

# SPLASH!

---

Rapport final - SUP A2

Nicolas MAGNANI

Marième NDIAYE

Maxime ZHANG

Bruno ZHENG



## Table des matières

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>2</b>	<b>Préambule</b>	<b>5</b>
2.1	La Team . . . . .	5
2.1.1	Bruno Zheng . . . . .	5
2.1.2	Nicolas Magnani . . . . .	5
2.1.3	Marième Ndiaye . . . . .	6
2.1.4	Maxime Zhang . . . . .	6
2.2	L'Histoire . . . . .	6
2.3	Choix de gameplay . . . . .	7
2.3.1	Prémices du projet . . . . .	7
2.3.2	Choix final . . . . .	7
2.4	Répartition générale des tâches . . . . .	8
<b>3</b>	<b>Développement de Splash</b>	<b>9</b>
3.1	L'Interface . . . . .	9
3.1.1	Menu principal . . . . .	9
3.1.2	Menu Multijoueur . . . . .	10
3.1.3	Menu Scoryboard . . . . .	10
3.1.4	ATH . . . . .	11
3.2	Modes de jeu . . . . .	12
3.2.1	Mode Multijoueur . . . . .	12
3.2.2	Mode Solo . . . . .	13
3.3	Le Level Design . . . . .	14
3.3.1	Les Maps . . . . .	14
3.3.2	Le Personnage . . . . .	16
3.3.3	Sons et Musiques . . . . .	17
3.3.4	Conception des niveaux solo . . . . .	17
3.4	Le Gameplay . . . . .	18
3.4.1	Les Personnages . . . . .	18
3.4.2	Interaction avec la carte . . . . .	20
3.5	Le Réseau . . . . .	21
3.5.1	Schématisons . . . . .	21
3.5.2	uNet & PUN . . . . .	22
3.6	L'Intelligence Artificielle . . . . .	23
3.6.1	Bases . . . . .	23
3.6.2	Avancement . . . . .	23
3.6.3	Final . . . . .	24
3.7	La Communauté . . . . .	25

---

3.7.1	Le Site . . . . .	25
3.7.2	Les Plateformes Sociales . . . . .	27
<b>4</b>	<b>Avancée du projet - Apport des critiques</b>	<b>28</b>
4.0.1	Première Soutenance . . . . .	28
4.0.2	Deuxième Soutenance . . . . .	29
4.0.3	Soutenance Finale . . . . .	30
4.0.4	Rappel du planning des soutenances . . . . .	30
<b>5</b>	<b>Conclusion personnelle</b>	<b>31</b>
5.1	Bruno Zheng . . . . .	31
5.2	Nicolas Magnani . . . . .	32
5.3	Marième Ndiaye . . . . .	33
5.4	Maxime Zhang . . . . .	34
<b>6</b>	<b>Conclusion finale</b>	<b>35</b>
<b>7</b>	<b>Remerciements</b>	<b>36</b>
<b>8</b>	<b>Liens - Ressources</b>	<b>37</b>
8.1	Installation du jeu . . . . .	37
8.2	Manuel du jeu . . . . .	38
<b>9</b>	<b>Sources</b>	<b>39</b>
<b>10</b>	<b>Budget total</b>	<b>40</b>
10.1	Software . . . . .	40
10.2	Hardware . . . . .	40
10.3	Kit de survie . . . . .	40
<b>11</b>	<b>Index</b>	<b>41</b>

## 1 Introduction

Le studio **Thanos Inc.**, chargé du projet **Splash !**, est heureux de vous présenter notre rapport final.

L'équipe en charge est composée de quatre membres : Nicolas Magnani, Marième Ndiaye, Maxime Zhang et Bruno Zheng (chef de projet).

Elle s'est réunie autour de la même idée : créer à partir de rien un jeu qui saura plaire au goût de chacun tant dans la conception que dans la réalisation.

Réalisé à l'aide de l'infrastructure de développement **Unity**, Splash ! est un jeu de tir à la troisième personne (TPS) qui se déroule dans un univers cubique. Solitaire ou multijoueur, le principe est simple : recouvrir le plus de surface possible en peinture en un temps limité ! Ralentir l'équipe ennemie ou faire avancer la sienne ? Un choix est à faire pour gagner !

Bien que notre jeu se concentre principalement sur le mode multijoueur, nous vous proposons également un mode solo, dont le scénario vous sera explicité plus loin dans ce rapport.

L'originalité du jeu se base sur ses graphismes : entre Minecraft, Crossy Road et Cube World, l'univers cubique attire à attirer l'oeil.

Ce document est le fruit de cinq mois de travail et aura pour but de vous présenter l'étendu du projet créé.

## 2 Préambule

### 2.1 La Team

Notre équipe est constituée de quatre personnes provenant de la même classe, A2. Notre groupe s'est formé à l'origine par entente mutuelle, et par l'envie de créer un jeu vidéo. D'abord en désaccord, nous nous sommes très vite rassemblés autour de la même idée : un jeu de tir en multijoueur.

#### 2.1.1 Bruno Zheng

J'ai toujours été passionné par les jeux vidéo notamment les RPG alors en créer un dès la première année d'Epita semble être une opportunité pour nous les étudiants d'Epita, de montrer ce que nous sommes capables de faire. Ce projet nous permettra d'apprendre à travailler en équipe, d'utiliser les capacités de chacun à notre avantage, de pouvoir appliquer les connaissances que nous avons acquies au cours de l'année que ça soit par les mathématiques, la physique ou l'électronique mais aussi de pouvoir approfondir nos connaissances notamment en C#. C'est un projet qui nous tient à coeur. Ce ne sera pas facile, nous rencontrerons probablement de nombreux problèmes mais c'est ce qui nous poussera à aller plus loin. Nous sommes les créateurs et notre imagination est notre seule limite!

#### 2.1.2 Nicolas Magnani

Je m'appelle Nicolas Magnani, j'ai toujours aimé le monde des jeux vidéo, donc le projet d'en concevoir un ne peut que me ravir profondément. J'ai peu de connaissances en code pour l'instant mais c'est justement l'occasion pour moi d'en acquérir. Avant d'intégrer l'école, je m'étais déjà beaucoup impliqué dans les quelques projets en groupe que j'avais eus à réaliser au lycée. Ces projets furent pour moi très bénéfiques car ils m'ont permis d'acquérir une certaine autonomie. De plus, je suis passionné par tout ce qui touche au graphisme et j'aime manipuler des outils comme Photoshop, Blender ou dernièrement MagicaVoxel.

### 2.1.3 Marième Ndiaye

Ayant toujours eu un intérêt pour la programmation et la conception de jeux vidéo, je pense que ce projet me permettra d'acquérir plus de compétences en informatique. En effet, j'ai fait une terminale scientifique option science de l'ingénieur et, grâce à la spécialité ISN (Informatique et Science du Numérique), j'ai pu concevoir mon premier jeu vidéo en C. Avant EPITA, j'ai fait un semestre à la fac en MIPI (Maths Informatique Physique Ingénierie) et, j'ai également eu un projet où je devais créer un jeu en C. Hâte de commencer celui-ci en C# !

### 2.1.4 Maxime Zhang

Étant novice dans le domaine de l'informatique, la réalisation d'un projet aussi complet que celui de la conception d'un jeu vidéo représente pour moi un challenge à la fois colossal et très laborieux. Il s'agit là de notre tout premier, d'une longue série de projets et j'espère que le résultat sera à la hauteur de mes espérances.

