

# CAHIER DES CHARGES

## FONCTIONNEL

RELATIF A LA CONCEPTION D'UN JEU DE TIR EN 3D  
IMPLÉMENTANT UN MODE MULTIJOUEUR EN  
RÉSEAU

**“ PATH MAKER ”**

---

Paul CREDOZ	E2
Charles-Antoine LEGER	E2
Clément LABBÉ	E2
Pierre-Louis POZZO DI BORGO	E2

---

# Sommaire

<b><u>I - Contexte et définition du projet</u></b>	<b>3</b>
I.1 - Origine et nature du projet	3
<i>I.1.1 - Nature</i>	3
<i>I.1.2 - Origine</i>	3
<i>I.1.3 - Objet de l'étude</i>	4
I.2 - Présentation de l'équipe	4
I.3 - Etat de l'art et positionnement du projet	5
<b><u>II - Périmètre fonctionnel</u></b>	<b>7</b>
II.1 - Difficultés et risques identifiés	7
<b><u>III - Définition fonctionnelle des besoins</u></b>	<b>10</b>
III.1 - Spécifications fonctionnelles	10
<i>III.1.1 - Gameplay</i>	10
<i>III.1.2 - Mode multijoueur</i>	11
<i>III.1.3 - Mode solo</i>	11
<i>III.1.4 - Magasin dans le jeu</i>	12
<i>III.1.5 - Gestion de profil</i>	12
<i>III.1.6 - Paramètres</i>	12
III.2 - Spécifications techniques	13
<i>III.2.1 - Technologies</i>	13
<i>III.2.2 - Méthodologies</i>	15
<b><u>IV - Planning prévisionnel</u></b>	<b>16</b>
IV.1 - Différentes parties du projet	16
IV.2 - Répartition des parties	16
IV.3 - Planning	17
<b><u>V - Budget</u></b>	<b>18</b>
V.1 - Coûts du projet	18
V.2 - Objectifs de bénéfices	18
<b><u>VI - Conclusion</u></b>	<b>18</b>

# CONTEXTE ET DEFINITION DU PROJET

Cette première partie du cahier des charges vise à définir le projet relatif à celui-ci. Elle expose la nature du projet, son origine mais aussi de façon plus générale le contexte dans lequel il s'inscrit.

## **I.1 - Nature et origine du projet**

### **I.1.1- Nature du projet**

Le projet est un jeu vidéo de tir à la troisième personne avec un mode de jeu de siège en 3D et en multijoueur. C'est un logiciel standalone, développé majoritairement en C# sur le moteur de jeu vidéo gratuit Unity 3D. Il est possible d'y jouer sur ordinateur sous les systèmes d'exploitations MacOS et Windows à l'aide d'un clavier et d'une souris ou bien d'une manette Sony ou Xbox.

C'est un jeu d'action et de réflexion. Il se joue à plusieurs en réseau ou seul mais face à des ennemis contrôlés par l'ordinateur.

### **I.1.2 - Origine du projet**

Afin que ce jeu ne soit pas un copier-coller de jeux déjà existants, nous avons besoin d'une idée innovante. Nous avons donc basé nos réflexions sur des fonctionnalités liées à la perception du joueur, notamment sur l'ouïe mais aussi sur l'espace et le temps dans le jeu. De nombreux jeux axés sur l'espace tels que Portal (basé sur la téléportation) existants déjà, nous nous sommes tournés vers l'écoulement du temps.

Bien qu'il existe aussi des jeux tels que "Lemnis Gate" avec des boucles temporelles ou encore "Super Hot" dans lequel le temps ralentit pour le joueur, nous avons imaginé un jeu où l'écoulement du temps serait spécial, uniquement les objets, en particulier les balles seraient ralenties, contrairement au joueurs qui ne seraient quant à eux pas impactés.

