

# Rapport de projet - Rogue-CAML

Enzo BERRY - Maxime DANION - Simon LECUBIN - Sacha VAUDEY

28 mai 2023

# Table des matières

<b>I</b>	<b>Reprise du cahier des charges</b>	<b>3</b>
0.1	Introduction . . . . .	4
0.2	Origine et Nature du Projet . . . . .	4
0.2.1	Les bases de l'idée du projet . . . . .	4
0.3	Origine de l'étude . . . . .	4
0.3.1	But et intérêt du projet . . . . .	4
0.3.2	Qu'est-ce que ce projet nous apporte ? . . . . .	4
0.4	État de l'art . . . . .	4
0.4.1	Quels sont les autre projet existant de ce type ? . . . . .	4
0.4.2	Quelles sont leurs fonctionnalités propres ? . . . . .	5
0.5	Le Groupe . . . . .	5
0.5.1	Enzo . . . . .	5
0.5.2	Maxime . . . . .	5
0.5.3	Simon . . . . .	5
0.5.4	Sacha . . . . .	5
0.6	Découpage du projet . . . . .	5
0.6.1	Enzo - Chef de groupe . . . . .	6
0.6.2	Maxime - Le Concepteur . . . . .	6
0.6.3	Simon - L'Expert . . . . .	6
0.6.4	Sacha - Le Designer . . . . .	6
0.7	Structure du projet . . . . .	7
0.7.1	Aspect fonctionnel . . . . .	7
0.7.2	Technologies utilisées . . . . .	8
0.7.3	Méthodologie . . . . .	8
0.7.4	Aspects Opérationnels . . . . .	8
0.7.5	Pour résumer . . . . .	8
<b>II</b>	<b>Rapport des réalisations du projet</b>	<b>9</b>
<b>1</b>	<b>Chronologie du projet</b>	<b>10</b>
1.1	Le groupe . . . . .	10
1.1.1	Conception du jeu . . . . .	10
1.1.2	Apprentissage des outils . . . . .	10
1.1.3	Répartition des tâches . . . . .	11
1.2	Rappel des soutenances . . . . .	12
1.2.1	Première soutenance . . . . .	12
1.2.2	Deuxième soutenance . . . . .	12
1.3	Les réalisations individuelles . . . . .	14
1.3.1	Enzo BERRY . . . . .	14
1.3.2	Maxime DANION . . . . .	16
1.3.3	Simon LECUIN . . . . .	17
1.3.4	Sacha VAUDEY . . . . .	19

<b>2</b>	<b>Récit de réalisation</b>	<b>22</b>
2.0.1	Point positifs . . . . .	22
2.0.2	Difficultés rencontrés . . . . .	22
2.0.3	Avis global . . . . .	23
<b>3</b>	<b>Conclusion</b>	<b>24</b>
<b>III</b>	<b>Annexes et exemples</b>	<b>25</b>
<b>4</b>	<b>Présentation générale</b>	<b>26</b>
4.1	Annexes générales . . . . .	26
4.2	Annexes de design . . . . .	26
<b>5</b>	<b>Annexes générales</b>	<b>27</b>
5.1	Discord . . . . .	27
5.2	GitHub et Git Desktop . . . . .	28
<b>6</b>	<b>Annexes de design</b>	<b>29</b>
6.1	Player . . . . .	29
6.2	Ennemis . . . . .	30

Première partie

Reprise du cahier des charges

## 0.1 Introduction

Aujourd'hui, les jeux sur mobile, PC ou encore console sont multiples. Chacun possède leur style, leur mécanique et leurs caractéristiques. C'est notamment le cas des jeux de type RogueLike, possédant une charte graphique simple mais très efficace. Les RogueLike sont des jeux de type donjons ou le joueur doit affronter des énigmes et des ennemis à travers les niveaux avant d'atteindre un niveau final, souvent représenté par un boss ou par une ultime énigme à résoudre. À la suite du niveau final, le joueur pourra gagner une récompense symbolisant la fin de la partie.

C'est dans cette idée que nous avons voulu avancer pour créer notre projet. Celui-ci reprend en effet les codes d'un RogueLike avec une mécanique de Donjons, mais avec quelques innovations lui permettant de se démarquer des autres projets de ce type déjà existant. En effet, le but de notre projet est d'implémenter dans le jeux différentes référence au monde de l'informatique et du développement de logiciel. Nous pourrons ainsi y retrouver des références à des langages de programmation ou des situations rencontrée quotidiennement par des développeurs afin d'en faire un jeu unique et ludique.

L'enjeu de ce projet sera donc à la fois de réaliser un jeu en tant que tel, mais aussi de constituer une histoire cohérente pour donner à l'utilisateur envie de jouer.

## 0.2 Origine et Nature du Projet

### 0.2.1 Les bases de l'idée du projet

L'idée de ce projet découle principalement d'une volonté commune au groupe de construire un jeu sur cette mécanique que l'on affectionnait. Ainsi, nous nous sommes basé sur le jeu Soul Knight, jeux sur mobile auquel certains d'entre nous jouaient. Selon nous, il s'agit du jeux qui représente le mieux cette mécanique de RogueLike que nous voulons implémenter dans notre projet, il était donc tout à fait naturel qu'il s'agisse de notre principale base d'inspiration.

## 0.3 Origine de l'étude

### 0.3.1 But et intérêt du projet

Ce projet va être un défi pour nous à tous les niveaux. En effet, aucun de nous n'a déjà réalisé un jeu vidéo pour PC, mais nous possédons chacun des compétences dans différents domaines utiles pour la construction de ce projet. Ainsi, celui-ci nous permettra de mettre en œuvre ces connaissances grâce au travail en équipe afin de produire le meilleur jeu possible, le plus proche de l'idée que nous en avons.

### 0.3.2 Qu'est-ce que ce projet nous apporte ?

Ce projet va tout d'abord nous aider à développer nos compétences dans le développement puisque l'intégralité de nos scripts sera en CSharp. Étant donné que ce projet nous donne une vue d'ensemble sur tout les aspect constituant un jeux à savoir la programmation fonctionnelle, mais aussi le traitement de donnée, l'intelligence artificielle, l'affichage graphique ou encore la connexion entre 2 joueurs pour qu'ils puissent jouer ensemble.

Ce projet va aussi nous apporter une meilleure coopération entre nous quatre. Travailler en équipe va en effet développer notre capacité à travailler en groupe, échanger des informations et des connaissances pour pouvoir avancer.

## 0.4 État de l'art

### 0.4.1 Quels sont les autre projet existant de ce type ?

Le style RogueLike est assez rependu dans le monde des jeux vidéo. Nous nous sommes donc inspiré sur les projets qui selon nous s'inscrivaient le plus dans ce que nous voulions faire. Ainsi, nous nous sommes principalement basés sur Soul Knight qui est le jeu qui nous a le plus inspiré pour cette idée. Dans le même thème, nous pouvons également citer le jeu The Binding of Isaac qui reprend aussi cette

mécanique de Donjons dans lequel il faut affronter des niveaux avant de combattre le boss. Enfin, le jeu Rogue Legacy est aussi un grand nom dans les Jeux de ce type et de ce fait, aussi une source d'inspiration pour nous. L'ensemble des jeux cités sont de type RogueLike, mais possède chacun une méthode différente pour avancer dans le jeu ce qui permet d'avoir un large éventail de type de jeux possédant la même mécanique.

#### 0.4.2 Quelles sont leurs fonctionnalités propres ?

Les fonctionnalités sur les jeux de type RogueLike sont souvent identiques même si quelques jeux peuvent faire exception. Le but standard de ce type de jeux est d'affronter différentes énigmes ou ennemis à travers des salles représentant les niveaux du jeu. Le but est de venir à bout des énigmes ou des ennemis d'une salle pour passer à la salle suivante. Ce système de succession de salles reprend la mécanique de Donjons présente dans ce type de jeux.

La fin du jeu est souvent représentée par l'affrontement d'un ultime boss, représenté par un ennemi à combattre ou une énigme finale à résoudre. Certains jeux peuvent posséder quelques mécaniques supplémentaires. On pourrait par exemple penser à un système d'inventaire offrant la possibilité à l'utilisateur de stocker différents items dont ils pourraient avoir besoin dans les différentes salles.

### 0.5 Le Groupe

#### 0.5.1 Enzo

18 et demi et passionné par l'informatique. Je fais également du handball en club 3 fois par semaine. Mon stage au sein du groupe SEB au sein du département Cybersécurité a confirmé ma passion pour la Cybersécurité. J'aime comprendre comment les choses fonctionnent et m'approprier leur fonctionnement. J'aime aussi le travail en équipe et par conséquent, je suis très motivé par ce projet.