Néanmoins, l'amateur de jeux vidéo que je suis est tout simplement ravi de pouvoir en créer un dès ma première année à l'EPITA. Rêve de gosse et éventuellement métier futur, la possibilité de réaliser un jeu de A à Z selon ses propres désirs est une occasion qui ne pourra qu'être bénéfique à mon épanouissement intellectuel.

Le travail en groupe quant à lui est un exercice intéressant, en totale adéquation avec notre formation d'ingénieur dont le but sera concilier les idées et propositions de chacun tout en restant dans une cohérence par rapport au thème initial.

## 2.2 L'Histoire

Tout se passait pour le mieux dans la ville colorée de Splash, les oiseaux chantaient, les gens riaient de bon cœur et le soleil rayonnait, jusqu'au jour où arrivèrent Greyness et ses affreux sbires. Ils y répandirent grisaille et morosité. La ville se meurt à petit feu sans que personne ne réagisse, faute d'avoir assez de courage pour se dresser contre le perfide Greyness qui était connu dans la région pour sa grande cruauté. Nous incarnons un jeune héros qui décide malgré tous les avertissements de ses semblables de s'opposer à ce mécréant qui s'était invité dans sa ville, autrefois si vivante. Notre protagoniste aura pour but de redonner à la ville sa beauté d'antan et fera face, dans sa quête, à Greyness et ses subalternes plus sinistres les uns que les autres.

## 2.3 Choix de gameplay

### 2.3.1 Prémices du projet

Les premières idées du projet n'ont aucun rapport avec le jeu actuel. Après un *brain storming* bien développé, si nous avons décidé de tout intégrer, nous aurions obtenu un melting pot bien garni.

Pour vous donner une idée, nous aurions fait un jeu d'horreur multijoueur où les joueurs pourraient y pratiquer le kung-fu contre des pandas, dans un environnement fun bercé par des arc-en-ciel et des licornes.

Évidemment, bien que ça paraissait vraiment convainquant, nous décidâmes de revoir les idées de chacun une par une afin d'y voir des points communs. Le plus important restait tout de même l'originalité et la cohérence du scénario et du jeu en lui-même.

### 2.3.2 Choix final

C'est ainsi que nous sommes partis sur un **jeu multijoueur**.

Pour la suite, tout nous vîmes facilement.

Fans de Splatoon, nous avons très vite fait ce choix de gameplay pour notre projet : une équipe de joueur doit battre l'équipe adverse à l'aide de peinture ; elle doit recouvrir le plus de surface possible en un temps limité.

Fans de Minecraft, notre choix artistique s'est porté sur ce jeu bien connu du public. Notre univers sera cubique aussi bien pour la carte que pour nos personnages ! Ce choix est aussi technique que graphique ; le fait de créer notre jeu bloc par bloc nous facilitera la mise en place de notre gameplay.

Nous y sommes, prêts à démarrer ce fabuleux projet. Avant de se lancer, nous nous sommes répartis équitablement les tâches ; répartition que vous pourrez voir dans le tableau ci-dessous.

## 2.4 Répartition générale des tâches

	Marième	Nicolas	Maxime	Bruno
Level Design		T	S	
Gameplay			T	S
Modèles 3D		T		S
IA	T	S		
Réseau		S	T	
Interface	S			T
Site	T		S	
Sons	S			T

Titulaire : T

Suppléant : S

Après délibération du gameplay, nous avons décidé de diviser le travail de façon équitable. Vu que nous sommes quatre, il nous paraissait impartial que chacun ait deux tâches principales à réaliser. Nous avons divisé notre projet en huit catégories, les mêmes que vous allez voir dans le développement du gameplay.

En divisant le travail en huit et en étant quatre, tout est super facile à répartir. Chacun avait donc deux tâches principales, et en est le titulaire; et deux autres tâches secondaires, et en est le suppléant. Le travail a été dispatché suivant les affinités avec la matière abordée.



## 3 Développement de Splash

Vous voici à présent dans la partie la plus importante, et sûrement la plus prépondérante de notre rapport final sur le projet Splash!. C'est ici que nous décomposons pièce par pièce notre jeu pour mieux vous le présenter.

### 3.1 L'Interface

Tout d'abord, les menus de notre jeu! C'est ce que vous voyez en tout premier quand vous le démarrez. Il en existe quatre différents.

#### 3.1.1 Menu principal

Le menu principal est comme son nom l'indique le tout premier menu que vous atteignez au lancement du jeu. Il se compose de 4 parties :

- **Jouer** : lance le joueur dans le mode histoire du jeu.
- **Multijoueur** : permet d'accéder au menu multijoueur du jeu.
- **Options** : permet au joueur de modifier les paramètres du jeu tels que le son ou la sensibilité.
- **Quitter** : ferme le jeu.

Nous avons voulu de ce menu qu'il soit le plus clair possible pour l'utilisateur. En effet, le joueur ne doit pas se sentir perdu dès ses premiers instants, ces premiers moments qui sont déterminants. S'il ne se sent pas à l'aise, il risque de quitter rapidement. De plus, le menu doit aussi montrer un avant goût du jeu, en démarrant, le joueur y découvre l'environnement et le thème du gameplay. Le but étant de lui montrer ce que le jeu peut lui offrir. Nous voulons que le joueur, en découvrant celui-ci, ait envie de continuer et puisse passer un agréable moment, seul ou avec ses amis. Le joueur découvre ainsi un environnement clair, ensoleillé, bercé par des nuages et un arc-en-ciel.

### 3.1.2 Menu Multijoueur

Le menu multijoueur est atteint par le joueur lorsque celui-ci clique sur le bouton "multijoueur" du menu principal. Ce menu est propre au mode de jeu. Il se décompose en deux blocs. Le thème reste dans la continuité du menu principal.

Le **premier** "bloc" permet au joueur de créer sa propre partie. Il peut lui donner le nom qu'il veut et un mot de passe s'il souhaite jouer en privé. Il a également la possibilité de sélectionner la carte sur laquelle il souhaite jouer. Il devient alors l'hôte du jeu.

Le **deuxième** "bloc" présente au joueur les parties déjà existantes et en cours. Il peut rejoindre le salon de jeu qu'il veut en double cliquant dessus, et y insérer un mot de passe si besoin. Un bouton lui permet de mettre à jour les parties nouvellement créées ou détruites.

### 3.1.3 Menu Scoryboard

Ce dernier menu est de loin le plus important pour le joueur. Il affiche le score de la partie en train d'être jouée. Il permet de rendre compte au joueur de son score total. Il prend en compte : le nombre d'adversaires qu'il a tué, son nombre de morts, et ses assistances quand il aide ses coéquipiers. Les données sont synchronisées en temps réel, du début jusqu'à la fin de la partie pour tout les joueurs. Il est dynamique et se met à jour automatiquement au cours de la partie. Ainsi le joueur pourra à tout moment au cours de sa partie afficher les résultats du jeu.