### **I.1.3- Objet de l'étude**

L'objet relatif à ce cahier des charges s'inscrit dans le projet du second semestre de la première année d'étude à l'EPITA (Ecole Pour L'informatique et les Techniques Avancées). Ce projet a ainsi pour objectif premier de nous faire découvrir les coulisses de la création d'un logiciel moderne et en particulier celle d'un jeu vidéo en trois dimensions implémentant un mode multijoueur en réseau. Le développement d'un tel jeu, nous permettra :

- D'améliorer notre compréhension du langage C# (langage de programmation utilisé pour la majorité du développement du jeu sur le moteur graphique Unity 3D). Cela est d'autant plus utile que le langage C# se positionne en cinquième position des langages de programmation les plus populaires par le classement TIOBE. De plus, le langage C# permet de programmer en "orienté objet", paradigme de programmation très utilisé dans le développement d'applications modernes.
- D'apprendre à utiliser un outil de contrôle de version comme Git mais aussi une documentation comme celle d'Unity 3D mais aussi celle du langage C# ou bien d'autres bibliothèques.
- D'apprendre à travailler dans une petite équipe de quatre personnes et donc à se répartir le travail efficacement.

## **I.2 - Présentation de l'équipe**

- **Paul Credo - Codeur assidu**

J'ai débuté l'informatique en 2nd avec l'option ICN (informatique et création numérique), en commençant avec du HTML et du JS. J'ai vite pris goût à la programmation. Afin de poursuivre cette nouvelle passion je choisis la spécialité NSI en première puis en terminale.

Étant curieux et possédant un camarade aussi passionné que moi, je décide de me lancer dans de petits projets. Je commence alors par de petits sites internet, puis je me lance dans la programmation de jeu vidéo en C#. J'espère que ce projet de groupe ira aussi loin que nos espérances.

- **Charles-Antoine Leger - Chef de projet**

J'ai choisi les spécialités mathématiques et physique en terminale. L'informatique et la science sont les domaines dont je veux faire mon métier. Le processus de conception propre à l'ingénieur et aux métiers du numérique est au cœur de mes ambitions. Je code et m'entraîne en lignes (codewars) en résolvant des problèmes souvent algorithmiques, ce qui m'a conduit à faire un stage chez Berty, startup qui développe une application de messagerie sécurisée innovante. J'ai dû y étudier le protocole réseau IPFS qui lui permet d'être décentralisée (i.e. sans serveur). Cette découverte des systèmes réseau m'a plu, c'est pour cela que je suis le chef de la partie réseau de ce projet. J'aime faire partie de projet en groupe et particulièrement quand cela touche à la conception d'un jeu vidéo.

- **Clément Labbé - Directeur du gameplay**

Sorti d'une terminale maths physique j'ai toujours été très scientifique. De plus ma spécialité en première était NSI et il m'arrivait de créer des jeux connus et basiques dans le langage étudié, soit le python. Étant grand fan de jeux vidéo, il est enfin temps pour moi de pouvoir me plonger dans l'univers des créateurs qui ont tant comblé mes soirées. Je n'ai pas vraiment de connaissances en la matière. Je dirais que ma force réside dans ma créativité et dans la connaissance des jeux, ce qui va nous permettre d'ajouter des petites mécaniques pouvant à première vue sembler négligeables mais qui en réalité peuvent rendre le jeu plus amusant et intuitif, pour rendre l'expérience du joueur la plus agréable possible.

- **Pierre-Louis Pozzo di Borgo - Directeur artistique**

J'avais l'option ICN en seconde, les spécialités Mathématiques, physique et NSI en première et mathématique et physique avec l'option mathématiques expert en terminale.

J'ai toujours été passionné de jeux vidéos, y jouer et même en créer.

En effet, dès la quatrième avec mon camarade Charles-Antoine qui est lui aussi dans ce groupe, j'ai commencé à apprendre les bases du Javascript puis du C# sur Unity 3D pour créer des petits jeux.

