prendre les bonnes décisions, agir vite et aux bons moments, et faire preuve d'ingéniosité pour tromper et triompher de leurs adversaires.

5.2 La Palette graphique

Le monde Pour des raisons de lisibilité, le jeu doit rester le simple possible en terme d'ornement. C'est pourquoi l'intérieur des cartes peut être décoré sans problèmes, alors que l'intérieur sera débarrassé de toutes fioritures excessive. Des couleurs pâles sont utilisée pour le sol de l'arène, afin de contraster avec les pièces. Le jeu se permet de transporter le joueur dans des atmosphère totalement différentes, chaque carte disposant de ses caractéristiques propres.

Les bâtiments et unités Les modèles des bâtiments sont basés sur des pièces d'échecs. Le Générateur est représenté par une Dame, pièce mère des unités. Les Laboratoire prennent la forme d'un Fou, jouant sur les savants les peuplant. Comme le Pare-Feu dispose d'un rôle défensif, il prend apparence de la Tour, disposant elle même de ce rôle au Échecs. Les unités sont symbolisés par des Pions. Les pièces n'ont pas de couleur propre, elle prennent celle du joueur les contrôlant. De même, lorsque que les bâtiments gagnent en niveau, le matériaux les composant gagne en noblesse. Ainsi, une tour niveau 1 sera faite de plastique, elle se durcit en métal au niveau 2, avant de se transformer en marbre au niveau 3.

L'interface L'interface doit avant tout être fonctionnelle. Une surcharge en ornements créerait du bruit visuel pour le joueur. Cependant, cela ne l'empêche pas d'adopter un style épuré et agréable à l'œil. Les boutons d'actions sont ornés de logos afin d'être plus visible (la où lecture ferait perdre de précieuse millisecondes). L'interface délivrera des informations capitale pour le joueur, il faut donc que ces dernières soit accessible en mouvement d'œil et contraste donc avec l'arrière plan de la carte.

5.3 La Musique

Notre jeu dispose d'un style assez épuré. Pour rester en accord avec ce choix, nous avons décidé de réduire les bruitages. Le fond musical sera une musique légère mais pouvant tourner dans l'épique si besoin est. Elle ne doit pas cependant prendre le dessus sur le Gameplay, mais accompagner le joueur dans ces actions.

Comme notre jeu à une base électronique, nous utiliserons des musiques sans paroles, uniquement composé de beats. Pour ce faire, nous utiliserons des musiques libres de droits.

Nous allons aussi ajouter des sons lors d'action du joueur sur le jeu pour pouvoir permettre au joueur de comprendre ce qu'il se passe et de voir ce qu'il va se passer sur le terrain.

6 Explication des taches

Mécaniques bas niveau On qualifie de "*bas niveau*" toutes les mécaniques qui forment le noyau du jeu (le *Core*). Ce sont elles qui font fonctionner le jeu telle que l'on entend : Le fait qu'un générateur produise des unités fait partie du bas niveau. Ce sont elles qui font tirer le Pare-Feu, qui font déplacer les unités, qui permettent le changement de type des bâtiments, etc. Elles forment la fondation du jeu.

Mécaniques haut niveau Les mécaniques haut niveau sont construites par dessus les mécaniques bas niveau. Elles structurent et organisent le jeu, ce sont elles qui appellent la bonne partie du code selon l'action à effectuée. Elles sont généralement liées à l'utilisateur, dans le sens où elles interprètent ses ordres.. A titre d'exemple, le fait qu'une tour puisse envoyer une unité vers une autre tour cible est une mécanique bas niveau (c'est la méthode *Spawn*). Cependant, le fait que l'utilisateur puisse sélectionner une tour d'origine et une tour cible, ainsi que choisir une proportion d'unités à envoyer est une mécanique haut niveau (c'est la méthode *Move*, qui appelle autant de fois *Spawn* que nécessaire, avec la bonne destination). Elles forment la transition entre le joueur et le noyau du jeu

Aspect graphique On range dans l'aspect graphique tout ce qui est lié à l'apparence globale du jeu. Ceci englobe principalement les *Assets*, à savoir les textures, les modèles, les matériaux, les couleurs, etc. La création d'assets n'étant pas le but du projet, nous nous sommes occupé uniquement de leur implémentation. Les particules et autres retours visuels ("*feedbacks*") font aussi partie de cette catégorie.

Interface (UI) Lorsque le joueur interagit avec le jeu, il le fait grâce à l'interface (*User Interface*). Elle est constitué de divers boutons, de menus, elle réagit aux clics et aux ordres du joueur. De plus, elle informe aussi le joueur sur l'état de la partie (histogramme des population contrôlées par les joueurs, rapport des forces en terme de force de combat, etc). Bien qu'elle dispose d'une partie graphique, son rôle est avant tout fonctionnel. Elle communique avec le joueur, celui-ci ordonne, cette dernière informe.

Multijoueur Bien qu'explicite, cette partie comporte de nombreux aspects. La recherche d'un ou plusieurs adversaire et le paireage avec ces derniers, la création de caméra différentes pour chaque joueur (étant que les deux joueurs n'ont accès aux mêmes informations), la reconnaissance de la victoire et l'inscription dans un classements. Nous utilisons pour cela des serveurs tiers (*Photon*).

IA Mode Solo C'est "l'ordinateur" contre lequel se bat le joueur en mode campagne. Cette partie comprends l'intelligence en elle même, mais aussi la création des cartes de campagne, dont la difficulté augmente progressivement. C'est une tache très complexe, dans le sens où faire une IA réellement "intelligente" demande énormément de temps. Beaucoup de facteur doivent être analysés, et énormément d'informations doivent être acquises.

Les "bot" dans les RTS sont connus pour être facile à battre à cause de ces même problématiques. Comme l'expérience d'un joueur sera bien meilleur contre un autre joueur, cette partie est mise en retrait par rapport au multijoueur.