#### 0.5.2 Maxime

J'ai 17 ans et demi et je suis passionné d'informatique et notamment de cybersécurité. C'est mon esprit d'analyste qui m'a amené à découvrir ce domaine tôt dans lequel je me suis directement plongé et qui m'a beaucoup apporté tout comme mon stage de 3e à Campingaz dans le service informatique. Je pratique aussi la natation de manière hebdomadaire.

#### 0.5.3 Simon

J'ai 19 ans et suis passionné d'informatique et de jeux vidéo. J'ai découvert la programmation grâce à mon père en 3ième par la création d'un petit jeu en Python. Le stage le 3ième confirma cette passion. Curieux, travailleur et impliqué, je pense que ce projet aura beaucoup à nous apporter.

#### 0.5.4 Sacha

Pour finir les présentations, j'ai 19 ans et je suis moi aussi passionné d'informatique. J'ai cette passion depuis que je suis tout petit, car j'étais, déjà à cette époque, de nature très curieuse. Cette passion s'est confirmée lors d'un stage que j'ai fait dans l'entreprise MedianeSystem où j'ai pu voir de près le métier de développeur, de chargé de test ou encore de chef de projet. Depuis, je veux faire de l'informatique mon métier et je suis convaincu que ce projet me permettra d'en découvrir plus sur ce domaine immense.

### 0.6 Découpage du projet

Cette section vise à présenter le découpage du projet en différentes tâches. Elle présente donc ce que chaque membre du groupe aura à réaliser durant les différentes étapes du projet. Cette répartition est également présente sous forme de tableau en annexe.

### 0.6.1 Enzo - Chef de groupe

Enzo prendra en charge l'organisation et la coopération générale du projet. Il veillera à ce que nous puissions développer convenablement et travailler ensemble tout en faisant que les différentes fonctionnalités développées puissent fonctionner à les unes avec les autres. De ce fait, il sera en charge de la gestion du dépôt GitHub en s'assurant de la bonne gestion des différentes branches créées. Il s'occupera également de la programmation fonctionnelle du jeu, c'est-à-dire de développer les scripts CSharp garantissant le bon fonctionnement du projet. Enfin, il apportera également son soutien sur le développement du multijoueur et de l'intelligence artificielle des personnages dans le jeu.

### 0.6.2 Maxime - Le Concepteur

Maxime assurera la création des différentes mécaniques du jeu et cogérera les designs graphiques et sonores des différents niveaux et scènes du jeu. Son but sera de réaliser les graphismes de chaque salle selon les codes du jeu roguelike définis plus haut et de rendre les salles cohérentes entre elle, tant au niveau des graphismes et que des énigmes à résoudre ou des ennemis à battre. Il apportera son soutien au développement des différents scripts du jeu afin de le rendre fonctionnel.

### 0.6.3 Simon - L'Expert

Simon prendra en charge les aspects plus théoriques et abstraits du projet avec le développement de l'intelligence artificielle et du réseau multijoueur. Ce sont des rouages essentiels au projet, car l'ensemble du jeu est construit autour de ces deux mécaniques. Son but sera donc d'assurer le fonctionnement de ces deux entités et de s'assurer également de la bonne cohabitation des nouvelles fonctionnalités qui pourront être créées et de ce système essentiel au projet. Il apportera son soutien au développement fonctionnel du jeu, mais aussi au lien des différentes fonctionnalités qui seront implémentées afin qu'elle puisse fonctionner parfaitement ensemble.

### 0.6.4 Sacha - Le Designer

Enfin, Sacha prendra en charge avec Maxime du développement graphique du jeu et s'assura du bon fonctionnement des animations des différents joueurs ou ennemis sur chaque niveau. Il prendra également en charge le développement du site web et de la rédaction des différents rapports utiles au projet. Il apportera son soutien à la programmation fonctionnelle du jeu afin de s'assurer que les fonctionnalités développées puissent fonctionner correctement et apporteront également son soutien à la gestion globale du projet.

## 0.7 Structure du projet

### 0.7.1 Aspect fonctionnel

**Schéma global** Le jeu reprendra le schéma standard des jeux de type RogueLike suivant la mécanique suivante :

- Mécanique de jeux : salles successives avec des énigmes à résoudre et/ou des ennemis à combattre. Après un nombre donné de succession de salles, l'utilisateur pourra affronter un boss final pour pouvoir terminer la partie.
- Design graphique et sonore : charte graphique et sonore reprenant les standards d'un jeu type RogueLike.
- Multi-joueurs : possibilité d'affronter les différentes salles à 2 en réseau pour aller plus vite ou résoudre des énigmes supplémentaires pour avancer plus rapidement dans le jeu.
- Intelligence artificielle : ennemis pouvant réagir automatiquement et de manière réaliste en fonction des différentes actions de l'utilisateur.
- Système de menu : système de menus permettant d'orienter l'utilisateur à travers le jeu et lui offrant la possibilité de paramétrer quelques fonctions primaires du jeu pour adapter au mieux son expérience.
- Site internet : vitrine du projet qui constituera la page de téléchargement de celui-ci, mais aussi différentes explications et informations sur le jeu.

L'ensemble de ces mécaniques vont être détaillées ci-dessous avec la méthode de fonctionnement et avec une précision sur la manière avec laquelle nous allons les implémenter.

**Mécanique de Jeu** Le jeu adoptera une mécanique de Donjons, c'est-à-dire que l'utilisateur devra traverser des niveaux successifs en résolvant des énigmes ou en affrontant des ennemis avant de pouvoir accéder au boss final. Dans notre projet, le joueur passera dans différentes salles dans lesquelles il aura l'opportunité de résoudre des énigmes. Celles-ci seront en lien avec le monde de l'informatique (références à des langages de programmation, des situation vécues par les développeurs ou encore des énigmes directement en lien avec le monde de l'informatique).

À la fin d'un nombre donné de succession de salles (nombre à définir en fonction de notre avancée sur le projet), l'utilisateur pourra combattre un boss final. Ce boss sera doté de capacité supérieure aux différents ennemis rencontrés durant le jeu. Il possédera également l'ensemble des capacités des différents ennemis que l'utilisateur aurait pu rencontrer afin d'apporter de la cohérence au jeu.

**Système de multijoueur** Le système de multi-joueur sera implémenté de telle sorte à offrir au joueur la possibilité de jouer avec quelqu'un en réseau. Cela donnera la possibilité aux deux joueurs de réaliser des quêtes supplémentaires ou de battre plus rapidement les ennemis pour avancer plus rapidement dans le jeu.

**Système d'intelligence artificielle** L'intelligence artificielle dans le jeu sera implémentée dans les différents ennemis que va rencontrer le joueur. Ceux-ci seront, en effet, capables de réagir automatiquement et avec réalisme aux différentes actions de l'utilisateur. Cela permettra d'ajouter une difficulté supplémentaire au jeu.

Ce système d'intelligence artificielle sera à terme réglable rapidement par l'utilisateur selon différents niveaux lui permettant de personnaliser son expérience au sein du jeu.

**Design graphique et sonore** Le design du jeu sera entièrement en 2D avec un effet de perspective pour s'adapter à la charte graphique des Jeux type RogueLike. Il sera donc composé de petits personnages dessinés en Pixel Art, tout comme les différents éléments de décors. Ils auront donc des animations graphiques liées avec les différents menus du jeu, mais aussi avec les différentes quêtes : animations d'ouverture et fermeture d'une boîte, interaction animée avec les ennemis et animations graphiques de certains éléments de décors. En ce qui concerne le design sonore du jeu, il sera également inspiré des jeux de ce type avec des effets audio simple. Ces animations pourront être émises lors d'interaction de la part de l'utilisateur avec les différents éléments de l'interface (menus par exemple) mais aussi lors de résolution de certaines énigmes ou encore de combat contre des ennemis. Certains éléments de décor pourront aussi émettre certains sons afin de plonger au maximum l'utilisateur dans l'ambiance du jeu.



**Site internet** Le site internet aura pour but de présenter le jeu. Il servira de page de présentation quant à l'histoire et au mode de fonctionnement général du jeu. Il pourra embarquer une page notifiant des nouvelles améliorations du jeu. Une page de téléchargement sera également disponible afin que tout le monde puisse télécharger le jeu.

### 0.7.2 Technologies utilisées

**Unity** Afin de gérer l'ensemble du design graphique, du design audio ainsi que des interaction avec l'utilisateur, la suite Unity sera utilisée afin de pouvoir avoir une grande flexibilité sur l'ensemble des fonctions qui vont être implémentées. Cette suite nous permettra aussi de pouvoir, en cas de besoins, implémenter de nouvelles fonctionnalités ou de simplement améliorer et/ou modifier celles qui existent déjà.