Je suis donc familier avec Unity 3D ce qui sera utile dans ce projet, de plus je dessine beaucoup et grâce à une tablette graphique je pourrais aisément participer à la création de textures et de dessins pour le jeu.

### **I.3 - Etat de l'art, positionnement du projet**

Inspiré du nouveau film Matrix Résurrections sorti le 22 décembre 2021. Notre jeu est un jeu de type TPS dont le but est simple au premier abord : tuer les ennemis pour atteindre un objectif.

- **Rogue company**

Rogue Company est un jeu nommé TPS “third person shooter” ( où l'on voit le personnage de dos) . Le jeu propose des modes de jeu basés sur des objectifs et diverses cartes. Les matchs commencent par le parachutisme des deux équipes de l'avion au-dessus de la carte pour s'affronter dans divers objectifs. Le mode de jeu le plus jouer est l'extraction qui est un mode de jeu 4v4, où l'équipe attaquante est chargée de pirater un objectif, le vainqueur d'un match est déterminé une fois que l'objectif est piraté ou que tous les joueurs d'une équipe ont été éliminés, et Wingman est un mode de jeu 2v2.

- **Super Hot**

Le jeu utilise les mécaniques classiques du FPS, en éliminant des ennemis en se servant d'armes diverses, le cours du temps dans le jeu n'avance que lorsque le joueur effectue une action ; cela crée la possibilité pour le joueur d'analyser sa position en détail et de choisir sa prochaine action, rendant le système de jeu similaire à un jeu de stratégie en temps réel. Le jeu est présenté dans un style minimaliste, avec les ennemis en rouge pour contraster avec un décor gris et blanc.

- **Counter-Strike: Global Offensive**

Counter-Strike: Global Offensive est un jeu de type FPS, deux équipes de 5 joueurs s'affrontent (les Anti-terroristes et les terroristes). Pour gagner la partie il suffit de gagner 16 rounds sur les 30 maximum. Pour gagner un round il existe deux conditions de victoire par équipe. La condition de victoire commune est de tuer tous les joueurs du camp adverse. La deuxième diffère en fonction des camps mais reste tout aussi liée ! Les terroristes doivent faire exploser une bombe qui après avoir été posée explosera au bout de 40 secondes. Quant aux Anti-terroristes ils doivent désamorcer la bombe pendant ces 40 secondes. Au bout de 15 rounds les équipes changent de côté. Mais il existe un mode 2v2 dont le principe est le même que le 5v5 à l'exception que le jeu est beaucoup plus dynamique, car la carte de jeu est beaucoup plus petite, et que les matchs se déroulent en 9 manches gagnantes soit 16 manches dans le cas d'un égalité. On appelle ce mode “Wigman”.

Comme vous l'aurez compris, notre jeu nommé Path Maker reprend des caractéristiques de chacun des jeux cités précédemment. Nous allons reprendre la dynamique du 2v2 Wigman de CSGO pour avoir un jeu rapide ou les joueurs ne s'ennuient pas, la vue de Rogue Company nous permettant d'avoir une immersion pour de meilleure sensations de jeu, et le principe de Super Hot avec des balles très lentes et la possibilité de les accélérer. Le but est d'avoir un jeu à la fois fun et rapide permettant d'enchaîner les parties pour ne jamais s'ennuyer !

# PERIMETRE FONCTIONNEL

Cette partie tend à analyser les difficultés qui pourront être rencontrées lors de la phase de développement du jeu mais aussi à définir les limites de ce projet.

## II.1 - Analyse des difficultés

Les points de difficultés cités ci-dessous constituent un risque tant leur réussite individuelle influence la réussite globale du projet et tant ils peuvent poser des problèmes d'ordre technique.