### 3.1.4 ATH

L'ATH pour l'acronyme de Affichage tête haute est un ensemble d'informations affiché en périphérie du centre de l'écran et renseignant le joueur sur son personnage ou son environnement. Notre ATH comprend :

- **Une barre de vie** : elle est située dans le coin supérieur droit de l'écran et indique l'état de santé du joueur. Cette barre de vie est dynamique représentant la vie du joueur en temps réel. Nous avons décidé de personnaliser cette barre de vie afin que celle-ci ne soit pas neutre mais au contraire, qu'elle soit accordée au thème du jeu. La barre de vie est de forme rectangulaire avec un fond vert. Elle indique la quantité de vie restante en pourcentage. Celle-ci a été implémentée de manière à ce que la quantité de vie diminue graduellement et non instantanément, rendant ainsi de notre point de vue le jeu beaucoup plus lisse.
- **Une jauge de peinture** : celle-ci se situe dans le coin inférieur droit et permet d'indiquer la quantité de peintures restante. Nous pourrions constater la présence de 2 types de données, la première indique la quantité de peintures restante dans le chargeur et la deuxième indique le nombre de chargeurs restant. Si le chargeur est vide, alors le joueur ne pourra plus tirer et il sera dans l'obligation de dépenser une recharge.
- **Un compte à rebours** : celui-ci démarre au commencement de la partie et indique le temps restant avant la fin de la partie. Il est actuellement fixé à 5 min par défaut.
- **Un curseur** : le curseur se trouve au centre de l'écran et évite au joueur de tirer à l'aveugle. Élément incontournable des jeux de tir, il permettra au joueur de viser et de tirer sur ses objectifs. En forme de +, le curseur ne doit être ni trop voyant, ni trop camouflé. Il doit simplement remplir son rôle sans gêner l'expérience du joueur.

## 3.2 Modes de jeu

Au lancement de notre jeu, le joueur a le choix dans le menu principal entre deux types de jeu : en solitaire ou en multijoueur avec des amis.

### 3.2.1 Mode Multijoueur

Ce mode de jeu est pour nous le plus intéressant. Il est à la base même de notre projet. Par équipe de **6 joueurs** maximum, le joueur doit affronter l'équipe adverse sur une carte à l'aide d'un pistolet et de billes de peintures. Nous avons choisis 6 joueurs pour des raisons de gameplay. La map n'étant pas d'une très grande taille, il a été préférable de limiter le nombre de joueur à 6.

Le principe est simple : l'équipe qui a disposé le plus de peinture de la couleur de son équipe sur la carte gagne la partie! Mais attention, il est possible au joueur d'attaquer ses ennemis afin de les éliminer et de les ralentir, cela permet à son équipe d'obtenir une avance considérable! Le joueur tué ne réapparaît pas subitement, il doit atteindre un délai de 5 secondes avant de pouvoir réapparaître. Lorsqu'il réapparaît, les statistiques sont remises à zéro, le joueur possède alors la quantité de munitions de base et sa barre de vie est à remise à 100%.

La partie démarre sur une séquence de plans de la carte puis sur l'apparition des joueurs dans leur camp respectif. Après le petit compte à rebours, le jeu démarre et les joueurs auront pour but de peindre un maximum de carrés. Élaborant des stratégies de siège plus rusées les unes que les autres ou fonçant dans le tas tels des sangliers déchainés, les joueurs feront face à des ennemis tout aussi enragés dans la guerre des territoires. À la fin du temps imparti, un résultat est calculé à partir du nombre de carrés totaux et l'équipe ayant peint le plus de surface remporte la partie.

À la fin de la partie, les caméras offrent une vue aérienne de la carte. Le résultat et le tableau des scores s'affichent et indiquent directement l'équipe gagnante. La partie se termine après le temps imparti fixé par défaut à **5 minutes**.

### 3.2.2 Mode Solo

Ce mode de jeu se rapproche d'un mode histoire et présente l'histoire de Splash.

Le joueur apparaît et découvre Splash remplie de noirceur. Tout n'est que grisaille et morosité. Le joueur aura pour but d'atteindre l'autre bout de la ville afin de défaire Greyness, le bandit responsable de l'empoisonnement de Splash.

Il fera face pendant son périple aux sbires plus sournois les uns que les autres de Greyness. Après avoir vaincu celui-ci, le joueur devra rechercher l'antidote caché sur la carte afin de redonner à Splash ses couleurs d'origine.

Le mode aura cependant quelques particularités par rapport au mode multijoueur :

- Se trouver sur un carré gris fait perdre de la vie.
- Un carré peint redeviendra gris au bout d'un certain temps.
- Tuer les sbires de Greyness fait remonter la jauge de peinture, il est également possible d'en récupérer à des endroits stratégiques de la carte.
- Le joueur perd la partie si sa jauge de vie atteint 0.

### 3.3 Le Level Design

Le level design est le processus dans la création de jeu vidéo qui s'occupe de la réalisation des niveaux, ou « cartes », de jeu (level ou map en anglais). Il s'agit ici de rendre compte de tout le travail qui a été réalisé pour créer notre univers. Nous y détaillons aussi les différents moyens mis en oeuvre pour y parvenir.

#### 3.3.1 Les Maps

La carte est selon nous l'élément clé de notre jeu et c'est ce qui nous a pris le plus de temps. D'abord trop simpliste, puis trop détaillée, il a fallu trouver le juste milieu pour la rendre agréable et fonctionnelle.

Notre toute première carte est une ébauche réalisée pour notre première soutenance. Fans du logiciel **MagicaVoxel** et surtout des travaux de **Sir Carma** (@ Sir\_carma sur Twitter ou saymygame.com), nous nous sommes inspirés de ses créations pour former notre carte.

A l'aide de **MagicaVoxel**, logiciel de création 3D, nous avons formé les différents éléments qui la composeraient : arbres, maisons, fleurs, étang, etc. Petit à petit, nous avons réussi à construire quelque chose qui nous semblait fonctionnel et jouable (voir index 1).

Mais beaucoup de détails rendaient notre map impraticable en multi-joueur. Des soucis de **colliders** (obstacles), de **shaders** (lumières traversant les arbres) nous obligèrent à la revoir entièrement. C'est pourquoi, à la suite de la première soutenance, nous partîmes sur un nouveau modèle de carte : plus grand mais plus simple au niveau des finitions.

Pour ce faire, nous lâchâmes **MagicaVoxel** sans regrets pour **Blender**. En effet, bien que très facile en terme de prise en main, **MagicalVoxel** est aussi très limité ; il ne permet que la création pure et dure de modèles 3D en bloc. **Blender** nous permet de mieux gérer le maillage du squelette et l'association des blocs un par un.

Concrètement, le modèle 3D de la carte est composé d'une multitude de carrés distincts indépendants les uns des autres, ce qui en fait un modèle très lourd mais nécessaire au principe de notre jeu, qui est de peindre ces dits-carrés.

Nous disposons actuellement de deux cartes pour le mode multijoueur, baptisées « **Turbo** » et « **Buildings** ». Il s'agit de cartes miroirs avec deux camps : respectivement bleu/orange et vert/violet. Dans chacune d'elle, les deux équipes auront pour but de conquérir l'entièreté de la carte en peignant le maximum de surface possible aux couleurs de leur armoiries.

La disposition actuelle de notre première carte, concentre la majorité des batailles au niveau du centre, où couloirs étroits et espaces fermés s'entremêlent pour entraîner davantage de morts. (index 2)

**Buildings** quant à elle se démarque par sa taille largement plus grande. Repartie sur 3 étages, elle offre en conséquence plus de possibilités aux joueurs dans leur stratégies, si celle-ci existent. Pour accéder au camp adverse, trois passages sont possibles : par le centre, par la gauche ou par la droite. Terrain plus grand veut aussi signifier plus de carrés. Ainsi, il est aussi plus difficile d'évaluer l'état de la carte et/ou de prédire quel équipe a l'avantage.