Communication & Site La communication est une partie qui doit permettre au moins averti de comprendre et d'avoir les informations en un minimum de temps. De ce fait, nous avons choisi l'utilisation d'un site web, d'une page Facebook et d'un compte Twitter, permettant de toucher les utilisateurs sur une large palette de réseau.

Détails des réseaux En premier lieu, nous avons mis en place le site web, qui doit contenir des éléments du jeu représentatifs et assez simple à distinguer, tel que des screenshots, les change logs (résumé des modifications d'une version à l'autre), des données sur le jeu, tout type d'élément pertinent pour l'utilisateur. Ensuite, il y a la mise en place du Twitter, qui pourrait être basique, de par sa forme de message relativement court, mais qui s'avère être d'une complexité assez poussée lorsqu'il s'agit d'attirer au premier coup d'œil l'utilisateur lambda.

Et enfin, il reste le Facebook, plateforme qui permet une plus grande liberté, que ce soit par le partage de vidéo, sur des textes plus longs ou sur des messages avec plus de message. Il faut alors jouer sur la première image pour attirer l'utilisateur qui scroll dans sa barre de recherche et n'a alors qu'une micro seconde pour se décider si il s'arrête ou non sur une publication.

7 Projet et avancement réel

Première soutenance

Projet S1	Charles	Antoine	Tugdual	Tony
Découverte Unity	***	***	***	***
Recherche de solutions			**	***
Aspect graphique	**	***		
Communication, Site	**		***	
Squelette jouable	*	*	*	**

Avancement S1	Charles	Antoine	Tugdual	Tony
Découverte Unity	***	***	***	***
Recherche de solutions			**	***
Aspect graphique	**	***		
Communication, Site	*		***	
Squelette jouable				***

Seconde soutenance

Projet S2	Charles	Antoine	Tugdual	Tony
Mécaniques bas niveau	**			***
Mécaniques haut niveau	**		**	**
Interface (UI)	**	***		
Communication, Site		**	***	

Avancement S2	Charles	Antoine	Tugdual	Tony
Mécaniques bas niveau				***
Mécaniques haut niveau	***			**
Aspect graphique	***	**		
Interface (UI)	**	***		
Communication, Site		**	***	

Troisième soutenance

Projet S3	Charles	Antoine	Tugdual	Tony
Mécaniques finales	**	**	**	***
Aspect graphique	*	***		
Multijoueur	***		**	*
IA Mode Solo				***
Communication, Site		**	***	

8 Chronologies des implémentations

8.1 Première soutenance (Vendredi 16 Mars)

Découverte Unity : 6 Heures (Tous) Pendant la période de travail nous avons tous appris les bases d'Unity grâce à une matinée intensives de tutoriel proposée par le logiciel.

Aspect Graphique

Recherche d'assets : 2 Heures (Charles, Antoine) Lors de cette partie nous avons longtemps regardé le site des assets de Unity pour pouvoir trouver des assets qui correspondaient à notre jeu. Nous avons donc pris des pions d'échecs comme expliqué plus haut.

Création du menu principal : 8 Heures (Antoine) Le menu d'accueil permet grâce à des boutons de choisir son mode de jeu ou d'aller voir les options du jeu. Ensuite chaque bouton du menu d'accueil renvoie à un autre menu qui lui est le menu du type de jeu qui permettra dans le futur de pouvoir lancer une partie en multijoueur ou en solo. Dans chaque menu un bouton permet de quitter le jeu ou de retourner au menu précédent. Les graphismes ne sont pas encore à leur état final, ils évolueront lors de la soutenance 2.



FIGURE 2 – Ancien menu principal

Création des settings : 8 Heures (Antoine) Dans cette page d'option nous pouvons changer la résolution du jeu, mettre en pleine écran, descendre ou monter les sons et la musique du jeu. La partie "difficile" des options est de pouvoir les enregistrer pour qu'elles ne se réinitialisent pas lorsque l'on change de scène, donc changer une fois les options et ne pas les changer à chaque fois que l'on change de scène. Cette partie n'est pas encore effectué lors de la première soutenance.

Création des maps : 2 Heures (Antoine) Cette tâche était très importante pour pouvoir avoir un support de jeu. Premièrement j'ai donc essayé de chercher des maps déjà crée sur le site de Unity mais elles étaient trop détaillé pour notre jeu. J'ai donc décidé de les faire moi même grâce a Unity. Nous avons pour l'instant 4 maps. Les maps sont aussi utilisé dans les menus.

Recherches de solutions

"Click to move" : 3 Heures (Charles) Cette tâche était importante pour la compréhension du déplacement des unités dans notre jeu, même si elle n'a pas été implémentée directement. Cela consiste a déplacer un certain nombre d'unité jusqu'à un point X défini par le clic du joueur.

Compréhension du multijoueur : dizaines d'Heures (Charles) En effet, j'ai suivi des tutoriels qui, malheureusement étaient adaptés à des jeux différents du nôtre étant donné qu'il n'y a pas énormément de tutoriels disponibles sur l'implémentation du multijoueur pour les RTS.

Création d'un squelette jouable : 666 Heures (Tony)

Description Cette partie du travail se base sur le "bas niveau", au delà de tout aspect graphique. Il y a donc cylindres en guise de bâtiments, des sphère en tant qu'unités et un plan pour symboliser la map. Cela peut paraître primaire, mais le fait que ce cylindre se comporte en bâtiment, et que cette sphère se comporte en unité, fait de ce travail un succès.

Le joueur Avant toute chose, il a fallu créer un "objet" joueur. Celui ci contient un numéro de joueur, la couleur associée à ce celui-ci, le nombre total d'unités possédées, et leur force de combat (liée aux nombre de labos contrôlés).

Le bâtiment Il est constitué d'un cylindre, et d'une étiquette, le tout commandée par un script unique script. Grâce à ce dernier, la tour peut être initialisée avec un joueur, un nombre d'unités, et un type de bâtiment. Cette dernière prends alors la couleur de son contrôleur et l'étiquette affiche son nombre d'unités. [...]