- **Mode de jeu multijoueur**

Le mode de jeu multijoueur est un point clé de ce projet. Il est identifié comme une difficulté pour deux raisons :

- Le jeu se construisant exclusivement autour de ce mode de jeu, il constitue sa base. Dès lors, s'il se trouve mal optimisé, peu réactif et inefficace, il met en danger les performances de l'ensemble du jeu.
- Il est complexe à implémenter et demande donc un temps de développement plus important.

- **Gestion de compte liée aux joueurs**

Le mode multijoueur détaillé ci-dessus repose en partie sur la capacité de chaque joueur à posséder un compte spécifique au jeu. Cette fonctionnalité doit être intuitive et doit bien s'intégrer au reste du jeu. La difficulté derrière cette fonctionnalité réside dans la diversité des systèmes qu'elle fait intervenir. En effet, une telle fonctionnalité nécessite.

- Une base de données annexe qui s'occupe de stocker les informations de chaque joueur.



- Un serveur (backend) permettant d'authentifier le joueur à son compte mais aussi de relier la base de donnée au jeu.

- **Aspect graphique du jeu**

Au cours de nos différentes recherches à propos de la création d'un jeu en 3D, nous nous sommes rendu compte que le premier aspect par lequel un utilisateur juge un jeu est la partie graphique de celui-ci : est-il plaisant à regarder ? En effet, bien qu'il existe des jeux à l'aspect graphique simple qui arrive néanmoins à attirer de nombreux joueurs (Minecraft pour n'en citer qu'un seul), beaucoup de jeux ont du mal à attirer des joueurs simplement à cause de leur aspect purement graphique. De plus, ce sont toutes les fonctionnalités du jeu qui seront portées par une bonne cohérence graphique, et non l'inverse. Dès lors, il semble primordial de ne pas négliger un tel aspect, bien qu'il soit plus éloigné des compétences de notre équipe.

Le "graphisme" dans le domaine du jeu vidéo est un champ large. Il comprend la qualité des assets (modèles 3D, textures, animations) qui seront utilisés mais aussi l'interface du jeu (menu du jeu, interface in game) ou encore le level design, voir même l'ambiance globale du jeu. Il touche ainsi à beaucoup d'aspects du jeu, ce qui augmente sa complexité par ailleurs.