Autre aspect technique de la carte, les textures ont été créées en utilisant les logiciels de retouche **Paint** et **Paint.NET**. D'autres textures ont été directement importées depuis des assets Minecraft, c'est notamment le cas pour celles des portes, des arbres, des feuilles et des barrières.

### 3.3.2 Le Personnage

Nous avons rencontré des difficultés quant à la création de notre personnage. Comme la carte, une ébauche a été nécessaire pour mieux nous rendre compte de son implémentation dans notre univers.

A l'origine, nous nous sommes inspirés de l'univers carré et cubique du jeu **Minecraft**. Grâce à **MagicaVoxel**, nous avons réalisé un tout premier modèle de A à Z. Il correspondait parfaitement à l'ambiance graphique de notre carte (index 1). Bien que satisfaisant, il manquait d'animations et rencontrait les mêmes problèmes que pour la carte (shader, collider, etc.).

C'est pourquoi, comme pour la carte, nous avons recommencé notre héros depuis le début. Ce nouveau modèle a été conçu entièrement à l'aide du logiciel **Blender**.

De la création à l'animation, le plus dur auront été les étapes de **rigging** (déformation 3D du squelette) et de **skinning** (mise en place de la peau sur le squelette).

Chaque partie du corps a été dissociée pour faciliter l'exportation et la manipulation ultérieure sur Unity. Cette séparation des membres du corps facilite aussi la mise en place des différents scripts animés : le fait de ne travailler que sur les jambes par exemple ne va pas impacter le reste du personnage. Si un de nos scripts fonctionne mal, nous savons alors exactement quel membre est en cause. (index 4)



### 3.3.3 Sons et Musiques

Pour notre gameplay, nous avons décidé de mettre en place une musique reflétant l'univers de Splash!, celle-ci a été choisie suite à de longues recherches afin de trouver la musique idéale qui concorderai avec notre univers. Elles ont été téléchargées en ligne puis importées sur Unity. La musique est un facteur important du jeu afin d'y assurer au joueur une bonne immersion. Nous avons aussi sélectionné plusieurs autres fichiers audio comme celui de tir, de la recharge, de la mort, de la réapparition et du compte à rebours. Les musiques d'ambiances du menu et du jeu ont été choisies selon deux principaux critères : le rythme et le style. Il nous fallait une musique rythmée, sans non plus trop cadencée. Le but était de pouvoir la rejouer en boucle selon la longueur de la partie sans pour autant la rendre répétitif et alourdir l'expérience en jeu.

### 3.3.4 Conception des niveaux solo

La carte du mode solo a été conçue à partir de **Buildings**. Le terrain a été quelque peu modifié pour mieux coller au scénario de notre jeu. Il n'y est pas possible pour le joueur de se balader comme il pourrait le faire en temps normal sur Buildings en mode multijoueur. Celui-ci est dirigé à travers des couloirs fermés pour éviter toute incohérences dans le déroulement du scénario.

Ainsi, la carte du mode solo est une sorte de Buildings divisée en plusieurs sections : l'arène initiale, l'arène finale, et les différents couloirs intermédiaires, chaque sections comportant des ennemies à vaincre. La dernière arène mettra en scène le combat final contre le big boss du jeu.

## 3.4 Le Gameplay

### 3.4.1 Les Personnages

Les personnages ont été fortement inspirés de l'univers de Minecraft. Leurs aspects cubiques étaient ce qui correspondaient en effet le mieux au thème de notre jeu. Ainsi, le principal modèle 3D utilisé est celui du très célèbre Steve, une version de notre corps humain simplifiée en 6 membres : la tête, le corps, le bras gauche, le bras droit, la jambe gauche, ainsi que la jambe droite.

De ce fait, nous avons aussi voulu fournir un personnage très léger en termes de nombre de polygones pour ne pas surcharger davantage le jeu contenant des cartes déjà très lourdes de par leur conception.

Évidemment, plus le modèle comporte une grande quantité de polygones et plus celui-ci sera lourd, difficile à charger en jeu. A contrario, moins il en comporte et plus la fluidité du jeu sera améliorée, en excluant la puissance de la machine sur laquelle tourne le jeu.

Quant aux mensurations, elles ont été établies grâce à ce que l'on appelle dans Minecraft des « skins ». Le skin est une image 2D de dimension très petite et généralement au format Portable Network Graphics (.png) représentant le patron du modèle 3D qui lui est associé, comme si l'on avait déplié un cube sur une surface plane suivant ses arêtes. Sur celle-ci, il est possible de compter le nombre de pixels de chaque membre et ainsi obtenir par la même occasion les dimensions de ces membres.

Ces dimensions en nombre de pixels/carrés vous sont résumées dans le tableau ci-dessous :

	<b>Tête</b>	<b>Corps</b>	<b>Bras</b>	<b>Jambes</b>
Dimensions(h,l,p)	8x8x8	16x8x4	16x4x4	16x4x4

Grâce à cette image, le texturage du personnage devient aussi très simple puisqu'il suffit simplement de changer la couleur de chaque pixel du patron, ceci à l'aide de n'importe quel logiciel de retouche d'image.

Il est ensuite possible sur Blender à l'aide d'une fonctionnalité qui se nomme l'UV Mapping d'apposer cette image nouvellement texturée sur les membres du modèle 3D correspondants. Nous disposons actuellement de 4 skins pour 4 personnages de textures différentes pour 4 camps de couleurs bleue, orange, vert et violette.

Les personnages sont également dotés de plusieurs animations, chacune associée sur Unity à un déplacement ou à une action spécifique du personnage. Sans cela, le personnage se contenterait par exemple lorsqu'il se déplace de translater tout en restant statique.

Les animations réalisées sous Blender sont celles de la marche vers la gauche, vers la droite, vers l'avant, vers l'arrière, de la course, du saut, de la mort et de tir.

### 3.4.2 Interaction avec la carte

Exportés sur Unity, les cartes et les personnages se feront ensuite ajouter un composant « Collider » qui leur permettront d’avoir une surface solide. Cela empêche par exemple au joueur de traverser la carte lorsqu’il tombe d’une altitude plus élevée. Il s’agit aussi de l’élément de base à la conception de notre jeu.

Finalement, avec les outils en place, peindre la carte est pour le joueur assez simple. Prenons le cas de la carte de jeu Turbo où les deux camps sont de couleur respective bleue et orange.

Turbo est constituée d’une multitude de carrés de taille identique (5025 carrés exactement) contenant pour chacun d’entre eux un composant Collider et une texture. Lorsque le joueur clique sur sa souris et tire une bille de peinture, celle-ci est instanciée avec un script.

Ce script contient un code qui s’active lorsque la bille de peinture rentre en contact avec un carré de Turbo. Si c’est le cas, alors ce script a pour ordre de modifier la texture du carré en question en sa texture. Si la bille est de couleur bleue, c’est-à-dire tirée par un joueur bleu alors le carré devient de couleur de bleu. Le mécanisme est le même dans le cas d’une bille orange ou d’une couleur quelconque.

## 3.5 Le Réseau

Le réseau est la base de notre jeu. Comme dit dans le préambule, notre idée de départ était de créer un jeu de tir **multijoueur**. Le but du réseau est super simple : pouvoir créer un système permettant au joueur d'interagir avec d'autres ; l'idée principale étant d'arriver à faire se connecter tous les joueurs sur la même plateforme.