**Visual Studio** Afin de tirer une meilleure partie du développement du jeu, l'IDE Visual Studio sera utilisé pour le développement du jeu, car celui-ci propose une excellente intégration de CSharp ainsi qu'une excellente compatibilité avec Unity.

### 0.7.3 Méthodologie

**Communication** Afin de pouvoir communiquer rapidement sur ce que nous avons à faire, ce qui a été fait ou encore échanger sur les fonctionnalités à implémenter, nous utiliserons Discord afin que nous puissions organiser facilement nos idées, mais aussi pour être averti rapidement en cas de problème de la part d'un membre de l'équipe.

**Partage du code source** Pour que l'échange de code soit simple entre nous, nous avons fait le choix d'utiliser GitHub qui nous permettra, grâce à la création de différentes branches, d'organiser notre code selon les différentes implémentations sur lesquelles nous travaillons.

**Partage d'éléments utiles au projet** Enfin, la suite OneDrive rattachée à EPITA nous permettra de stocker des documents/images relatifs au projet afin que nous puissions y avoir accès facilement et rapidement.

### 0.7.4 Aspects Opérationnels

Afin de pouvoir réaliser au mieux ce projet, les coûts à prévoir sont l'hébergement du site internet du jeu, d'une éventuelle base de données ou encore de l'achat d'un nom de domaine. Tout ceci pourrait s'élever à environ 15 € pour une année de fonctionnement, puisque nous considérons que le projet ne générerait aucun revenu durant cette année.

### 0.7.5 Pour résumer

Cette partie était un rappel du cahier des charges. Il n'a subi aucune modification depuis celle apportée à la première soutenance sur la répartition des tâches. Nous avons fait notre maximum afin de respecter les objectifs fixés dans ce cahier des charges et avons mis en place la répartition des tâches prévues par celui-ci.

## Deuxième partie

# Rapport des réalisations du projet

# Chapitre 1

## Chronologie du projet

Pour ce rapport du projet, la partie suivante va détailler la chronologie de réalisation des différentes fonctionnalités du projet. Nous sommes en mesure de vous présenter ici un projet abouti avec des mécaniques complexes et stables, une UI permettant une navigation claire au sein du projet, un multijoueur développé et entièrement fonctionnel ainsi qu'une intelligence artificielle réaliste et cohérente avec la vision que nous avons de notre jeu. Cette partie de chronologie sera séparée en 2 sous-parties majeures, le premier détaillant nos réalisations en tant que groupe et le seconde détaillant le travail de chacun tout au long du projet.

### 1.1 Le groupe

Ce rapport de projet permet de vous détailler clairement ce qui a été réalisé par le groupe lors de l'ensemble du développement de notre projet. Le but de cette partie est d'apporter un maximum de détail sur la conception du jeu et la réalisation des différentes fonctionnalités.

#### 1.1.1 Conception du jeu

La conception du jeu a été la première étape du développement du projet en tant que groupe. Nous avons dû mettre en commun nos idées, nos envies et nos capacités afin de dresser un tableau clair de ce que nous voulions faire, mais aussi de ce qui était réalisable. La difficulté a été ici de trouver un projet qui permettait de satisfaire tous les membres du groupe et qui arrivait à lier la rapidité de développement, mais difficulté dans son utilisation afin que l'utilisateur bénéficie de la meilleure expérience de jeu. Pour ce faire, nous avons effectué diverses recherches en lien avec nos centres d'intérêt pour dresser une idée claire de ce que nous voulions. L'idée d'un rogue-like de type donjons est arrivée rapidement, car il s'agissait d'une idée qui correspondait au goût de chacun de nous. La deuxième étape de la conception a été d'écrire et de mettre au propre ce que nous voulions précisément dans ce jeu, les différentes bases d'énigmes que nous voulions implémenter et plus largement, le but de notre jeu. Cette étape nous a permis de nous mettre au clair sur ce que nous voulions faire afin que nous puissions commencer l'étape de développement le plus sereinement possible.

#### 1.1.2 Apprentissage des outils

Une fois la première étape de conception réalisée, nous avons dû nous approprier les différents outils dont nous allions avoir besoin. En effet, aucun de nous n'avait développé de jeux ; il a donc fallu se documenter sur les différents outils nécessaires à la réalisation de notre projet.

#### Unity

La première étape de notre apprentissage a été de nous approprier le logiciel Unity, point essentiel sur la conception d'un jeu. L'idée a été au début de réaliser des mini-applications pour tester les différentes fonctionnalités qui étaient à notre disposition. Nous avons ainsi pu nous approprier le concept de hiérarchie, de scène, de prefab ou encore le système de script qui allait nous permettre de réaliser la partie code de notre projet. Il s'agit ici de la plus grosse partie de l'apprentissage des outils,

car Unity est un logiciel profondément complexe qui nécessite des heures de travail avant de pouvoir être appréhendé correctement.

## **GitHub**

Comme indiqué dans le cahier des charges, nous avons fait le choix d'utiliser pour ce projet la plate-forme GitHub. Nous avons tous utiliser au moins une fois cette plateforme pour des projets personnels, mais son fonctionnement est bien différent pour un travail de groupe. Nous avons donc dû nous approprier le concept de branche et de tags afin de pouvoir travailler le plus efficacement possible au sein du groupe.

## **Visual Studio**

Enfin, nous avons du nous approprié l'IDE Visual Studio pour le développement, IDE que nous avons retenu pour sa compatibilité avec Unity et sa rapidité d'exécution. Étant donné que nous avons tous déjà développé avec Rider, la transition sur Visual Studio a été plutôt rapide.

## **Discord**

Enfin, nous avons mis en place le serveur Discord sur lequel nous avons réalisé durant tous les projet différentes réunions afin de se mettre d'accord sur le cadre précis de développement des fonctionnalités. Discord a aussi été la plate-forme où nous avons pu organiser les différentes tâches à réaliser. Le serveur nous a également permis de mettre en place un système de logs pour les dépôts du projet et du site internet afin d'être au courant lorsqu'une modification a été effectuée pour adapter notre travail en conséquence.

### **1.1.3 Répartition des tâches**

Enfin, la troisième étape a été de prévoir une répartition des tâches en fonction des capacités de chacun. La répartition des tâches que nous avons adoptées est celle indiquée dans le cahier des charges ci-dessus et rappelée en annexe. Elle a été modifiée suite à la première soutenance pour s'adapter au mieux à nos besoins à a depuis été conservée.

## 1.2 Rappel des soutenances

Cette deuxième partie vise à rappeler la chronologie de présentation des différentes étapes de notre projet. L'intérêt est ici de mettre en lumière l'évolution du projet au fil du temps, c'est-à-dire le développement des fonctionnalités et l'avancer de la conception des différentes mécaniques. Ainsi, pour cette partie, nous allons reprendre des informations énoncées dans les précédents rapports afin d'en faire une synthèse et de les mettre en parallèle de la vision du projet que nous avons aujourd'hui.

### 1.2.1 Première soutenance

Cette première soutenance nous a permis de vous présenter l'idée générale de notre projet, les différentes fonctionnalités que nous avons pu réaliser ou encore les modifications apportées au cahier des charges.

#### Cahier des charges

Lors de cette première soutenance, une modification mineure du cahier des charges vous avait été présentée. Elle concernait majoritairement l'organisation du projet et la répartition des différentes tâches à réaliser. En effet, la première étape de développement nous avait permis de nous rendre compte de quoi nous étions capables, ce que nous pouvions faire et ce que nous pouvions améliorer. De ce fait, cette nouvelle répartition des tâches présentée lors de cette soutenance nous a permis de travailler plus efficacement par la suite.

#### Fonctionnalités du projet

Pour rappel, lors de cette première soutenance, nous avons été en mesure de vous présenter un niveau entièrement fonctionnel. L'intérêt pour nous était de réaliser pour cette première soutenance un multijoueur entièrement fonctionnel (comme il le sera détaillé plus précisément dans la suite du rapport). Ce premier niveau a donc été le moyen de vous présenter cette base solide de multijoueur. Le jeu possédait déjà un fond sonore d'ambiance conservé sur la version finale que nous vous présentons. Enfin, un menu primaire étaient également disponible permettant au joueur de ne pas arriver directement sur le premier niveau.

### 1.2.2 Deuxième soutenance

Lors de cette deuxième soutenance, nous avons pu vous présenter un projet bien plus complet, avec un UI retravaillé, une intelligence artificielle mieux développée ou encore un design sonore plus abouti.