- **Intelligence artificielle**

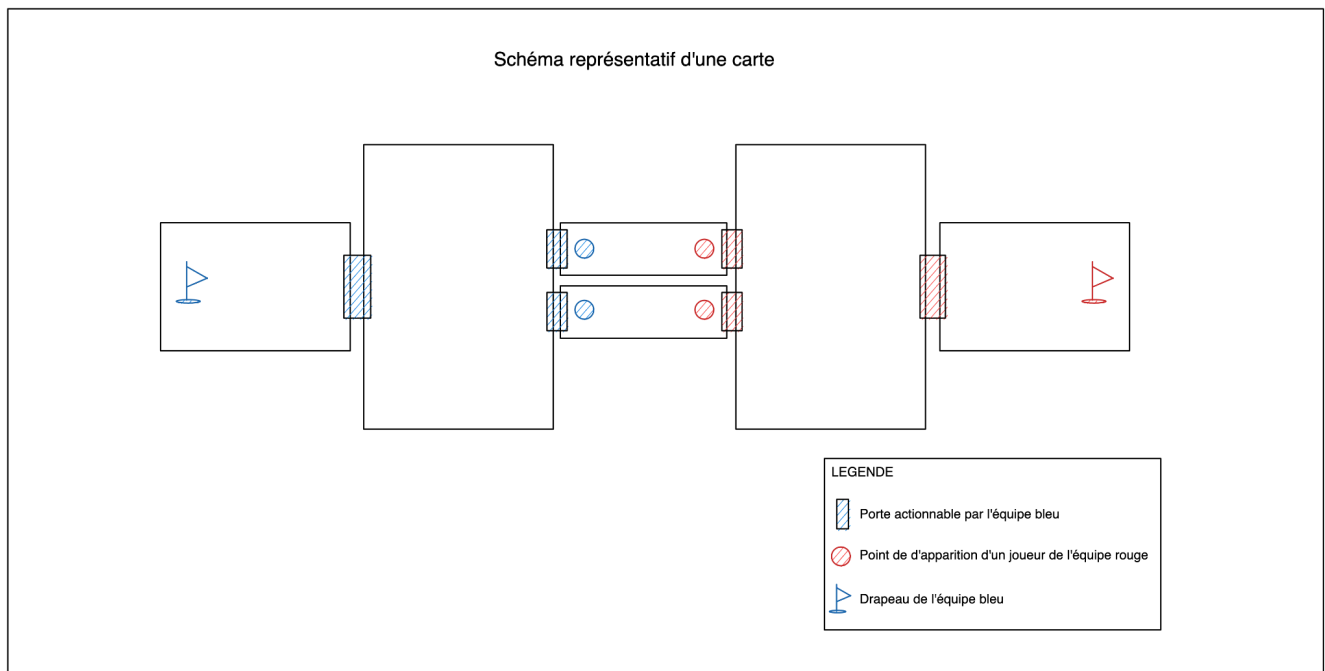
Les systèmes dit "d'intelligence artificielle" dans les jeux actuels renvoient en réalité au personnages non jouables ou bien aux ennemis. Ces intelligences artificielles sont souvent la cause de beaucoup de frustration chez les joueurs. En effet, dès qu'un personnage non joueur (PNJ) ou un ennemi comme cela sera le cas dans notre jeu ne semble plus se comporter comme un humain (surtout pour les PNJ) ou comme un autre joueur (surtout pour les ennemis contrôlés automatiquement par l'ordinateur), la totalité de l'illusion de jouer avec un environnement réels se brise et l'immersion du joueur dans le jeu est fortement impactée. C'est pour cela que cette partie du développement du jeu sera particulièrement suivie par notre équipe. Enfin, comme l'ensemble des précédents points de difficultés cités ci-dessus, le développement d'une telle intelligence artificielle est complexe.

# DESCRIPTION FONCTIONNELLE DES BESOINS

## III.1 - Spécifications fonctionnelles

### III.1.1- Gameplay

- Déplacement en trois dimension (avancer, reculer, pas de côté et saut)
- Tirer des balles ralenties (presque figées) avec un pistolet puis leur donner une vitesse normale (très rapide, presque instantanée) en appuyant sur une touche particulière.
- Chaque joueur ne dispose que d'une seule balle qu'il peut voir à travers les murs et qui peut revenir vers lui en traversant joueurs et obstacles (les balles infligent des dégâts aussi bien en étant tirées qu'en revenant vers leurs propriétaires).
- Mode de jeu: Siège.  
Il y a deux équipes de deux joueurs.  
Les joueurs doivent ouvrir des portes protégeant le drapeau de l'équipe adverse et le ramener dans leur camp de départ.  
Des points sont attribués aux équipes lorsqu'elles ouvrent la dernière porte de l'autre équipe et si elle arrive à ramener leur drapeau.  
La partie prend fin soit au bout de 8 minutes, soit lorsque l'un des drapeaux est ramené par l'une des deux équipes.  
Chaque élimination offre la possibilité d'attribuer un avantage (bonus de vitesse ou de santé) à un joueur au choix, même à un ennemi ce qui permet un meilleur équilibre de la partie. En effet, il est ennuyant de jouer pendant 8 minutes consécutives contre un adversaire bien moins talentueux que soi, une partie trop facile n'est pas amusante (donner un avantage à l'adversaire permet aussi d'obtenir à la fin de la partie un bonus de Point d'Honneur, qui est la monnaie du jeu).  
Il y a un délai de réapparition de 10 secondes après la mort d'un joueur.