### 3.5.1 Schématisons

Pour débiter, nous avons commencer par créer un plan d'attaque : comment concrètement va fonctionner notre réseau ? Quel est le chemin type d'un utilisateur dans notre réseau ?

- **1 - Démarrage** : Le joueur arrive sur le menu **multijoueur** au lancement du jeu après avoir cliqué sur "multijoueur".
- **2 - Menu** : Dans le menu, le joueur fait face à deux choix : ou il crée sa propre partie ou il en rejoint une déjà en cours. Pour créer sa partie, il rentre dans le cadre prévu à cet effet son pseudo et un mot de passe s'il souhaite que la partie soit en privée. Sinon, il clique sur une partie déjà en cours pour la rejoindre.
- **3 - En partie** : Une fois dans le jeu, le joueur peut voir les autres joueurs à l'aide du menu dynamique prévu à cet effet. En effet, les noms des joueurs connectés au serveur y sont affichés. Si un joueur se connecte, alors une notification aura pour effet de l'informer d'un nouvel utilisateur entrant.
- **4 - Ragequit** : Un joueur peut quitter à tout moment la partie en cours, soit en appuyant sur alt + f4 comme tout bon joueur qui se respecte, ou plus simplement à l'aide du menu dynamique, et en appuyant sur quitter. Le joueur est alors de retour sur le menu principal.

Grâce à ce schéma, nous savions exactement de quoi nous avons besoin pour avoir un mode multijoueur digne de ce nom. La décision fut maintenant de savoir quelle infrastructure de réseau utiliser ? **uNet** ou **PUN** ?

### 3.5.2 uNet & PUN

Unity possède par défaut un système permettant de simplifier les échanges réseaux (Unity Networking ou simplement uNet). Nous avons donc choisi ce module par défaut.

Le développement du réseau n'a pas été la partie la plus facile de notre projet. Nous avons rencontré de nombreux problèmes quant à la gestion des différentes caméras : entre celles du joueur et de son adversaire, nous n'arrivions pas à les synchroniser.

Ces problèmes nous ont beaucoup ralenti. C'est pourquoi, très vite, nous avons pris la décision de nous tourner vers PUN. Par simplicité, et de par sa grande communauté, il nous a été plus facile de corriger cette erreur qu'en utilisant uNet. De plus, il a les avantages d'être rapide et plus rapide en terme d'actions réseau.

Grâce à **PUN**, le joueur est capable de se connecter au réseau multijoueur après avoir créé son propre salon ou en accédant directement à un salon préalablement fait par un autre joueur. Le jeu vérifie systématiquement si un nouveau salon a été créé, mais ne le fait pas de manière dynamique. C'est pourquoi, un bouton "**Refresh**" a été mis en place, afin que le joueur puisse mettre à jour la liste.

De fait, notre multijoueur nous semble actuellement performant. Nous avons choisi de limiter le nombre de joueurs par salon à 6, soit 3 par équipes afin de ne pas encombrer le réseau. En effet, si le nombre de joueurs est trop important, la jeu a tendance à se saccader rendant la partie injouable. Nous aurions pu ajouter un script à notre réseau pour fluidifier notre réseau, mais en plus de le surcharger pour rien, les cartes ne sont pas adaptées pour accueillir plus de 6 personnes.

## 3.6 L'Intelligence Artificielle

### 3.6.1 Bases

Le développement de l'intelligence artificiel n'a pas été de tout repos. Étant donné que notre jeu se base principalement sur le multijoueur, celle-ci, liée au mode solo, n'était pas une priorité. Nous avons commencé à développer l'IA à la deuxième soutenance dans le seul but d'accroître le sentiment d'immersion de l'utilisateur qui doit en tirer un sentiment positif. L'IA possédait des caractéristiques basiques tout comme le joueur de départ. Il était capable de marcher et de se tourner en direction des coordonnées du joueur. Bien sûr, les actions de l'IA sont aussi liées à ses mouvements.

### 3.6.2 Avancement

Peu à peu, nous constatâmes une réelle avancée dans le développement de l'IA. De nombreuses améliorations ont été mises en place. L'IA qui était capable de bouger, peut maintenant courir, attaquer au corps-à-corps et tirer des projectiles de peintures. De nombreuses fonctions ont été implémentées à l'IA. Celle-ci peut attaquer au corps-à-coprs si un joueur se trouve proche de lui, il peut aussi tirer des projectiles sur les joueurs proches dans le but de les éliminer. Ces attaques sont reconnues par les joueurs qui perdent de la vie. Les animations, telles que le tir ou l'attaque au corps-à-corps ont été réalisées sur Blender. Une fonction a été implémentée de telle sorte à ce que l'IA puisse réagir à la présence d'un joueur en fonction sa distance. Si le joueur se trouve à une très grande distance, l'IA est indifférente mais si le joueur s'approche, l'IA commence à se tourner vers le joueur et si le joueur est beaucoup trop proche, elle peut commencer à l'attaquer. L'IA possède aussi des caractéristiques qui peuvent être modifiées, il a une barre de vie, une vitesse liée au mouvement et une vitesse de course.

### 3.6.3 Final

L'IA réalisée est plutôt prometteur, celle-ci est très proche des IA rencontrées dans les jeux vidéo mais elle est aussi très simple. L'intelligence artificielle est maintenant capable de marcher, attaquer, tirer et courir. Les actions dépendent de la présence ou non d'un ennemi. Si un joueur est trop proche, l'IA peut commencer à se tourner vers celui-ci, l'attaquer ou lui tirer dessus suivant sa distance par rapport à lui. Tout comme le joueur, elle possède des caractéristiques propres à elle comme une barre de vie ; si la vie est à 0, alors elle meurt en effectuant son animation de mort puis elle disparaît. Elle possède aussi un nombre de munition limité ; s'elle n'a plus de munitions, alors elle sera obligée d'attaquer au coprs-à-coprs. Cette intelligence artificielle n'est que dédiée aux ennemis. Nous avons aussi décidé de mettre en place une nouvelle IA propre au boss du jeu. Le boss du jeu doit se différencier des ennemis de bases, ses attaques et ses mouvements sont propre à lui. Tout comme la première IA, il est capable d'attaquer, de tirer mais ses mouvements sont également plus rapides, ses attaques font plus de dégâts.

Le résultat obtenu est très satisfaisant, il concorde avec ce que l'on attendait. Permettant de donner vie au monde virtuel, son but principal est d'empêcher le joueur de ressentir un sentiment de "contrôle".



## 3.7 La Communauté

Pour grandir, un projet a besoin d'un public. Un jeu, lui, a besoin d'une communauté.

### 3.7.1 Le Site

Notre site internet est la vitrine de notre projet. À l'image de notre jeu, sa prise en main est tout aussi simple. En voici une courte description de celui-ci :

Il se décompose en plusieurs parties :

- **Accueil** : Bien évidemment, qu'est ce qu'un site accueillant sans une page d'accueil attirante, vous souhaitant la bienvenue et, vous présentant à la fois notre jeu et l'organisation du site en perspective. Dès cette page, vous aurez déjà envie de continuer à naviguer et à avoir plus d'informations sur notre jeu.
- **A propos** : Dans cette rubrique, vous y trouverez notre cahier des charges ainsi que le PDF de nos différents rapports de soutenance. Une partie de cette page est aussi réservée aux difficultés rencontrées durant notre avancement, sur la mise en place du jeu. Vous pouvez également trouver une présentation brève du rôle qu'a joué chaque membre du groupe.
- **Règles du jeu** : À l'image de ce qu'on vous a présenté dans ce rapport, il s'agit ici d'informer le joueur sur ce qu'il va trouver en téléchargeant notre jeu : les différents modes de jeu, et les règles qui s'y rapportent. Un lien est déjà disponible sur cette page pour qu'il puisse lancer notre projet directement chez lui.
- **Contact** : Il est possible de nous contacter directement via le site. Pour cela, il vous suffira de rentrer vos coordonnées ainsi que le message que souhaitez nous faire passer dans les champs requis. Vous pourrez ensuite cliquer sur "envoyer" afin que votre information nous soit transmise.