De même, nombre de méthodes ont été créées, lui permettant d'être détruite, de changer de contrôleur, de changer de type (Gen/FW/Lab, actuellement sans effets), monter de niveau, etc. Un second script lui permet interagir avec les unités : d'être attaquée, de recevoir des renforts, mais aussi de déplacer des unités vers une autre tour (allié ou ennemi).

L'unité Cette dernière a été la plus complexe. En soit, elle n'est pas censé interagir directement avec le joueur (seulement par les bâtiments), donc ce ne sont que des capsules mobiles d'informations, contenant : le joueur auxquelles elles appartiennent, leur bâtiment d'origine et le bâtiment cible. Le problèmes viens du mot "mobiles", en effet, elles doivent disposer d'un I.A. de pathfinding (recherche d'un chemin vers une cible). Unity gère très bien cela tout seul grâce aux NavMeshAgents, mais la partie la plus compliquée est de le configurer correctement sur une entité faite pour être clonée (Configurer soi-même le NavMeshAgent est simple. Faire en sorte qu'un script le configure pour vous à chaque apparition d'entité est bien plus compliqué).

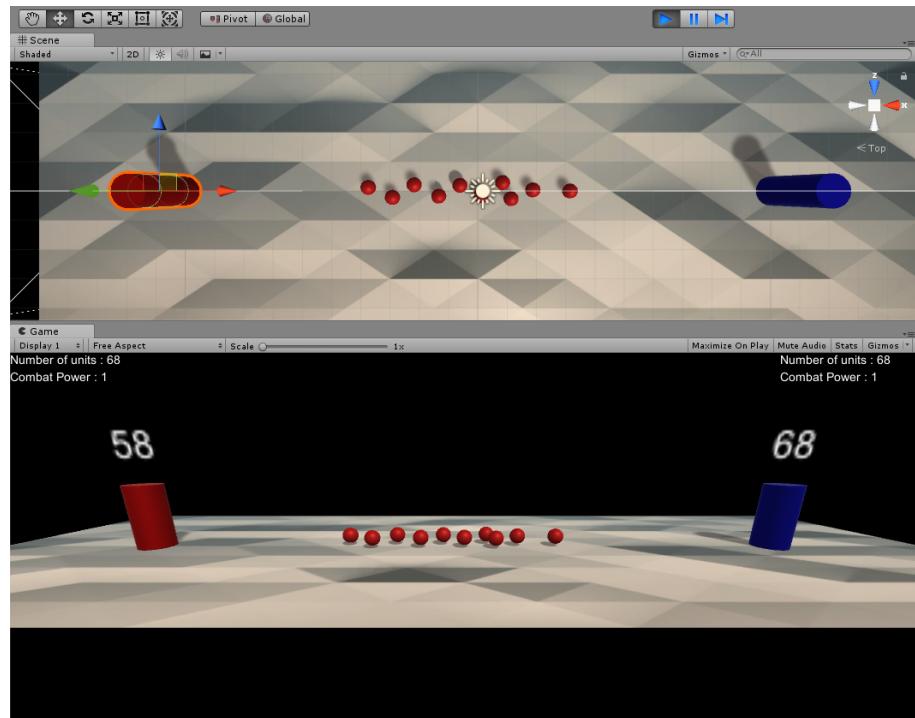


FIGURE 3 – Des unités rouges en chemin vers la tour bleue

8.2 Seconde soutenance (Vendredi 27 Avril)

Mécaniques bas niveau

Gamerules : 4 heures (Tony) Les gamerules sont un concept omniprésent dans les jeux indépendants. Elle permettent à l'utilisateur de personnaliser son expérience de jeu en modifiant les "règles" régissant le jeu. Ici, les Gamerules prennent la forme d'un script contenant toutes les valeurs "arbitraires" du jeu, tel que la vitesse de déplacement des unités, la portée/fréquence de tir du Pare-Feu, la vitesse de production des unités, etc. Ceci nous permettra de faire très aisément des modes de jeu différents : un mode "Quick Play" comprendra des générateurs plus rapides mais ayant des cap plus stricts, un mode "Expert" diminuera la vitesse des unités, augmentant les enjeux et punissant ainsi les erreurs des joueurs. Dans le cas éventuel où nous créerons un mode "Custom Game", l'utilisateur pourra lui-même choisir les dites valeurs, et le jeu lui obéira.

Job - Générateur : 2 heures (Tony) Le Générateur a été implémenté. Son rôle est de produire des unités. Par défaut, au niveau 1, il génère une unité toutes les 3 secondes (respectivement toute les 1,5 et 1 seconde pour les niveaux 2 et 3).

Job - Labo : 2 heures (Tony) Le Labo a été implémenté. Par défaut, chaque tour de ce type augmente la force de combat de 0.5 quelque soit le niveau, pour un bonus maximum de 1.5. (3 labos à 0 signifie qu'une unité du premier joueur combat 2.5 unités du second).

Changement de type : 1 heures (Tony) Le changement de type, aussi appelé conversion, permet à un joueur de convertir une tour d'un type en un autre. Lors d'une conversion, la tour concernée perd instantanément 30 unités. Pendant 5 secondes, elle est inopérationnelle. C'est seulement après ce temps qu'elle change de type et retombe au niveau 0 (Temps et coût modifiables dans les Gamerules). Cette méthode appelle un changement de modèle et de collision.

Level UP : 2 heures (Tony) Le level UP augmente les caractéristiques du bâtiment selon son type (hors labos). Le coût pour atteindre le level 2 est de 40 unités (50 pour le level 3). De la même manière que la conversion, la tour devient inopérationnelle pendant 5 secondes avant la prise en compte de l'augmentation. Cette méthode appelle un changement de texture.

Méthode Move : 4 heures (Tony) Cette méthode envoie un certain nombre d'unités d'une tour vers une autre. Cette méthode prend en argument un pourcentage, et calcule en fonction de la population le nombre d'unité à envoyer. Le joueur a le choix d'envoyer 25, 50, 75, ou 100% des unités comprises dans la tour.