### **III.1.2 - Mode de jeu multijoueur**

- Le jeu disposera d'un matchmaking (création de partie en regroupant automatiquement des joueurs connectés au jeu et entrain d'attendre de rejoindre une partie).
- La partie est hébergée sur la machine de l'un joueur qui l'aura choisi auparavant. (Cela permet en effet de simplifier le mode multijoueur mais peut aussi avoir un impact sur les performances de jeu de tous les joueurs connectés à la partie en cours).
- Si possible il y aura un système de dialogue à l'écrit pour communiquer avec son équipe et les membres de la partie.
- Un système d'ajout d'amis permettra de jouer plus facilement avec ses amis, et de rejouer avec des joueurs rencontrés dans des parties avec un matchmaking aléatoire.

### **III.1.3 - Mode de jeu un seul joueur**

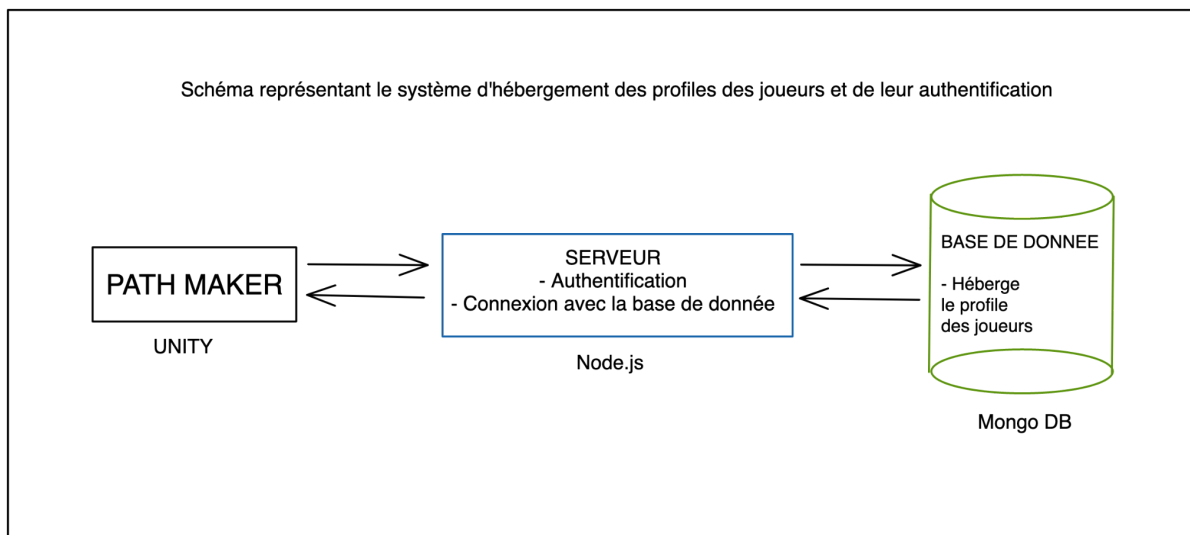
- Des IA seront programmés pour simuler des joueurs en agissant comme tel et permettre ainsi de jouer seul (déplacements, tirs...).

### III.1.4 - Magasin dans le jeu

- Différents cosmétiques seront disponibles à l'achat avec l'argent du jeu (Points d'Honneurs, gagnés en jouant ou achetés avec de l'argent réel) afin de personnaliser l'apparence du personnage et de l'arme.  
Il est en effet gratifiant de débloquer des éléments de personnalisation après avoir beaucoup joué.

### III.1.5 - Gestion de profil

- Pour jouer il faut se créer un compte, auquel on se reconnecte à chaque fois qu'on lance le jeu afin de conserver son expérience, ses amis, ses Points d'Honneur et ses cosmétiques (les cosmétiques ne procurent aucun avantage dans le jeu afin de conserver équité entre les joueurs).