#### UI retravaillé

Lors de cette deuxième soutenance, nous avons été en mesure de vous présenter des menus bien mieux développés. En effet, le joueur peut accéder à la page "Play" pour commencer une partie ou choisir de faire quelques réglages avant de commencer. De plus, cette deuxième soutenance a vu l'apparition du menu pause permettant à l'utilisateur d'arrêter sa partie pendant un certain temps avant de la recommencer. De plus, cette deuxième soutenance a vu apparaître la distinction de room publique/privée, adapté également au niveau des menus de manière cohérente avec le reste.

#### Intelligence artificielle

Cette deuxième soutenance a surtout été l'occasion pour nous de vous présenter une intelligence artificielle nettement améliorée. En effet, les ennemis réagissaient de manière plus réaliste et cohérente avec le déplacement du joueur. Cela a permis d'augmenter le réalisme global du jeu. De plus, cette deuxième soutenance a vu l'apparition d'un nouveau type d'ennemis volant, capable de tirer à distance.

#### Mécaniques d'attaque

En plus d'une grande avancée sur la plane intelligence artificielle, la deuxième soutenance nous a permis de vous montrer un nouveau système d'attaque à distance via l'utilisation de pistolet. Cela

a permis de diversifier le Gameplay en apportant à l'utilisateur la possibilité de vaincre les ennemis d'une autre manière.

### **Interactions diverses**

Cette deuxième soutenance nous a également permis de vous présenter de nouvelles interactions entre le joueur et son environnement extérieur. Ces interactions peuvent à la fois permettre à l'utilisateur d'avancer au niveau suivant mais aussi représenter des interactions secondaires permettant d'améliorer le Gameplay général.

## 1.3 Les réalisations individuelles

Cette partie est centrale au rapport puisqu'elle détaille avec précision les réalisations de chacun pour ce projet. Contrairement à la partie précédente du rapport consacré à la réalisation de groupe, elle détaille l'ensemble des fonctionnalités et modification apportée au projet par chacun d'entre nous ainsi que notre rôle dans le développement global du projet.

### 1.3.1 Enzo BERRY

En tant que chef de groupe, Enzo a joué un rôle central dans ce projet. Il devait coordonner l'ensemble des actions que nous avions à réaliser et jouait également un rôle de gestion des différents outils que nous utilisions.

#### Mise en place des outils

Tout d'abord, Enzo a été en charge de la mise en place et de la gestion des outils. Il s'est également assuré de l'organisation structurelle du projet afin que son avancée ne soit pas ralentie ou interrompue par des divergences dans les outils. Afin de pouvoir échanger, organiser nos tâches et partager notre code, il nous fallait divers outils assurant chacun l'une de ces tâches. C'est donc pour cela que nous avons opté pour les outils décrits ci-dessus comme par exemple Discord pour les échanges et l'organisation des tâches à réaliser, GitHub pour le partage et la gestion du code source et enfin Visual Studio pour la réalisation des différents scripts afin que nous travaillions sur des outils similaires. Au cours du projet, nous avons également dû réaliser une migration de version de Unity afin de pallier des problèmes de lancement de projet. De ce fait, Enzo s'est chargé de coordonner cette migration afin qu'aucune donnée ne soit perdue et que tout le monde puisse travailler sereinement sans créer de problème inconsciemment.

#### Création d'objets globaux

Afin de faciliter la programmation globale du projet, Enzo s'est occupé de développer des objets globaux, intégrés dans chaque scène ou objet afin de faciliter leur utilisation. Ces objets pouvaient concerner des caractéristiques d'autres objets ou des méthodes redondantes dans chaque scène.

**ObjectCharacteristics** Afin de faciliter la caractérisation des objets que nous utilisions, Enzo a créé une nouvelle classe `"ObjectCharacteristics"` utilisant un encodage 32 bits permettant de décrire les caractéristiques de chacun de ces objets. Cette utilisation a permis ainsi de faciliter la compréhension de ces différents objets grâce à cette méthode de caractérisation normalisée au sein du projet.

Ce principe fonctionnait de la manière suivante. L'intégralité des objets héritait d'une classe commune nous permettant de remplacer le système de tag de Unity, complexe à utiliser, peu clair et apportant énormément de contraintes dans le développement de certaines mécaniques apportant ainsi une facilité accrue de programmation pour nous.

La création de cet objet représente pour nous un tournant dans la programmation du projet car nous avons permis de débloquer énormément de problématique rencontrée à cause du système de tag intégré à Unity.

**Prefabs globaux** Ensuite, dans cet objectif d'uniformisation, Enzo s'est chargé de la création de plusieurs prefabs permettant de simplifier la gestion de différentes ressources liées aux jeux. L'intérêt de ces prefabs n'était pas de constituer des personnages ou des objets avec lesquels interagir, mais des sortes de processus, intégrés à chaque niveau permettant d'accéder facilement à une donnée lorsque nous en avons besoin.

**Level Manager** Le premier d'entre eux est le "Level Manager". L'objectif de ce prefab est de rassembler l'ensemble des fonctionnalités qui ne diffèrent pas entre les différents niveaux du jeu. Ainsi, cet objet "LevelManager" reprend la menuette pause qui fonctionne de la même manière, peu importe le niveau du jeu, ainsi que la gestion de différentes variables globales du niveau. Ces dernières permettent d'accéder, depuis plusieurs autres objets aux informations essentielles à savoir si le jeu est en pause

ou encore combien de joueur sont connectés. Cela permet ainsi de centraliser les informations et de pouvoir y accéder rapidement pour les exploiter ou les modifier.

**Game Manager** Ce prefab possède une structure proche de celle du LevelManager, mais ça porté est plus général. En effet, alors que le LevelManager possède les informations propres au niveau dans lequel se trouve l'utilisateur, le GameManager lui va s'occuper de la gestion plus globale du fonctionnement du jeu. On peut y retrouver par exemple la gestion de l'audio d'ambiance des différents niveaux, mais aussi les différentes méthodes permettant d'initialiser le multijoueur. Pour ces dernières, le GameManager ne s'occupe pas de la création même du multijoueur, mais simplement de rassembler les informations nécessaires à la création de celui-ci.

### Support à d'autres tâches

Dans cette dernière phase de développement, Enzo a également apporté son soutien au développement d'autres fonctionnalités prise en charge par les autres membres du groupe.

Il a d'ailleurs pu venir en aide à Simon sur la réalisation du système d'attaque permettant de peaufiner certains détails et de gérer de manière plus efficace la synchronisation en réseau des différents éléments. Ce travail avec Simon s'est aussi porté sur le fait de rendre plus réaliste ce système d'attaque et de corriger les éventuels bugs qu'il a pu entraîner sur le reste du jeu et les interactions que peut avoir le joueur avec son environnement.

Enzo s'est également occupé de l'amélioration du comportement du joueur notamment dans le cas où celui-ci perd l'intégralité de sa vie. Ce joueur peut donc apparaître mort en multijoueur grâce à un effet de transparence appliqué sur celui-ci comme indiqué dans la figure suivante :

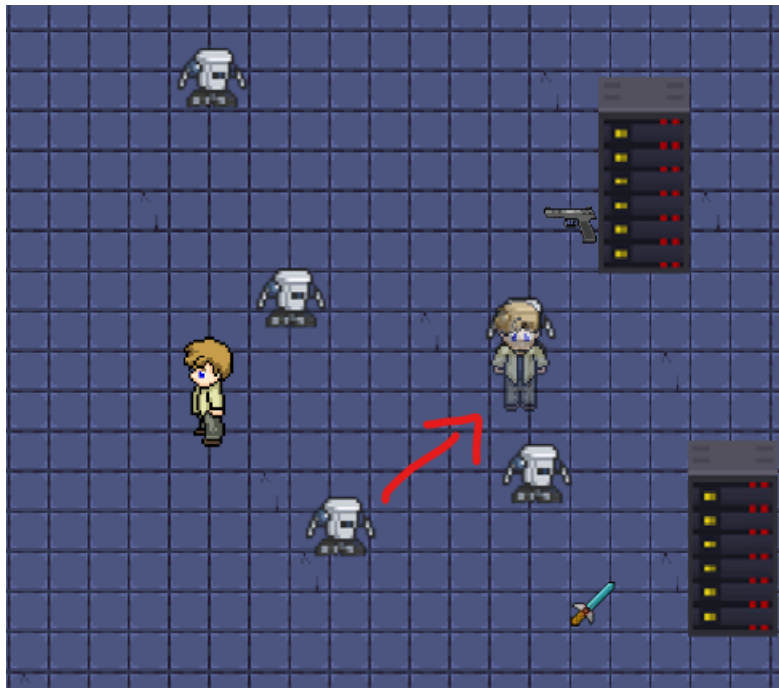


FIGURE 1.1 – Effet appliqué au joueur en cas de mort

Il est aussi venu en aide à Sacha sur la conception du menu principal, qui a été retravaillé ensuite.(cf.1.3.4)

### Pour résumer

En somme, Enzo a eu sur l'ensemble du projet un rôle de coordination du groupe avec la gestion des outils et des différentes tâches que nous avons à réaliser, mais il a également eu un rôle de gestion globale du projet en s'assurant que les scripts et fonctionnalités que nous réalisions étaient en accord avec les principes de développement que nous avons mises en place et suffisamment cimentée afin



qu'elles soient compréhensibles par tous. Enfin, Enzo a pu apporter son aide aux autres membres du groupe sur le développement de fonctionnalités diverses.