Mécaniques haut niveau

Click on Tower : 7 Heures (Charles) Cette fonctionnalité permet que chaque joueur puisse seulement sélectionner ses propres bâtiments d'un simple clic gauche ("bâtiment de départ") et les unités adverses ainsi que les siennes d'un simple clic droit pour définir un "bâtiment cible". A partir de là, le script agit avec l'interface pour itérer des ordres sur les tours sélectionnées.

Feedback visuel : 3 Heures (Charles) Cette fonctionnalité est la suite logique du Click on Tower, d'une part quand on sélectionne un bâtiment de départ (clique gauche) une zone circulaire blanche apparaît sous le bâtiment, et disparaît si on clique autre part. D'autre part quand on sélectionne un bâtiment d'arrivée (clique gauche) une zone circulaire de la couleur du joueur auteur de la sélection apparaît, communiquant ainsi à l'adversaire que sa tour est prise pour cible, et par qui.



(a) Bâtiment d'origine



(b) Bâtiment cible

FIGURE 4 – Sélection des bâtiments

Interface (UI)

Implémentation de l'UI : 10 Heures (Antoine) Cette tâche était indispensable et devait se faire rapidement pour voir si le code de Tony marchait sur la map sans utiliser les touches du clavier. J'ai donc réalisé cette tâche avec l'aide de Tony pour pouvoir relier son code avec l'utilisateur. Pour cela 8 boutons ont été créés, 4 boutons pour déplacer les unités en % (100%, 75%, 50%, 25%) et ensuite 3 autres boutons pour changer les types des bâtiments (générateur, pare-feu, laboratoire) et pour finir un dernier bouton pour améliorer ces derniers. Deux boutons discrets sont ajoutés en bas pour quitter le jeu ou retourner au menu principal.

L'interface utilisateur sera aussi reliée à des touches sur le clavier permettant une jouabilité plus ergonomique et plus rapide. Par la suite, un écran d'affichage des scores sera ajouté en haut de l'écran permettant de savoir en temps réel quel joueur à l'avantage ou quel joueur perd (Histogramme concernant la population de chaque joueur).

Menu principal : 5 Heures (Antoine) Suite a la soutenance 1, nous avons trouvé que notre menu principal n'était pas du tout a notre goûts, nous avons donc décidé de tous refaire. Pour ceci, nous avons créé un menu avec les Maps de jeux en arrière-plans. Les Maps nous donnent donc une mise en bouche du jeu et donnent donc envie de plus découvrir le jeu et d'essayer les différents modes de jeu. Il y a donc le menu principal contenant des boutons permettant d'aller dans le multijoueur, singleplayer ou d'aller voir les settings.

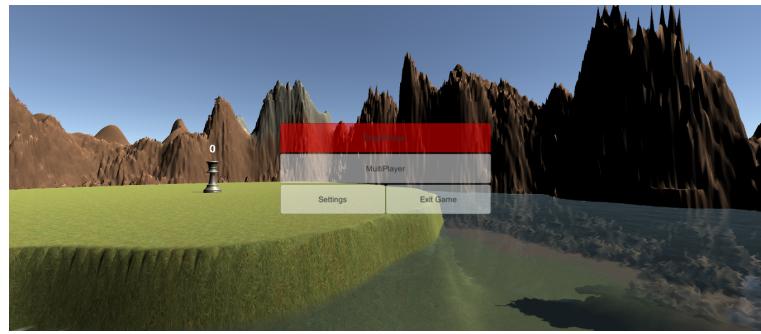


FIGURE 5 – Menu principal

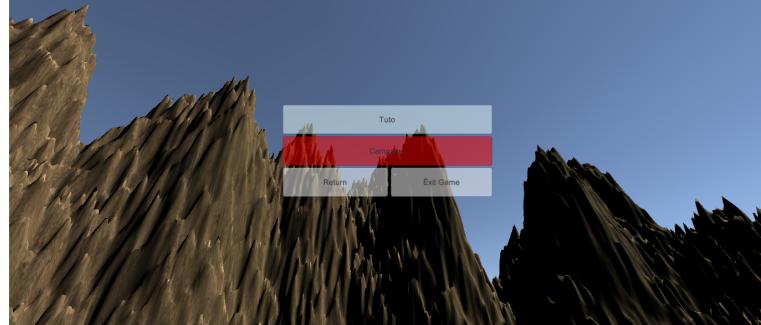


FIGURE 6 – Menu principal SinglePlayer

Nous avons aussi la possibilité à tout moment de quitter le jeu en appuyant sur Exit. Par la suite nous aurons d'autres menus permettant de configurer les parties, le multijoueur etc. Ces menus seront implémentés lors de la soutenance 3. Sur ces menus nous pourrons choisir la map sur lequel nous voulons jouer ou choisir la couleur du joueur etc. Cela permettra de créer sa partie personnalisée très facilement.

Settings : 1 Heures (Antoine) Pareil pour les settings, nous avons modifié rapidement les settings. Nous avons changer l'arrière plan et rajouter quelques fonctionnalités. Par la suite un autre menu settings fera son apparition pour pouvoir changer les touches de jeu etc... Les menus sont comme tous les menus du jeu, avec une décoration d'une des maps de jeu.



FIGURE 7 – Menu settings

Aspect graphique

Création des maps : 18 Heures (Antoine) Nous avons créé plusieurs maps différentes. Notamment une carte "Échec" destinée au tutoriel du jeu. Pour les autres Maps nous avons créé nos Maps manuellement car nous n'avons trouvé aucune map qui correspondaient à nos attentes. La plupart des Maps sont trop détaillés et ne rentrent pas dans nos attentes et beaucoup sont payantes. Nous avons donc créé les Maps grâce directement à Unity.