### III.1.6 - Paramètres

- Il sera possible à travers un menu des paramètres dans le jeu de modifier :
  - la qualité des graphismes
  - la résolution du jeu
  - la sensibilité du mouvement de la caméra
  - le mapping des touches de contrôles
  - le volume des effets sonores du jeu
- Le menu des paramètres permet aussi d'afficher les crédits du jeu, c'est-à-dire les noms des personnes ayant contribué à son développement.

## **III.2 - Spécifications techniques**

### **III.2.1 - Technologies**

Sont ici présentés les logiciels, outils et bibliothèques qui sont utilisés pour le développement du jeu relatif à ce cahier des charges. Cette liste est non exhaustive.

#### **III.2.1.1 - Logiciels**

- **Unity**

Unity est un moteur de jeu multiplateforme développé par Unity Technologies. Ce moteur est reconnu pour sa simplicité et sa performance. En plus d'être très répandu, Unity possède une documentation riche et détaillée.

- **VS Code, Visual Studio et Rider**

Ces trois IDE sont utilisés pour le développement du jeu. (Visual Studio Code et Visual Studio sont tous les deux développés par Microsoft. Rider IDE est développé par JetBrains)

- **Git et Git Large File Storage**

Git est un logiciel de gestion de versions décentralisé. Il nous permet de travailler à plusieurs sur un seul projet unity en évitant au maximum les problèmes de conflit (lorsque plusieurs travaillent sur le même fichier par exemple).

Git Large File Storage (FLS) est une extension de Git qui permet de gérer la version de fichiers volumineux. Il sera très utile pour gérer les scènes Unity ainsi que les modèles 3D, les textures, les animations etc.

- **MongoDB Atlas Database**

MongoDB Atlas est un service d'hébergement en ligne de base de données MongoDB. Il nous permet de stocker les profils des joueurs.

- **Bootstrap**

Bootstrap est un logiciel utile à la création de design de sites internet et d'applications web. Il facilite l'ajout de formulaire, boutons, outils de navigation ainsi que des éléments interactifs. Il sera utilisé dans le cadre de la création du site internet, afin d'obtenir un design original, épuré et responsive.

### **III.II.I.II - Bibliothèques**

- **Netcode for GameObjects**

Netcode for GameObject est une bibliothèque de mise en réseau de niveau intermédiaire construite pour le moteur de jeu Unity afin d'abstraire la mise en réseau. Cela nous permet, en tant que développeur, de nous concentrer sur notre jeu plutôt que sur les protocoles de bas niveau et les cadres de mise en réseau.

Pour le choix de la bibliothèque de mise en réseau, nous avons longuement hésité. En effet, il existe beaucoup d'autres bibliothèques de ce type : Mirror, UNet, Photon etc. Notre choix réside dans le fait que Netcode soit développé par Unity Technologies. Cela nous garantit le support de la bibliothèque par les différentes versions d'Unity. De plus, Netcode possède une documentation détaillée et précise avec des tutoriels pas à pas et des projets d'exemples.

- **Node.js**

Node.js est une plateforme écrite en javascript qui permet de faire tourner du code javascript sur un serveur. Le serveur du jeu utilise Node.js pour son développement

- **Express.js**

Express.js est une bibliothèque javascript pour développer des applications web basées sur Node.js. Express.js permet de simplifier le développement de notre serveur qui assure l'authentification des joueurs à leur compte ainsi que le lien entre le jeu et notre base de données.

- **Passport.js**

Passport est un logiciel d'authentification pour Node.js. Extrêmement flexible et modulaire, Passport peut être intégré discrètement à toute application web basée sur Express. Un ensemble complet de stratégies prend en charge l'authentification par nom d'utilisateur et mot de passe, Facebook, Twitter, etc. Passport.js assure la sécurisation de l'authentification des joueurs à leur profil.

### **III.2.2 - Méthodologies**

- **Méthode agile**

La méthode agile en ingénierie logicielle met en avant la collaboration entre équipes auto-organisées. Elles s'appuient sur l'utilisation d'un cadre méthodologique léger mais suffisamment centré sur la communication. Elles préconisent une planification adaptative et un développement évolutif qui encourage les changements.