### 1.3.2 Maxime DANION

Comme les autres membres du groupe, Maxime avait un rôle et des responsabilités spécifiques. Il a dû réfléchir à la conception des différentes mécaniques à mettre en place ainsi qu'au level design afin d'obtenir le Gameplay le plus intéressant possible pour le joueur. Ainsi, les parties suivantes vous détailleront avec précision l'ensemble de ses réalisations au cours du projet.

#### Réalisation des menus

Tout d'abord, Maxime s'est attelé à la réalisation du menu pause, élément essentiel du jeu. En effet, ce menu permet à l'utilisateur d'arrêter son jeu en cas de besoins, mais aussi d'adapter le réglage du volume qui a été ajouté à ce menu pour cette dernière phase de développement. En équipe avec Sacha, ils ont tâché à garder une certaine cohérence dans les menus au niveau des couleurs, police d'écritures, etc. La charte graphique globale du projet à évoluer au cours de celui-ci pour migrer vers quelque chose de plus simple, reprenant les codes couleur du projet et le personnage principal. Ainsi, le menu principal s'est grandement simplifié possédant maintenant le strict minimum en termes d'informations afin que le joueur puisse s'y retrouver facilement. Ainsi, Maxime et Sacha ont pu travailler en équipe afin d'assurer cette cohérence entre les différents menus, respectant ainsi cette charte graphique mise en place.

#### Support à la réalisation des graphismes

Maxime a également aidé Sacha dans la réalisation des graphismes du premier et du deuxième niveau, l'idée étant d'obtenir un design cohérent et en rapport avec le thème de notre jeu à savoir le Rogue-Like et le monde de l'informatique. Ainsi, lors de la conception de ces deux niveaux, il a fallu obtenir différents éléments de décor permettant de rappeler ce monde de l'informatique, mais aussi trouver le moyen de casser la redondance qu'il pouvait y avoir dans les niveaux en désignant des salles avec une architecture complexes ou encore en ajoutant des éléments de décors permettant de casser la monotonie du niveau. Ce travail a été fastidieux, car il a fallu trouver le bon équilibre entre esthétique et encombrement du niveau afin de permettre à l'utilisateur de profiter de la meilleure expérience de Gameplay possible tout en rendant l'architecture du niveau complexe.

#### Design du troisième niveau

Maxime s'est également occupé du design général du niveau 3 cette fois en y apportant une mécanique de fonctionnement entièrement nouvelle. L'idée est ainsi d'obtenir un niveau entièrement différent des précédents menant le joueur dans de nouvelles mécaniques. En effet, Maxime a fait le choix pour ce 3e niveau de le concevoir de telle sorte qu'il soit constitué de petites salles dans lesquelles le joueur se déplace rencontrant monstres et mécaniques diverses.

De ce fait, ce troisième niveau repose sur le fait que l'utilisateur doit appliquer ce qu'il a appris au niveau précédent dans ce troisième niveau afin de pouvoir remporter la partie. Ce troisième niveau doit ainsi être en cohérence avec les précédents, c'est-à-dire contenir des mécaniques supplémentaires et plus complexes afin de s'inscrire dans cette volonté de croissance de difficulté au cours des niveaux. Le boss final présent en fin de niveau permet d'ajouter un but au jeu et de déterminer si l'utilisateur gagne la partie ou non.

Si l'utilisateur arrive à finir de niveau 3, selon la difficulté sélectionnée, c'est qu'il a pu arriver à bout de l'ensemble des mécaniques que nous avons imaginées et peut donc remporter la partie.

#### Support au développement des mécaniques

Enfin, Maxime a apporté son soutien au développement des diverses mécaniques secondaires du jeu. Ces mécaniques, bien que moins complexes que le système d'attaque, possèdent toute de même une importance capitale. En effet, ces mécaniques secondaires sont celles permettant aux joueurs d'être en immersion dans le jeu. Ces mécaniques peuvent prendre la forme de messages donnés pour l'aider à avancer, la possibilité de trouver des objets cachés ou encore de pouvoir interagir avec les différents

objets de la pièce. Ces mécaniques représentent séparément peu de travail, mais il faut en implémenter un nombre conséquent pour trouver de la cohérence dans le jeu. Maxime a donc élaborer certaines mécaniques et les a implémentées comme les transitions améliorées entre les niveaux qui permettent à l'utilisateur d'être plongé dans l'histoire du jeu.

### Pour résumer

En somme, Maxime s'est principalement occupé du développement graphiques du projet avec l'ajout de nouveaux niveaux et le design des menus, mais s'est aussi occupé de la conception et du développement des mécaniques du jeu afin d'offrir une expérience complète à l'utilisateur.

### 1.3.3 Simon LECUIN

Simon s'est attardé principalement sur le système d'attaques présentes tout au long du jeu ainsi que le développement du multijoueur. Ce travail long et fastidieux était un vecteur essentiel de notre projet puisque le jeu repose sur l'attaque entre joueurs et ennemis et tout ceci en réseau.

### Réalisation du multijoueur

La première tâche à réaliser que Simon a pris en charge était l'élaboration du multijoueur. En effet, l'intégralité de notre jeu repose sur la synchronisation en réseau des différents objets et utilisateur qui allait y figurer.

**Choix d'implémentation** Pour faciliter les choses, plutôt que d'implémenter deux versions du jeu, une solitaire et une multijoueur, nous avons préféré développer uniquement le multijoueur, mais l'utilisateur peut jouer en solo dans une salle multijoueur privé. L'intérêt pour nous d'utiliser cette technique et de traiter d'une manière unique l'ensemble des composants de notre jeu. Simon s'est donc attelé à cette tâche. Il s'est d'abord approprié la documentation du Photon, service retenue pour le multijoueur de notre projet, afin de pouvoir développer les différentes composantes dont nous avions besoin.

**Réalisation des différentes composantes** Une fois l'étape de développement pure du multijoueur, c'est-à-dire le processus de création de salle et la base de l'architecture réseau, Simon s'est occupé de la réalisation des différentes composantes du multijoueur. Cela signifie donc qu'il est occupé de la réalisation de l'interface primaire permettant la création de salle multijoueur, interface retravaillée ensuite par Maxime et Sacha. Cette interface devait permettre à chaque utilisateur de pouvoir s'identifier via le choix d'un pseudonyme, mais aussi de rejoindre soit une salle publique, soit une salle privé protégée par un code. La deuxième composante à mettre en place a été la réalisation des différents systèmes permettant la synchronisation en réseau des événements du jeu. Nous pouvons citer l'action "Pickup/Drop" qui a dû être gérée au niveau réseau afin que les armes possèdent le propriétaire adéquat, mais aussi la vie de l'utilisateur qui doit être synchronisée entre tous afin que le joueur soit averti de sa propre vie, mais aussi de celle de ses compagnons.

**Pour résumer** Cette étape a peut-être été la plus longue du projet puisqu'elle représente la base de notre jeu. Il a donc fallu s'assurer qu'elle était stable et capable de supporter les différentes fonctionnalités et mécanique que nous voulions mettre en place.

### Réalisation du système d'attaque

Simon s'est également occupé du développement d'une autre grosse brique de notre projet à savoir le système d'attaque. Ce système d'attaque représente un pilier majeur de notre projet, car il est la mécanique principale du premier niveau ainsi que du combat contre le boss final. Le plus complexe dans la création de cette mécanique d'attaque a été d'ajouter un comportement intelligence aux ennemis de l'utilisateur. En effet, il aurait été incohérent que les ennemis se déplacent de manière linéaire et de tir de façon mécanique et répétitive. Ainsi, Simon a dû élaborer un système d'ennemis qui reste à distance par rapport à l'utilisateur afin de générer une difficulté supplémentaire. De plus, il a élaboré un système de difficulté paramétrable permettant de modifier le comportement des ennemis et de leur

manière d'agir. Cela peut se ressentir sur la vitesse de tir et de déplacement ou encore sur la précision de tir des ennemis. À titre d'exemple, le boss final possède elle moyen de tirer en direction de l'ennemi et le projectile suit une trajectoire réfléchi pour avoir plus de chance de toucher le joueur. L'ensemble de ces mécaniques d'attaque ont donc demandé une réflexion poussée quant à leur fonctionnement afin qu'elle soit cohérente et le plus réaliste possible, en adéquation bien sûr avec le thème du jeu.