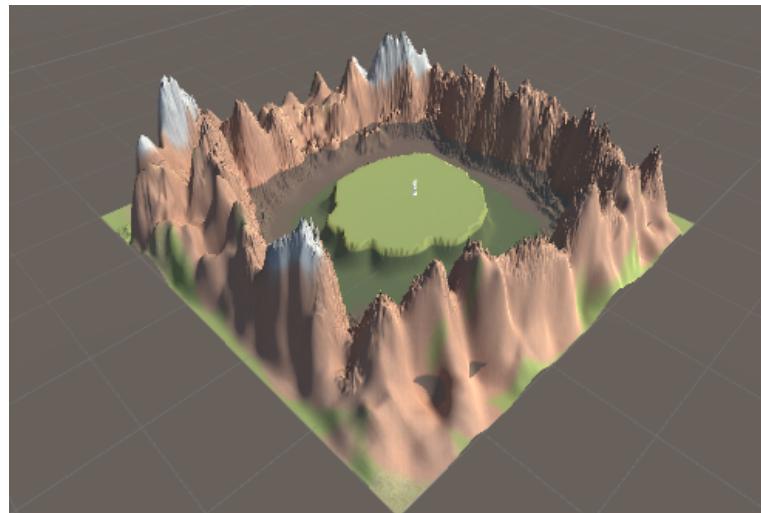


FIGURE 8 – Map "Montagne et Lac"

Implémentation des nouveaux modèles - Échecs : 1/2 Heure (Charles)

En effet, nous avons décidé de changer notre patte graphique, et de remplacer les anciens cylindres qui nous servaient d'assets par des belles pièces d'échecs (Vous trouverez une image en annexe).

Effet de particules : 12++ Heures (Antoine) Pour les particules, nous avons créé quatre particules différentes qui sont utilisées dans différentes occasions en fonction des actions suivantes :

Lors de l'absorption d'une unité par une tour, une particule blanches est jouée.

Lors d'une capture d'un bâtiment adverse, une particule de fumée apparaît qui englobe la tour, afin d'effectuer une transition lors du changement de couleur.

Lors d'une amélioration d'un bâtiment, pendant que le bâtiment est en construction, une animation est jouée pendant 3 secondes (particule de poussière brune). Ensuite lorsque le bâtiment à terminé sa construction avec succès une particule verte apparaît et marque la fin de l'amélioration en court.

Toutes ses particules permettent une meilleure vision du jeu et permettent de mieux comprendre ce qui se passe sur la map pendant que l'on fait une autre action.



(a) Construction



(b) Fin d'amélioration

FIGURE 9 – Exemple de particules

8.3 Troisième soutenance (Lundi 11 Juin)

Job - Pare-Feu : Projet (Tony) Un dernier type de bâtiments reste à implémenter, c'est le Pare-feu. Sans doute le plus complexes, du fait de son effet actif (par opposition aux effets passifs du Générateur et du Labo). Il devra détecter toutes les unités dans son rayon d'action, tirer un projectile autoguidé vers l'une d'entre elle (ce qui l'éliminera), et recommencer après un intervalle de temps défini. Son rayon d'action et sa vitesse de tire augmenteront avec son niveau.

Création d'un tutoriel : Projet (Tugdual) L'IA est une partie secondaire du projet. En effet, nous voulons nous focaliser sur le multijoueur pour permettre de faire quelque chose de très fonctionnel et d'agréable à jouer. Cependant, rien ne nous empêche de créer un tutoriel afin de pouvoir expliquer brièvement aux nouveaux joueurs toutes les mécaniques du jeu, et ainsi les lancer dans l'aventure qu'est poLy. Ce tutoriel sera donc créé avec des tâches effectuées avec une IA très simple. Nous aurons donc des tours à sélectionner, des tours à attaquer, des unités à déplacer, tout en gérant les attaques

Menu de Configuration : Projet (Antoine) Un problème a été rencontré dans la création du menu de configuration : les paramètres ne sont pas sauvegardés lorsque l'on sort de ce dernier. C'est un problème qu'il faudra adresser. De plus, le joueur devra pouvoir configurer son pseudo et la couleur qu'il arbore en arène. Il faudra donc stocker ses informations en local sur la machine.

Diversification des modes de jeu : Projet (Tony) Les bâtiments classiques ont été implémentés. Ils restent cependant les bâtiments spéciaux acteurs des différents modes de jeu : *The Hill* pour le mode *King of the Hill*, et les *Serveurs* pour le mode *Capture the Point*. Leurs implémentation nécessitera de créer de nouvelles classes de bâtiments, de nouveaux éléments d'interface pour communiquer les scores, ainsi que nouvelles conditions de victoire.

Multijoueur : Projet (Charles) Le multijoueur, étant une partie assez complexe à implémenter dans notre jeu, nous n'avons malheureusement pas encore commencé l'implémentation malgré les nombreux tutoriels regardés à ce sujet. Maintenant il nous reste encore plusieurs semaines avant la soutenance finale donc cette partie représente l'objectif majeur de cette troisième soutenance. Il faut savoir que cette partie, une fois terminée, permettra de faire jouer plusieurs joueurs dans une seule et même partie simultanément. Ces joueurs pourront jouer contre et le meilleur d'entre eux vaincra. En bref, le jeu sera terminé

9 Conclusion

9.1 Résultats du projet

Le projet n'étant pas encore arrivé à sa fin (et ayant pris le parti de ne rien inventer), cette doit rester malheureusement vide. Sauf cas de force majeure, ici s'ajoutera des lignes positives parlant du projet comme quelque chose d'abouti et de commercialisable. C'est notre but, et nous ferons tout pour y parvenir.

9.2 Les joies et les peines

Antoine

Les joies Durant les semaines de soutenance je me suis penché sur plusieurs tâches en même temps avec un logiciel que je n'avais jamais utilisé avant. J'ai donc découvert le logiciel avec de nombreux tutoriels ce qui m'a fait apprendre d'avantage le logiciel et permit d'apprendre des nouvelles choses sur le langage C sharp. Youtube a permis aussi de comprendre beaucoup de chose sur le logiciel et sur les tâches qu'il fallait faire. Par exemple le menu principal a été fait grâce à de nombreux tutoriel sur youtube. Grâce à l'aide de Tony j'ai réussi à régler les nombreux bugs de mes tâches et grâce à de nombreux tuto sur Youtube j'ai réussi à implémenter les particules et le menu principal sans aide et sans déranger mes camarades de travail.