Grâce au site web, l'utilisateur va pouvoir suivre l'avancée de notre projet grâce aux différents onglets mis à sa disposition.

L'utilisateur peut également télécharger nos différents fichiers tel que le cahier des charges ou nos rapports de soutenance. L'utilisateur trouvera également plusieurs tableaux issues de nos soutenances où il pourra y voir les difficultés rencontrées lors de nos préparations.

Au niveau des moyens techniques utilisés, le site a été entièrement fait grâce à CSS3 et HTML5, et est hébergé sur Github. Son design flat correspond aux tendances actuelles et à notre jeu.

**[https ://thanosinc.github.io](https://thanosinc.github.io)**

### 3.7.2 Les Plateformes Sociales

Afin de populariser notre jeu, de nombreuses techniques de **growth hacking** s'offrent à nous (littéralement accélérer la croissance de notre jeu) : un meilleur référencement dans les différents moteurs de recherche, un accès à la plateforme **Steam** grâce à **Steam GreenLight** qui met en avant les jeux les mieux côtés par les joueurs, etc.

Mais tous ces moyens sont payants, et aléatoires. De plus, nous avons globalement dépassé le budget général accordé pour notre projet Splash.

C'est pourquoi, nous avons décidé de nous tourner vers les plateformes sociales. Il en existe beaucoup à l'heure actuelle mais seulement deux nous paraissent intéressantes : **Twitter** et **Facebook**. Ces deux réseaux ont l'avantage, en plus d'être gratuit, de toucher un public très large. En effet, si le contenu se trouve être intéressant par une majorité, il n'est pas rare de voir un effet boule de neige se produire à court terme, offrant ainsi une meilleure visibilité.

L'avantage de ces plateformes est aussi la **communication**. Tout se fait par l'intermédiaire d'une simple "page" qui regroupe toutes les informations propres au jeu. Le public peut interagir librement avec nous par le biais de "billets" qui s'affichent directement sur la page. L'échange est plus rapide et plus singulière qu'un mail ; le public se sent plus proche de ses développeurs et forme ce qu'on appelle généralement une **communauté**.

Sur ces deux pages, nous y posterons régulièrement les mises à jour de notre jeu. Il en existera deux types :

- **Les Changelog** : rendront compte d'un gros ajout de contenu dans le jeu. Très détaillés, ils mettront au courant le joueur de modifications importantes.
- **Les Patch Note** : seront réguliers mais moins conséquents. Ils rendront compte d'un ou plusieurs petits changements non majeurs dans la plupart des cas.

## 4 Avancée du projet - Appport des critiques

Dans cette partie, nous avons voulu mettre en avant l'évolution de notre projet au fil des soutenances et l'apport concret qu'elles nous ont apportées.

### 4.0.1 Première Soutenance

Pour la première soutenance, nous avons une ébauche très simpliste du gameplay de notre futur jeu. Nous avons créé une carte sur laquelle reposait notre personnage. Il pouvait s'y déplacer librement grâce aux touches du clavier et viser/tirer à l'aide de la souris. La carte, réalisée à l'aide de MagicaVoxel, notre tout premier logiciel de modélisation 3D, reflétait l'univers que nous voulions présenter : cubique et coloré. Les seuls obstacles que pouvait rencontrer le joueur furent les frontières délimitant la zone de jeu, et les différents éléments décorant la carte. En plus du jeu, nous avons une première amorce de notre site internet.

Le jury fut très réceptif à notre début de projet. Dans l'ensemble, nous étions satisfaits de notre réalisation mais nous savions que tout était encore à faire. C'est pourquoi à la suite de la première soutenance, nous avons décidé de revoir complètement les fondations de notre jeu, le jury ayant remarqué beaucoup d'imperfections dans notre travail initial.

Nous savions dès lors vers quoi nous focaliser en priorité : le multijoueur (principe de base de notre jeu) et le site internet.

#### 4.0.2 Deuxième Soutenance

Pour la deuxième soutenance, nous avons revu entièrement les bases de notre jeu. La carte, le(s) personnage(s) et les menus ont changé d'identité visuelle grâce au logiciel de modélisation 3D Blender. Il s'agissait pour cette présentation de rendre compte de la vision réelle de notre futur jeu. C'est pourquoi nous avons mis un point d'honneur à revoir le visuel général.

Outre l'apparence global, nous avons réussi à réaliser les travaux les plus importants de notre projet : le multijoueur et son gameplay. En effet, le principe de notre jeu est de peindre la carte ; le gagnant étant celui ou celle qui la remplira le plus de la couleur son équipe. Or, il fut complexe de réussir à rendre la carte compatible à ce type de gameplay, cette dernière ne formant qu'un seul "objet".

Nous avons finalement outrepassé cette difficulté grâce à Blender. Le multijoueur quant à lui a été réalisé à l'aide des assets présents dans Unity.

Tout semblait bien parti pour notre soutenance. Malheureusement malgré que tout était prêt, que nous ayons testé et retesté et que tout fonctionnait parfaitement comme nous le voulions, notre jeu n'a pas voulu se lancer pendant la soutenance. Nous avons tout de même réussi à présenter notre gameplay et notre site internet.

En outre, nous savons maintenant sur quoi nous focaliser pour la soutenance finale. Notre gameplay fonctionne, le visuel correspond à celui qu'on imaginait mais des détails sont à revoir comme l'IA. Nous aimerions aussi présenter du nouveau contenu, c'est pourquoi nous allons créer une deuxième carte pour le multijoueur.

### 4.0.3 Soutenance Finale

Pour cette ultime soutenance, nous voulons nous dépasser, impressionner nos évaluateurs. Malgré nos premières soutenances pas très concluantes, nous souhaitons prouver à notre public que notre jeu est digne pour être classé parmi les meilleurs de la promotion, et que nous avons les capacités de faire une présentation des plus professionnels pour réussir à faire vendre notre jeu.

Nous en avons les moyens. Notre jeu est sur le point d'être abouti et offre un large panel de divertissement, tant par son gameplay, par son ambiance ou par ses graphismes.

### 4.0.4 Rappel du planning des soutenances

	Soutenance - 1	Soutenance - 2	Soutenance - 3
Level Design		★	★★★
Gameplay	★	★★	★★★
IA		★	★★★
Réseau	★	★	★★★
Interface	★	★	★★★
Site	★	★★	★★★
Sons		★	★★★

Ebauche : ★

En Progression : ★★

Achevé : ★★★

Pour achever cette partie, voici le tableau qui nous a servi de repère soutenance après soutenance. Il y décrit la progression voulue dans chacune des parties de notre projet pour chaque soutenance.