FIGURE 1.2 – Animation de l'attaque de l'épée - Level 1

### Développement des mécaniques

Enfin, Maxime s'est occupée de la conception et du développement de nouvelles mécaniques afin que le gameplay soit plus intéressant. Nous avons donc implémenté une mécanique de "pylon" où le joueur doit déplacer au boulet jusqu'à un socle afin de valider la mécanique. Cet objet, simple en l'apparence permet de débloquent beaucoup d'horizons pour notre jeu, certains niveaux devront demander de la réflexion de la part du joueur, car ces deux mécaniques nous isolent d'un gameplay uniquement linéaire où il faut battre des monstres. En effet, elle ajoute une interaction supplémentaire entre le Player et son environnement. Ainsi, cette politique de développement mécanique rejoint celle du développement de l'architecture des niveaux avec la volonté d'une difficulté croissante au fil du jeu.



FIGURE 1.3 – Mécanique du Pylon du Level 2

## Pour résumer

En somme, Simon s'est principalement occupé des fonctions techniques sur le jeu à savoir la gestion des synchronisations réseau des différents éléments ainsi que le développement des systèmes d'attaque qui ont des mécaniques de fonctionnement très complexes.

### 1.3.4 Sacha VAUDEY

Enfin, Sacha s'est principalement occupé du design globale du projet, que ce soit sur le jeu directement, mais aussi sur la réalisation des différents éléments graphique du site internet ou de la boîte dans laquelle le jeu vous a été rendu. De plus, il s'est occupé de la conception des mécaniques avec Maxime afin d'enrichir le jeu de diverses fonctionnalités supplémentaires.

## Réalisation des graphismes

La réalisation des différents graphismes concernent en réalité deux parties majeures à savoir la recherche des différents graphismes mais aussi l'implémentation de ceux-ci avec leur animations.

**Recherche** Comme il a déjà été indiqué dans le rapport de la première soutenance, cette étape a été fastidieuse, car il fallut trouver des ressources graphiques qui correspondaient à ce que nous recherchions. Cette étape a également été longue, car aucun de nous n'avait de véritables talents de conception graphique et la difficulté de créer nos propres sprite et composants de jeux était donc une idée assez complexe pour nous. Malgré cette longue heure de recherche, nous avons pu rassembler différentes ressources dont nous en avons pu modifier certains détails afin qu'elle colle parfaitement à ce que ne voulions pas. Le travail de Sacha a donc été sur la partie graphique à la fois de recherche du contenu de type pixel art pour notre jeu, mais aussi de modifier les ressources trouvées afin qu'elles aient une cohérence entre elles. L'objectif que nous avions ici était de ne pas perdre trop de temps à concevoir l'intégralité des ressources graphiques afin que nous puissions nous concentrer sur le développement même du jeu et sur ses différentes fonctionnalités.

**Implémentation** Une fois que l'ensemble des ressources ont été rassemblées et qu'elles ont été modifiées aux besoins, Sacha s'est occupée de les organiser correctement dans le projet afin d'en faire des objets préfabriqués nommés "prefab". Ces prefabs nous permettent de pouvoir gérer facilement nos différentes ressources afin que les diverses modifications apportées soient synchronisées dans l'ensemble du projet. Ces différents prefabs représentent à la fois les sprite (joueurs et ennemis), des éléments de décors avec lequel le joueur peut interagir ou encore les armes qui peuvent équiper le joueur. Certains de ces prefabs avaient des mécaniques communes. Par exemple, le joueur et les ennemis étant des personnages qui se déplacent, ils contiennent tous deux un RigidBody permettant de simuler de la physique entre eux (notamment en cas de collision). De plus, certains composants comme les éléments de décor ont eu besoin de chacun d'une attention particulière afin de paramétrer leur collision. Ce collision représente une zone dans laquelle les éléments rentreront en collision avec cet objet. Certains éléments de décors comme les murs possèdent des zones de collision infranchissables, ce qui n'est pas le cas de certains objets de décors où le joueur peut passer devant ou derrière. Le travail de Sacha a donc été ça, la configuration primaire des éléments graphiques afin qu'il puisse être exploitable lors de la programmation de leur mécanique.

## Support à la réalisation des menus

L'une des tâches de Sacha a également été la gestion des différents menus et de leur cohérence graphique par rapport à la charte mise en place pour notre jeu. Ainsi, Sacha s'est occupé de re-designer ces menus en fonction des éléments existants afin de l'unifier avec le reste du projet.

Parmi ces tâches, il y a l'ajout d'un UI permettant de donner des informations supplémentaires au joueur lors de sa partie.

## Réalisation du design Audio

L'audio étant une partie nécessaire d'un jeu, Sacha s'est penché à ce que celui-ci soit de qualité sur notre projet. L'intérêt était donc d'avoir des musiques d'ambiance accompagnant l'utilisateur

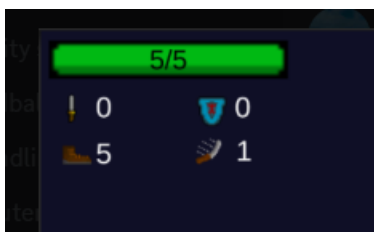


FIGURE 1.4 – UI du joueur

sur les différents niveaux (une musique différente par niveau), ainsi qu'une musique principale reprise sur le menu principal du jeux constituant la bande originale du projet. Le design Audio a aussi été de développer des effets sonores sur différents éléments du jeux lorsqu'ils interagissent entre eux ou avec l'utilisateur. Pour cela, il a fallu la encore trouver des ressources cohérentes avec le thème du jeu. Pour cela, il a fallu la encore trouver des ressources cohérentes avec le thème du jeu. De plus, les jeux ayant pour style graphique utilisent souvent ce genre de sons (par exemple les anciennes versions de Pokémon). Ainsi, il devient logique que nous devons utiliser ce style d'audio pour notre jeu. La difficulté a ici été de trouver des musiques qui nous plaisaient. Nous avons, après diverses recherches, finalement trouver ce que nous voulions de le style que nous recherchions. Sacha s'est ensuite occupé de traiter ces musiques afin que leur niveau soit cohérent les unes par rapport aux autres. En effet, étant donné que le volume sonore est géré de manière commune dans le jeu, il est primordial que le son de chaque musique soit à un niveau identique pour le réglage du son, dans le jeu par l'utilisateur, soit plus facile.

Sacha s'est enfin occupé, en ce qui concerne le son, au traitement des différentes ressources sonore utilisé pour les interactions dans le jeu. Ces ressources ont du également être traitées afin que leur niveau sonore soit cohérent par rapport à l'environnement dans lequel elles sont placées. De ce fait, Sacha à du la aussi effectuer un traité sur ses fichiers audio avant de les intégrer au projet.

### Réalisation des documents et site internet

Enfin, Sacha s'est occupé de la réalisation des différentes ressources administratives annexe au projet. Cela représente à la fois la rédaction des différents rapports, même si l'ensemble du groupe y a apporté des modifications, ainsi que la création et le développement continu du site internet.

Pour commencer avec les différents rapports, il était évident que, comme stipulé dans les consignes, chaque rendu de soutenance soit accompagné d'un rapport. Ce rapport a pour but de spécifié l'ensemble des avancées du projet depuis le dépôt du rapport précédent. Nous avons fait le choix de concentrer une unique personne sur ces tâches afin que la rédaction globale du rapport soit cohérente. En effet, laisser chaque membre du groupe rédiger sa partie aurait créé des inégalités sur le rapport, tant au niveau de la forme qu'au niveau de la manière de formuler le fond. Ainsi, le choix a été fait de laisser Sacha sur cette tâche. Bien que les différents rapports aient été rédigés par une personne unique, l'ensemble des membres du projet y ont ajouté leur touche personnelle via la reformulation de certaines parties, mais aussi par la relecture globale des documents. En plus de la rédaction des rapports, Sacha a effectué les différents manuels (d'installation/désinstallation et utilisation) qui vous ont été fournis en même temps que ce rapport.