Les peines Des difficultés ont été rencontré tout au long du projet lors de la découverte du logiciel et lors des différentes tâches comme les settings qui ont été compliqué à implémenter. Le travail de groupe a quelques limites comme le moment où Tony a effacé 3 fois mes particules car on avait pas bien compris l'utilisation de GitHub.

En somme Pour l'instant les semaines de projet m'ont permis de me surpasser dans le travail quotidien et dans les horaires de travail. J'ai réussi à apprendre beaucoup de chose ainsi que le travail de groupe et la répartition du travail.

Charles

Les joies Ce projet m'a permis de découvrir ce qu'étais le travail de groupe, étant donné que c'est une première pour moi. Se partager le travail, essayer de s'y tenir et surtout travailler ensemble résume plus ou moins nos semaines de rushs. De plus, le fait de découvrir un logiciel que nous n'avions JAMAIS utilisé auparavant, à savoir Unity, était intéressant sur le plan technique. Pour la première soutenance, j'avais un peu de mal à bien travailler étant donné que ce genre de projet était tout nouveau pour moi, par contre je pense avoir appris de cela et ai mieux travaillé pour la seconde soutenance.

A l'heure où j'écris ce bilan, il nous reste encore une (longue) semaine de rush pré troisième soutenance dans laquelle j'ai pour but de finir le multijoueur (complexe) afin que nous ayons un jeu plus que jouable.

les peines Pour revenir aux premières soutenances, le multijoueur à quand même représenté des difficultés pour moi, en effet malgré quelques tutoriels (adaptés à des jeux totalement différents du nôtre) que j'ai suivi je n'ai pas encore bien compris comment l'implémenter à notre jeu, ceci pourrait éventuellement représenter une des rares "peines" que j'ai eu durant ce projet. Pour continuer sur les peines, au début nous avions un problème de communication et de travail que nous avons (à mon sens) correctement corrigé pour la seconde soutenance.

En somme Ce projet était intéressant à réaliser (bien qu'il ne soit pas encore fini) sur le plan technique et humain, et le travail de groupe est ce que j'en tire de plus positif.

Tony

Les joies J'adore les problèmes. Du moins, j'adore trouver la solution aux problèmes. (Je ne suis pas le seul : si vous laissez un humain enfermé avec une corde nouée, celui-ci la dénouera. Puis, par pure ennui, il la renouera). Ceci est pourquoi j'adore la programmation.

Dans un TP basique, je suis bridé par le sujet (implémentation de LA solution dictée par les auteurs) et je dois suivre aveuglément ce dernier. Au contraire, j'ai été ici complètement libre de rechercher et d'exprimer ma créativité.

De même, il y a toujours ce sentiment de fierté lorsque que cette chose pour laquelle on a travaillé sans relâche aboutit à quelque chose de fonctionnel, telle on l'avait imaginée.

Les peines La régularité du travail est ce qui nous a fait le plus défaut. Nous avions de superbes ambitions, nous savions comment rendre ce projet excellent, mais nous ne disposions pas d'une détermination suffisamment grande pour implémenter cela. Pour moi, il nous manque cette personne moteur qui, malgré des compétences pas forcément exceptionnelles, porte les membres du groupes par sa pure motivation.

Lors des semaines de rush, je me suis surpris à travailler un volume horaire conséquent, de l'ordre de 10h - minuit. De même, mon souci de bien faire me pousse dans mes retranchements. Cependant, c'est en dehors de ces rush que la motivation est moindre, laissant la routine quotidienne prendre le pas.

Lors des prochains projets de grande envergure tel que celui-ci, je pense que des réunions hebdomadaires réservées au projet pourraient pallier aux problèmes ci-dessus, permettant l'exécution de petite tâches. Certes, ces dernières sont -par définition- rapides à compléter, mais elles se mettent sur la routes des tâches plus complexes lors de période de rush.

En somme Ce projet m'a permis de me surpasser techniquement, de réfléchir à des solutions pour des problèmes ouverts. Les leçons ont été apprises, et l'expérience retenue est largement positive.

Tugdual

Je ne sais pas vraiment quoi dire dans cette partie, donc je vais essayé de condenser mes ressenties sur ce projet du mieux que je peux.

Ce projet était tout d'abord assez nouveau, j'avais déjà eu l'occasion de faire des travaux de groupe, mais pas avec cette intensité. Mes derniers travaux était plus court dans la durée déjà, mais il mettait aussi beaucoup moins en pratique l'auto apprentissage.

En plus de cela, il y a aussi la confrontation de façon de voir les choses différentes, selon la personnalité de chacun, qui n'est pas souvent le plus simple à gérer. Mais malgré tout, un des éléments qui aurait pu paraître le plus dur à réaliser, à savoir l'idée principale du projet, est venu assez facilement, et à été adoptée très rapidement et à l'unanimité.

Après cela, il a fallu mettre en place un rythme de travail, mélangeant des rushs au dernier moment avec des périodes assez vite, que j'espère pouvoir homogénéiser au fur et à mesure du temps. Mais au final, en se motivant bien et en s'écoutant bien, je pense que cette dernière soutenance pourra être un final marquant pour ce projet.

J'espère pouvoir remettre en pratique certaines des maigres connaissances que j'ai acquise, que ce soit en gestion de projet ou en théorie pratique. En espérant que mes camarades en tirent une expérience eux aussi.

Je clôturerai alors mon résumé par les paroles d'un groupe de musique, The Clash, qui selon moi résume bien notre début de projet : "If I go there will be trouble, And if I stay it will be double"

Bonne fin de projet.