- **GitHub**

GitHub est un service web d'hébergement et de gestion de développement de logiciels utilisant le logiciel de contrôle de version Git. GitHub permet de créer des organisations regroupant plusieurs membres et ayants de fonctionnalités de gestion de projets poussées ressemblant fortement à l'outil Trello.

# Planning Prévisionnel

## IV.1 - Différentes parties du projet

Nous avons décomposé le projet en 9 grandes parties qui nous serviront de fils rouges tout au long de la conception du jeu.

- Gameplay
- UI (User Interface)
- Modélisation 3D
- Texture
- Audio
- IA
- Réseau (mode multijoueur + gestion des profils)
- Site Internet
- Marketing

## IV.2 - Répartitions des parties du projet

Afin que la conception du jeu se passe au mieux, une bonne répartition des tâches s'impose. Nous avons donc décidé de répartir les tâches de la façon suivante :

	Charles-Antoine	Clément	Paul	Pierre-Louis
Gameplay		<b>CP</b>		<b>1</b>
User Interface	<b>1</b>		<b>1</b>	<b>CP</b>
Modélisation			<b>CP</b>	<b>1</b>
Texture			<b>1</b>	<b>CP</b>
Audio		<b>CP</b>		
IA	<b>CP</b>		<b>1</b>	



Réseau	<b>CP</b>	<b>1</b>		
Site Internet	<b>1</b>		<b>CP</b>	
Marketing	<b>1</b>	<b>1</b>	<b>CP</b>	<b>1</b>

Ce tableau à double entrée se lit de la manière suivante :

- Le CP (chef de partie), est responsable de sa partie et de son avancement.
- Le CP est aidé par celui/ceux qui sont noté **1** au niveau de la tâche.

### **IV.3 - Planning**

Le projet S2 représente une grosse masse de travail, nous avons donc décidé d'un planning à respecter pour éviter tout retard lors de la réalisation des tâches.

	Soutenance 1	Soutenance 2	Soutenance 3
Gameplay	33%	66%	100%
User Interface	0%	50%	100%
Modélisation	33%	66%	100%
Texture	33%	66%	100%
Audio	0%	50%	100%
IA	0%	50%	100%
Mode multijoueur	60%	90%	100%
Gestion de compte	40%	80%	100%
Site Internet	42%	84%	100%
Marketing	0%	0%	100%

# Budget

Notre objectif premier n'est pas de générer de l'argent, et même si c'était le cas cela nous serait impossible avec la politique de l'école qui récupérerait la majeure partie des revenus générés par notre projet.

## **V.1 - Coûts du projet**

Tout projet a un coût donc voici la liste non exhaustive des coûts estimés pour la réalisation de notre projet. Nous avons fait au mieux pour réduire les coûts en utilisant des logiciels/matériels gratuits ou déjà possédé.

	Prix en €
Mise sur le marché	100
Logiciels divers	0
Goodies	40

## **V.2 - Objectifs de bénéfices**

Aucun bénéfice n'est attendu par le projet S2, ou très peu dans le cadre où la mise sur le marché serait effectuée. Dans ce cas là, nous envisageons de mettre en vente notre jeu sur la plateforme "Steam" à un prix non-définitif de 5€. Des achats intégrés seront également disponibles, avec des prix allant de 0,99€ à 4,99€.

## **VI - Conclusion**

Nous sommes un groupe de 4 jeunes passionné par l'informatique et ce que nous faisons. C'est pourquoi nous sommes extrêmement motivés par ce projet S2, nous avons mis la barre haut avec ce cahier des charges et nous comptons bien respecter tout ce qui a été écrit. Le travail d'équipe associé à nos compétences respectives va nous permettre de nous dépasser et rendre un travail des plus abouti.