Dans notre répartition des tâches déposées lors du cahier des charges, nous avons prévus que cette tâche administrative comprendrait également la réalisation du site internet et c'est donc pour cela que cette tâche rentre aussi dans les réalisations de Sacha. Le site internet est un élément essentiel du projet puisqu'il est la base de celui-ci, mais aussi sa vitrine. En cas de publication du projet, on pourrait penser que le site internet constituant le moyen pour les personnes extérieures de connaître le projet et ainsi de l'installer. Il va donc de soit que ce site internet doit être ergonomique et puisse donner envie à l'utilisateur de télécharger ce projet. Pour cette tâche, nous avons fait le choix d'utiliser un outil de no-code nommé "NicePage" pour développer notre site internet. En effet, nous avons fait le choix de concentrer un maximum de notre énergie dans le développement du projet. De ce fait, nous avons dû organiser nos tâches et les outils dont nous allions avoir besoin en conséquence afin de libérer un maximum de temps au développement des différentes fonctionnalités. La encore, afin d'optimiser un maximum de temps, l'intégralité du site internet a été réalisé pour la première soutenance. Ainsi,

depuis cette soutenance, le site internet n'a subi que des améliorations mineures de contenu ou de corrections, mais aucune mise à jour importante de structure n'a été réalisé. En effet, pour la première soutenance, nous nous sommes concentrés à vous présenter un premier niveau avec un multijoueur fonctionnel. Étant donné que la partie réseau était géré par Simon et Enzo en majorité, cela a laissé du temps à Sacha pour développer le site internet. Ainsi, lors des futures étapes de développement, le site internet n'a pas été dans nos préoccupations puisque son développement était déjà bien avancé.

**Pour résumer**

En somme, Sacha s'est majoritairement occupé durant tout le projet sur le design, qu'il soit graphique ou sonore ainsi que la réalisation des différents documents administratifs. Il a aussi apporté son soutien sur la conception et le développement des différentes mécaniques implémentées dans notre projet.

# Chapitre 2

## Récit de réalisation

Ce chapitre vise à détailler plus précisément le cadre et l’ambiance de réalisation de ce projet. Nous vous détaillerons les difficultés que nous avons pu rencontrer, mais également les points positifs de notre projet. Nous pourrions terminer par un avis global de l’ensemble de la réalisation de notre jeu.

### 2.0.1 Point positifs

Beaucoup de points positifs sont à retenir de ce projet qui était une première pour chacun d’entre nous. En général, il nous a permis d’exprimer notre créativité, de pouvoir apprendre de nouvelles notions de programmation ou encore d’apprendre les différents outils de travail collaboratif.

#### Le groupe

Pour commencer par l’ensemble du groupe, nous avons su garder tout au long du projet une grande cohésion ce qui nous a permis de pouvoir débattre sereinement des fonctionnalités à implémenter dans le jeu ou encore d’avoir une certaine efficacité dans notre travail. Le fait que certaines d’entre nous se connaissaient avant EPITA a sûrement permis d’améliorer encore une fois cette cohésion puisque nous connaissions déjà la méthode de travail de chacun. En somme, ce groupe nous a permis d’arriver à notre objectif à savoir vous présenter un jeu Rogue-CAML que nous avons imaginé.

#### Le développement CSharp

Nous avons tout déjà eu une expérience en programmation personnelle. En effet, nous avons tous déjà développé de petits projets, minimes certes, mais qui nous ont permis de connaître déjà quelques bases sur des outils que nous avons utilisés. C’est le cas de GitHub ou déjà plusieurs d’entre nous avaient utilisé cet outil et connaissaient donc ses spécificités (branche, tag, commit, etc.). Notre expérience depuis le début de l’année à EPITA nous a aussi permis d’accroître nos connaissances dans ce domaine.

### 2.0.2 Difficultés rencontrés

Bien que l’aspect global de ce projet semble positif, nous avons tout de même rencontrés certaines difficultés lesquelles sont listées ci-dessous. Ces difficultés n’ont pas été de véritables points bloquants pour notre projet, mais plutôt des zones où nous avons pris plus de temps ce qui a pu nous retarder sur notre planning.

#### Rassemblement des ressources

La première difficulté à laquelle nous avons été confrontées, et ce, dès la conception du jeu a été le rassemblement des ressources. En effet, il existe beaucoup de ressources disponibles sur Internet avec le style de Pixel Art, mais celle-ci ne correspondait pas à ce que nous cherchions. Bien souvent, ces ressources graphiques possédaient beaucoup de détail ou n’étaient pas en lien avec le monde informatique. Or, pour notre projet, nous avons fait le choix de design simple ce qui ne nous permettait pas d’utiliser ces ressources. Le travail de chercher a donc été conséquent et long pour trouver tout ce dont

nous avons besoin. Malgré cela, nous avons quand même dû réaliser un bon nombre de correction sur ces ressources afin qu'elles s'adaptent parfaitement à ce que nous voulions.

### **Documentation des scripts**

La deuxième difficulté que nous avons rencontrée a été la bonne documentation des scripts réalisés par chacun d'entre nous. En effet, nous avons tous une manière différente de coder et n'avons pas forcément l'habitude de documenter notre code, du moins, pas de manière assez précise. Ainsi, il a été difficile pour nous au départ de reprendre des codes que nous n'avions pas forcément réalisés. Il fallait en effet se rapprocher la manière de coder avant de comprendre l'utilité du script. Nous avons bien évidemment rapidement corrigé ce problème en adoptant les standards de documentations de nos différents scripts afin qu'ils soient lisibles et compréhensibles par tous.

### **Synchronisations réseaux**

Nous avons également rencontré une difficulté majeure quant à la synchronisation de nos différents éléments en réseaux. En effet, lors de la première soutenance, nous vous avons présenté un jeu avec un multijoueur, déjà fonctionnel. Cependant, ce multijoueur avait été développé afin qu'il soit fonctionnel en l'état, mais l'ajout de nouvelles fonctionnalités et de niveaux n'était pas possible avec ce système. Cela s'est surtout fait ressentir avec les armes. Nous avons ainsi rencontré des difficultés quant à la synchronisation de divers éléments entre le multijoueur. Cela s'est surtout fait ressentir avec les armes. En effet, comme chaque joueur peut récupérer des armes au sol, il faut que le propriétaire de l'arme soit correctement synchronisé en réseaux afin que les autres joueurs ne puissent pas la contrôler. Cette étape a été une partie majeure du développement de notre projet, car nous avons dû nous assurer que tout fonctionne correctement dans toutes les situations possibles.

### **2.0.3 Avis global**

En somme, nous sommes plutôt satisfaits de l'ensemble de notre projet. En effet, nous avons su garder notre cohésion tout au long de celui-ci nous permettant de réaliser un jeu à hauteur de ce que nous avons imaginé. Nous avons certes rencontré certaines difficultés, mais elles nous ont permis d'en apprendre davantage sur la manière de programmer et de développer un projet en groupe. Indéniablement, ce projet a été l'occasion pour nous de faire d'énormes progrès en programmation, de pouvoir découvrir les ressources qui étaient à notre disposition ou encore de pouvoir progresser sur les différents outils de travail collaboratif. Ce projet a donc été pour nous quelque chose d'enrichissant qui nous a permis de faire place à notre créativité et à notre passion pour la programmation.



## Chapitre 3

# Conclusion

En somme, nous avons pu exprimer par ce projet notre créativité et notre volonté de développer un jeu pour PC. Bien que sa réalisation n'a pas été toujours facile, nous avons donné le maximum pour que le projet que nous vous rendons soit le plus abouti possible. Nous vous avons détaillé dans ce rapport l'ensemble des réalisations que nous avons menées pour ce projet, nos sources d'inspiration et le récit de sa réalisation.

Troisième partie

Annexes et exemples

## Chapitre 4

# Présentation générale

Les annexes de ce rapport ont pour but d'enrichir les propos que nous avons eus par des exemples précis, représentant conquêtes et simplement le fonctionnement des différentes fonctionnalités et mécaniques de notre jeu. Ces annexes seront réparties en différentes catégories afin qu'elles apportent le maximum de détail.

### 4.1 Annexes générales

Ces annexes porteront sur du aspect général du projet. Elle comprendra des illustrations de nos outils de travail ou encore de l'architecture globale du projet.

### 4.2 Annexes de design

Il s'agira donc d'exemple de Sprite utilisé dans notre jeu ou du travail réalisée sur certaines ressources graphiques que nous avons trouvés afin qu'elle s'adapte à leur contexte d'utilisation dans le jeu. Il s'agira donc d'exemple de Sprite utilisé dans notre jeu ou du travail réalisée sur certaines ressources graphiques que nous avons trouvés afin qu'elle s'adapte à leur contexte d'utilisation dans le jeu.