9.3 Derniers mots

Étant donné qu'il nous reste encore une longue semaine (et demi) de rush avant la dernière soutenance, nous pouvons avancer considérablement le projet. Mais pour conclure ce projet était particulièrement sympathique à réaliser, et nous avons tous appris sur le plan humain et technique. Nous aurions dû peut-être être plus régulier dans notre travail, malgré cela le projet s'est quand même bien déroulé. Sur ce, ne perdons pas plus de temps et à nos ordis !

(Cette section se verra complétée à la fin de la troisième soutenance)

Annexes

Inspiration



FIGURE 10 – StarCraft 2 : Le colosse des RTS



FIGURE 11 – Mushroom Wars 2 : Le RTS grand public

Illustrations

Modèles



FIGURE 12 – Modèle des Tours : Générateur, Labos, et Pare-Feu

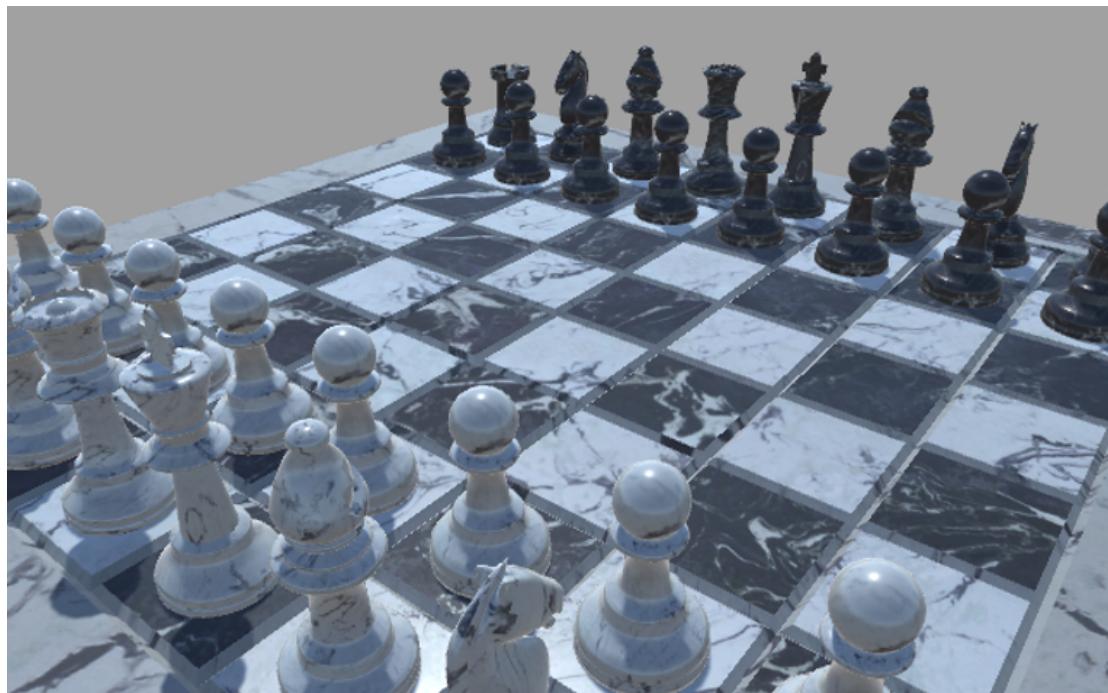


FIGURE 13 – Modèle des Tours : Modèle source

Interface

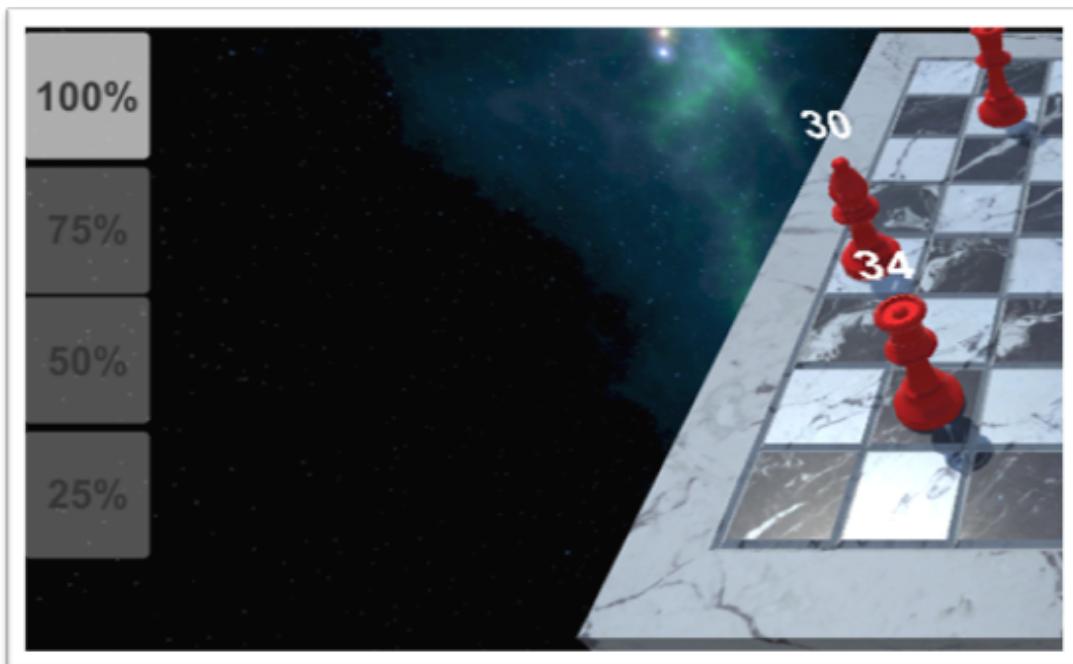


FIGURE 14 – Interface : Boutons de proportion (Déplacement d'unité)