## 5 Conclusion personnelle

### 5.1 Bruno Zheng

Etant chef de projet pour la première fois je me rend compte que ceci n'est pas de tout repos. La patience est primordiale, je pense que la plus grande difficulté est de réagir différemment suivant le type de personne. Vu mon rôle, je m'aperçois peu à peu que le chef doit tenir plusieurs rôles, veillez à ce que le travail soit fait n'est pas son seul job même si celui-ci est très importante. D'après l'expérience acquis, je pense que le chef de projet doit agir obligatoirement en tant que leader du groupe. Il doit immanquablement savoir prendre des décisions dans le bien du groupe et du projet et non pour le bien de l'individu seul. Faire en sorte d'éviter les conflits entre camarades, tenir les membres au courant des réunions, résoudre des conflits du groupe rencontrés dans le déroulement du projet et le plus important faire confiance à ses partenaires. Certaines fois, on ressent de la difficulté de jouer le rôle d'un camarade, d'un ami et en même temps celle du chef de projet. Si une personne n'a pas réalisé le travail demandé conduisant au retard du groupe, on se doit d'être objectif et donc de réprimander la personne, ce qui aussi, de mon point de vue, est l'un des plus grands inconvénients de ce rôle car celui-ci peut être opposé à notre propre personnalité.

Le travail en équipe a été très enrichissant, au départ il était difficile de se concorder mais peu à peu nous nous sommes mis sur la même voie. Le projet m'a permis d'acquérir de nombreuses connaissances notamment sur Unity et en C#. Celui-ci m'a permis aussi d'acquérir plus de rigueur au niveau de mon travail. Capable de jongler entre les cours, les devoirs, les activités personnelles et le projet est très difficile. On constate que le temps nous manque. Ainsi avec l'expérience acquise, je suis capable de recréer ued IA ou un petit jeux video sur Unity sans problème, grâce au travail effectué en à peine 5 mois.

Finalement je suis très satisfait de l'ensemble du projet, nous avons rencontré de nombreuses difficultés qui, quelle que soit la taille, ont été résolues. Nous avons accompli nos objectifs et nous sommes même allés au-delà de nos attentes. En tant que chef de projet je suis très fière de ce que nous avons accompli, sans le travail d'équipe, rien n'aurait été possible. Mes camarades et amis de projet ont été très sérieux et professionnels au niveau de leurs travaux, la mise en forme de notre jeu est aussi la preuve de ce qu'une équipe de 4 membre, ne partageant pas forcément les mêmes passions, est capable de réaliser.

## 5.2 Nicolas Magnani

Je suis fier, fier de notre projet, fier de ce que nous avons réussi à créer en 5 mois.

Débutant en programmation, je ne pensais pas qu'on puisse être capable d'un tel résultat en si peu de temps. Ce fût compliqué au départ, de trouver une idée, la bonne idée, puis de se mettre d'accord entre nous et ensuite de s'y investir du mieux qu'on peut.

Habitué à travailler par et pour moi-même, je n'avais jamais pensé que ce travail de groupe serait si enrichissant tant au niveau personnel que professionnel. J'ai réussi à me dépasser dans mon travail, à me donner toujours plus dans les périodes de rush, et à tenir durant deux nuits blanches consécutives (succès déverrouillé).

Ce travail de groupe est très formateur, il m'a permis de me rendre compte que la clé de la réussite d'un tel projet ne sont pas toutes ces lignes de code qui font fonctionner notre jeu, mais la communication. Sans communication, il est très dur de faire avancer un tel projet. De manière générale, notre groupe est resté soudé dans l'adversité. Je dis globalement car il y a toujours des tâches qui noircissent le tableau, et s'il y a bien quelque chose que j'ai appris à cause, ou grâce à ça, c'est qu'il ne sert à rien de blâmer les autres, le projet n'avancera pas plus vite, au contraire !

Plus concrètement, je suis content de mes progrès au niveau technique : j'ai progressé en C#! De plus, j'ai découvert des nouveaux logiciels comme MagicaVoxel, Blender, Unity et bien sûr LaTeX. Tous ces softs me seront certainement utiles pour de futurs projets.

Tous les moments passés à réaliser ce projet restent pour moi de bons souvenirs et je ne peux que remercier tous les membres de mon équipe pour ce que nous avons accompli.



### 5.3 Marième Ndiaye

Au départ sceptique à l'idée de créer un jeu entièrement fait et imaginé par nous même (bien que son concept soit légèrement inspiré du célèbre jeu "Splatoon"), je n'avais auparavant jamais touché aux jeux videos mis à part, Candy Crush ou encore Sims City.

Je me sentais très novice et ne connaissais le fonctionnement d'aucun jeu "type Epitéens" comme dirais-je. J'entendais des mots tel que FPS, ou TPS quand avec mes collègues, débatsions sur le type de jeu que nous allions choisir et, je n'en connaissais nullement leurs significations. Je ne m'y connaissais donc pas du tout ce pourquoi, je me sentais au départ perdu ce qui provoqua un démarrage, une prise en main et une implication plus lente que mes camarades. Cependant, grâce à eux et, à leurs forces de persuasion, ils ont réussi à me faire aimer ce monde.

Au fur et à mesure, je me sentais de plus en plus à l'aise et de plus en plus impliquée dans notre travail. Dès la première soutenance, je me sentais fière de présenter le fruit de nos efforts et des miens plus précisément. Je ne me pensais pas capable d'arriver à un tel résultat dès le départ. Pour la seconde, je me suis occupée d'une chose des plus importantes pour notre jeu, j'y suis arrivée et ce, pour ma plus grande fierté. Pour cette dernière qui arrive, j'espère me dépasser plus que les précédentes.

Ainsi, grâce à ce projet, j'ai pu retrouver une certaine confiance en moi et, j'ai également pu repousser mes limites notamment, ceux au niveau sommeil car, cette expérience m'aura initié aux nuits blanches à l'école pour terminer et présenter un jeu des plus structuré qu'on ne pouvait imaginer.

## 5.4 Maxime Zhang

Énormément de choses se sont passées depuis le premier cahier des charges. Il est clair que depuis le début de ce projet, j'ai acquis beaucoup de notions, que ce soit par le visionnage rabâché de vidéos explicatives sur Youtube ou par celles assimilées au travers des nombreux TP de programmation pendant toute la durée du semestre.

Certains travaux que j'avais du mal à réaliser me posent aujourd'hui plus aucun problème. Après une pratique répétée, je suis à présent capable de concevoir des modèles 3D et de les animer sans difficultés. D'autres missions me prenaient un temps considérable et j'ai appris au fur et à mesure à trouver des solutions pour y remédier. C'est un gain de temps et de productivité considérable qui me permet de tout de suite me diriger sur d'autres sujets essentiels au projet.

Le travail en équipe sur un projet aussi gros n'était pas non plus de tout repos. Après ces cinq derniers mois, je m'aperçois que la coordination entre l'avancement de chaque membre est un point non négligeable à prendre en compte pour garantir le bon déroulement d'un projet de cette envergure. Cela a d'abord été mon cas, après des débuts difficiles à cause de retards dans mon planning, j'ai su me ressaisir pour ne pas handicaper mes camarades.

Finalement, nous avons obtenu un résultat qui convenait encore plus à nos attentes initiales. Nous avons été capables de mettre en forme tout ce que l'étendue de notre imagination pouvait nous laisser offrir. Et rien que pour cela, ce projet représente pour nous notre fierté. Il a été pour moi une expérience enrichissante largement bénéfique à mon développement intellectuel.

## 6 Conclusion finale

Si nous devons retenir une seule chose de ce projet, c'est la puissance formatrice qu'il nous a inculquées.