# Chapitre 5

## Annexes générales

### 5.1 Discord

Cette annexe vise à présenter l'organisation que nous avons adoptée sur Discord pour la réalisation du projet. Le serveur se décompose en plusieurs salons ayant leur utilité propre.

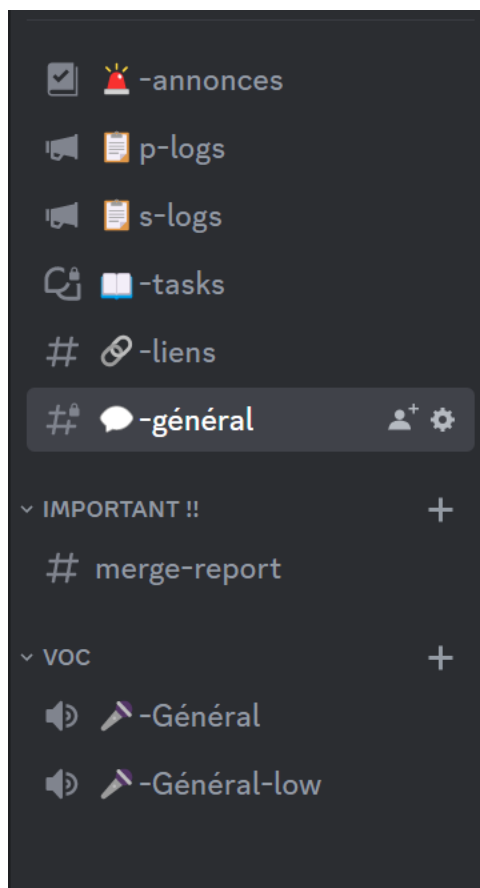


FIGURE 5.1 – Architecture du serveur Discord

**p-logs** Ce salon nous à permis durant tout le projet d'obtenir les logs du dépôt GitHub du projet afin d'être au courant de toute fonctionnalité

**s-logs** Ce salon possède la même utilité de le précédent, mais concerne cette fois les logs du dépôt du site internet

**tasks** Salon de type "Forum" permettant d'organiser nos différentes tâches selon des tags (urgent, UI, IA, mécaniques, etc.) et d'en discuter ensuite.

**Autres salon** Les autres salons nous permettaient de pouvoir communiquer les informations importantes (date de rendu, soutenance, etc.) ainsi que de discuter de la mise en place général du projet.

## 5.2 GitHub et Git Desktop

Structure générale des branches du GitHub à différents intervalles. Notre volonté ici a été de garder un minimum de branche ouverte et merge dès que possible dans "dev" pour avoir un code clair et réduire le risque de conflits. Cette annexe comprend également le Git Desktop, logiciel retenu par notre groupe pour la gestion du dépôt. Ce dépôt est disponible publiquement sur le lien suivant : <https://github.com/enzo-berry/Rogue-Caml>

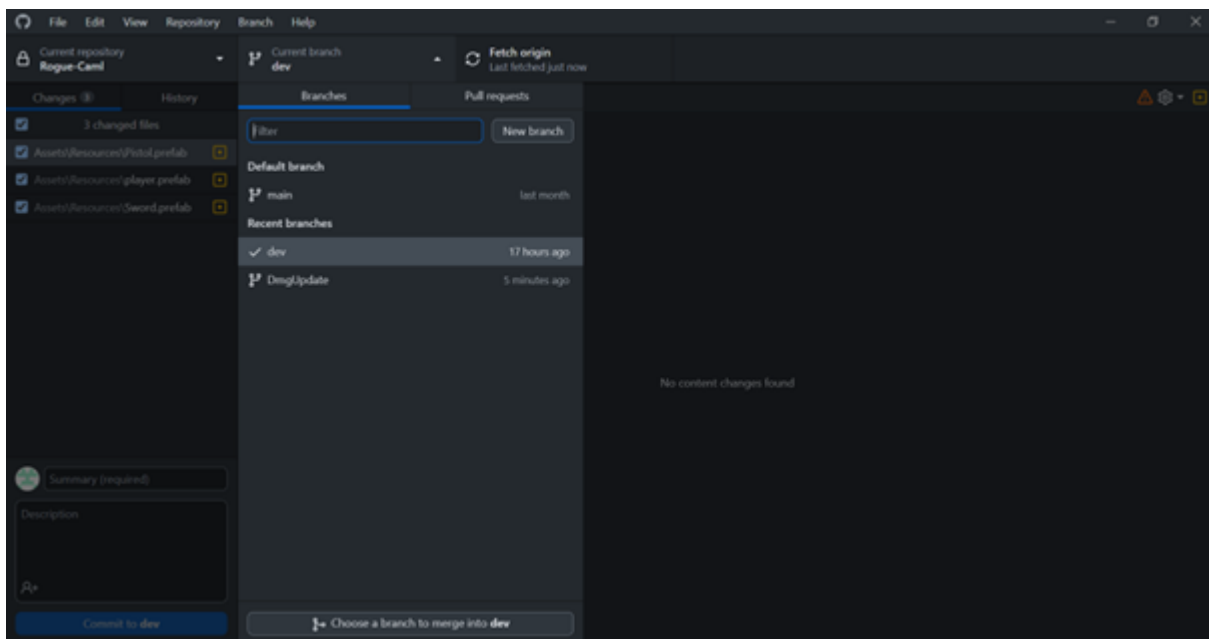


FIGURE 5.2 – Architecture du depot et utilisation du logiciel Git Desktop

## Chapitre 6

# Annexes de design

Cette annexe comprend les différents designs que nous avons pu utiliser et des exemples de modification apportés sur ceux-ci afin qu'il s'intègre parfaitement à notre jeu.

### 6.1 Player

Design du Player utilisé tout au long du jeu.

**Joueur en vie** Ce design de Player a été utilisé sans modifications majeures de notre part. Seule des modifications en jeux au moment de la mort du personnage ont du être réalisée.

**Joueur mort** Ce design a été créé par rapport à la position par défaut du personnage et a été légèrement modifié pour apparaître comme mort dans le jeu. Ceci repose sur un effet de transparence appliqué au joueur directement dans Unity.



FIGURE 6.1 – Design du joueur principal

## 6.2 Ennemis

Design des ennemis utilisés dans le jeu. Aucune modification particulière n'a été apportée sur ces sprite, seules quelques corrections colorimétrique ont été apportés sur certaines d'entre eux afin d'unifier certaines de leurs surfaces.

**Petits robots** Il s'agit du sprite de petits robots présents dans le premier niveau.



FIGURE 6.2 – Petit robot - Level 1 uniquement

**Gros robots** Également présent dans le premier niveau, il s'agit d'ennemis plus grand capable de tirer à distance

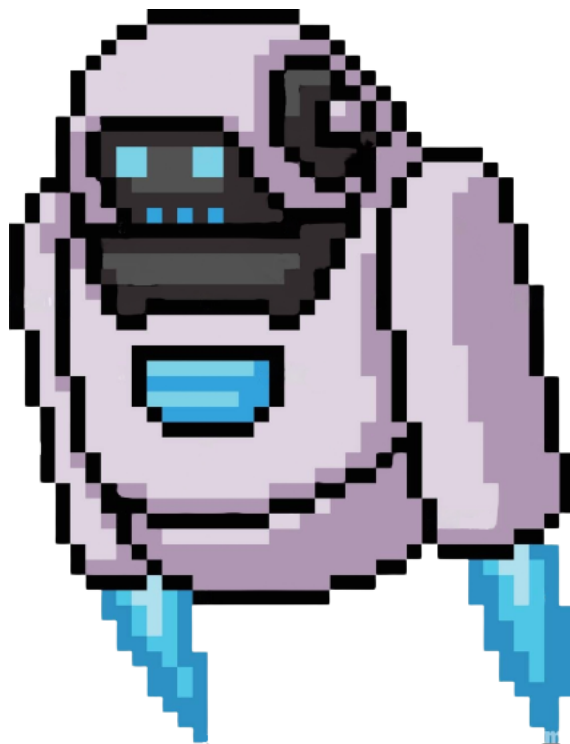


FIGURE 6.3 – Gros robot - Level 1 uniquement

**Boss** Ennemi principal du jeu présent dans le dernier niveau et que l'utilisateur doit battre pour gagner.

**Autres éléments graphiques** Éléments présents dans le jeu, ne présentant aucune mécanique particulièrement complexe, mais permettant d'enrichir l'expérience utilisateur via des interactions simple ou uniquement par leur présence esthétique.



FIGURE 6.4 – Serveur informatique - Level 1 et Level 3





FIGURE 6.5 – Bibliothèque basse - Level 1 et Level 2



FIGURE 6.6 – Bibliothèque haut - Level 1 uniquement