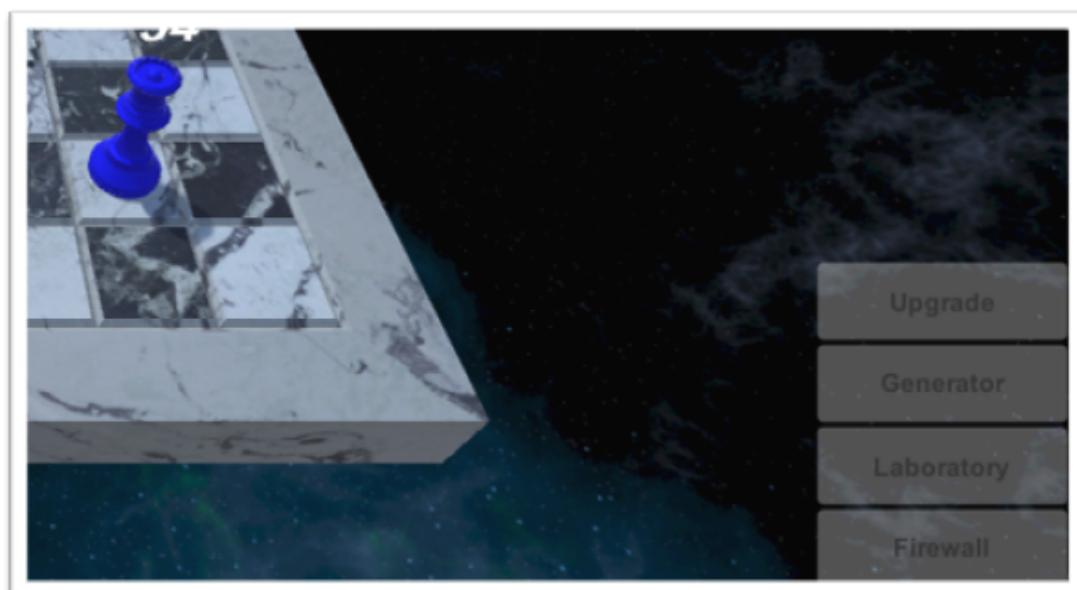
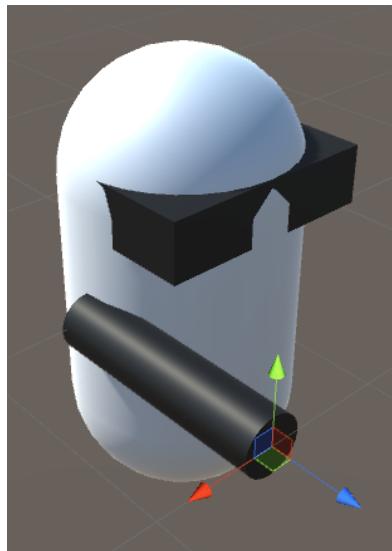
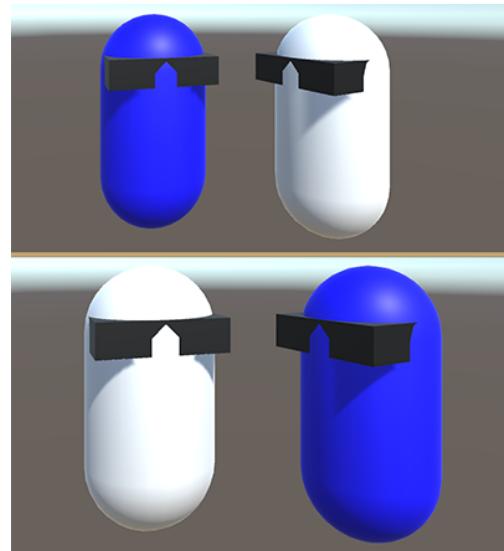


FIGURE 15 – Interface : Boutons de conversion et d'améliorations

Illustrations techniques



(a) Avatar utilisé pour le test du multijoueur



(b) Différenciation du joueur local (Host en haut, Guest en bas)

FIGURE 16 – Test d'un mode multijoueur (Jeu de tir)



FIGURE 17 – Gamerules : Ensemble de constantes modifiables

Communication



FIGURE 18 – Site : Page d'accueil

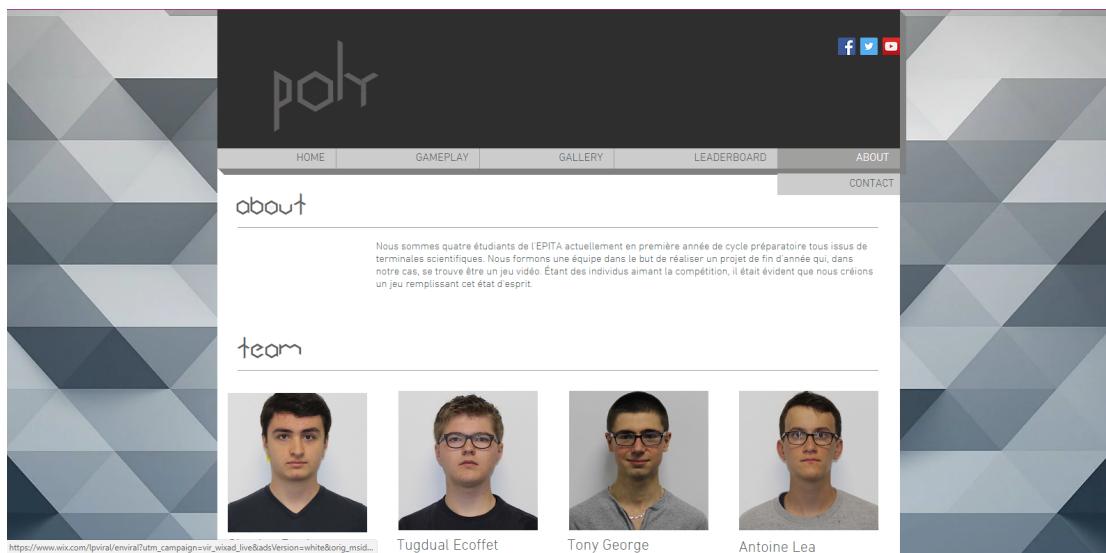


FIGURE 19 – Site : Page "A propos"