Novices, nous nous sommes lancés dans l'aventure sans trop savoir où ça nous mènerait. Cinq mois plus tard, il nous a changé.

Professionnellement, ce projet nous a permis de renforcer nos bases dans la programmation, particulièrement en C#. De plus, nous avons appris à maîtriser rapidement des logiciels de tous types comme Unity et Blender, pour ne citer qu'eux.

Outre l'aspect technique, il nous a offert la chance de travailler en groupe, avec ses avantages et ses contraintes. L'organisation, la communication, l'entraide et la patience sont pour nous les éléments forts de ce projet.

Nous sommes tous les quatre d'accord pour dire qu'individuellement nous avons beaucoup appris. Chacun a su tirer partie à sa manière de cet investissement collectif : de la découverte du travail en groupe aux rushs nocturnes à l'école, ce jeu nous a sorti de notre zone de confort à de nombreuses reprises, au point de nous surprendre à certains moments.

Cette aventure humaine a été longue et intense. Le fruit de ses efforts ne sera pas vain : ce projet nous a été très formateur. Conscient de nos qualités et défauts, nous avons su tirer le meilleur de nous même pour rendre compte de notre idée de départ. Ce projet, c'est plus qu'un jeu. Tout ce que nous avons acquis aujourd'hui nous sera très bénéfique pour demain.

Nous sommes fiers de ce que nous avons accompli aujourd'hui.

## 7 Remerciements

Enfin, nous voulons remercier les différents protagonistes qui nous ont épaulés de près comme de loin dans notre aventure.

Nous tenons tout d'abord à remercier chaleureusement et amicalement nos deux ACDCs sans quoi nos début sur Unity auraient été plus fastidieux.

Nous voulons aussi remercier nos deux confrères de S3# : Gabriel Duqué et Antoine Martin de nous avoir donné de leur temps pour le développement de notre site web.

Merci à EPITA de nous avoir offert l'opportunité de prendre part à un tel projet et de nous offrir leurs locaux pour couvrir nos rushs nocturnes.

Merci au corps professoral d'EPITA de nous avoir fait découvrir les joies du développement depuis le début de l'année, particulièrement à Gollum d'avoir pris le temps de nous donner ses critiques et opinions durant les deux premières soutenances qui nous ont beaucoup apportées pour le rendu final.

## 8 Liens - Ressources

Dans cette partie, nous allons vous décrire les différentes étapes pour installer, dans un premier temps, notre jeu, et ensuite pour en comprendre son fonctionnement.

### 8.1 Installation du jeu

Rien de plus simple que d'installer notre jeu :

- **Téléchargement ?** : En premier lieu, il vous faut acquérir notre jeu gratuitement. Deux solutions s'offrent à vous : soit vous le trouvez en CD (rare), soit vous le téléchargez directement depuis notre site internet (<https://thanosinc.github.io>). Sur notre site, il suffit d'aller dans la catégorie "téléchargement" et de cliquer sur "Splash!", tout simplement. Qu'importe le support, vous voilà maintenant en possession de notre jeu. Passons maintenant à l'installation.
- **Installation** : Lancer maintenant la procédure dite d'installation du jeu. Cliquez sur Splash.exe afin d'ouvrir la boîte de dialogue qui amorce le début de l'installation. Lisez attentivement la charte de notre jeu avant de passer à l'aspect technique de la procédure. Cliquez sur "suivant" après avoir accepté les conditions d'utilisation, et choisissez maintenant l'endroit où vous voulez installer le jeu. Enfin, cliquez sur installer, tout simplement.
- **Fin de procédure** : Lorsque la barre verte a fini de se remplir, vous pouvez cliquer sur "quitter" dans la boîte de dialogue. Vous avez terminé l'installation de Splash! Vous pouvez maintenant passer au manuel du jeu afin de comprendre le fonctionnement du jeu de façon simple et claire.

## 8.2 Manuel du jeu

Nouvel acquéreur de **Splash!**, vous vous demandez donc comment jouer ! Rien de plus simple ! Voici une petite liste non exhaustive des commandes de notre jeu. Elle n'est pas complète car elle a tendance à être augmentée au fur et à mesure des mises à jour de notre jeu. Les mises à jour seront directement téléchargeables depuis notre site.

Après avoir lancé notre jeu en double cliquant sur l'icône, vous voilà dès à présent devant le menu principal. Deux choix s'offrent à vous : soit vous préférez l'histoire, alors nous vous conseillons le mode solo, soit vous voulez jouer contre d'autres joueurs, alors le mode multijoueur est fait pour vous. Il est préférable de commencer par le mode solo pour bien prendre en main le jeu.

Dans tous les cas, voici les commandes pour gérer au mieux votre partie :

— **Clavier** : Le clavier est l'outil indispensable pour contrôler la partie.

> Les touches directionnelles :

- » Gauche et Droite : bougent votre personne à gauche et à droite
- » Haut et Bas : font avancer et reculer votre personnage

> Les touches classiques :

- » Espace : fait sauter votre personnage
- » Tabulation : permet de voir le score de la partie
- » Echap : permet d'accéder au menu de la partie pour gérer les options ou quitter

— **Souris** : permet de faire bouger le viseur de l'arme du personnage et de tirer. Pour tirer, il suffit simplement d'appuyer sur le bouton gauche de la souris.

Vous avez dès à présent tout ce qu'il vous faut pour vous amuser. A vous maintenant d'être le meilleur !

Bon jeu !

## 9 Sources

Pour réaliser notre projet, nous avons été amenés à utiliser bon nombre de logiciels :

- **Unity** : Il est le logiciel clef de notre projet. A la fois support graphique et technique, il est l'élément clef de notre réussite. Unity permet par le biais de son moteur graphique, la mise en place de jeux 3D/2D.
- **Blender** : nous a été d'une grande aide pour la modélisation de nos personnages et de nos cartes.
- **MagicaVoxel**
- **Photoshop** : nous a été utile pour façonner les textures.
- **Paint.Net** : nous a permis de texturer également certains modèles.

## 10 Budget total

### 10.1 Software

Outil	Description	Prix
Unity3D v5.5	Modélisations et animations 3D	Free
Blender	Assets	Free
MagicaVoxel	Map Creation	Free
Github	Code Sharing	Free
Audacity	Mixage Son	Free
Visual Studio 2015	Game Development (Text Editor)	Free
Photoshop CS6	Digital Retouching	1 000€
Monodevelop	Game Development (Text Editor on Linux)	Free
Overleaf	L <sup>A</sup> T <sub>E</sub> X(Text Editor)	Free

### 10.2 Hardware

	Modèle	CG	CPU	Prix
Nicolas	Dell	Nvidia GTX 960M	Core i7	1 100€
Marième	Asus	Nvidia GTX 840M	Core i5	1 000€
Maxime	Apple	Intel HD Graphics 6000	Core i5	1 400€
Bruno	Razer Blade	Nvidia GTX 1060M	Core i7	2 000€

### 10.3 Kit de survie

- Chips : 2.99€
- Gateaux : 1.5€
- Pizzas : 7€
- Ramens : 0.99€
- Soda : 1€
- Thé à la camomille : 0.70€
- Eau : 1€
- Boisson énergisante : 2€

**Kit de survie total : 17,18 euros \* 6 mois \* 30 jours = 3092.40€**

**Hardware total : 5 500€**

**Software total : 1 000€**

**Total : 9 592,40€**



## 11 Index



