

OK Computer
Knight's Journey: Sword and Serenity
Rapport de projet

Victor Epinette, Rayane Gomri, Kévin Petit
Promo 2027

31 mai 2023

Table des matières

1	Introduction	4
2	Origine et nature du projet	5
3	Etat de l'art	6
3.1	L'histoire des RPG/Aventure	6
3.2	Notre idée de jeu	7
4	Moyens logiciels	9
5	Découpage du projet	10
5.1	Les différents axes	10
5.2	Tableau de répartition	12
5.3	Tableau d'avancement	13
6	Réalisations de la première période	14
6.1	Création de la map (Victor)	14
6.2	Animation du joueur (Victor)	15
6.3	Suivi de la caméra (Victor)	15
6.4	Ennemis et combat tour par tour (Victor)	15
6.5	Connexion au multijoueur Photon (Kévin)	17
6.6	Interfaces et menus (Kévin)	18
6.7	Site web (Kévin)	19
6.8	Combat au tour par tour et IA (Rayane)	19
7	Réalisations de la seconde période	21
7.1	Refonte du système de combat au tour par tour (Victor)	21
7.2	Remote Procedure Calls de Photon (Victor)	22
7.3	Animation d'ennemi de donjon (Victor)	23
7.4	Tilemaps des donjons et transitions entre les scènes (Kévin)	23
7.5	Scripts audio : musique et effets sonores (Kévin)	25

7.6	Mise à jour du site web (Kévin)	25
7.7	Barre de vie et inventaire (Rayane)	26
8	Réalisations de la dernière période	27
8.1	Combat en multijoueur fonctionnel (Victor)	27
8.2	Finalisation de la barre de vie et de l'inventaire (Victor)	27
8.3	Réorganisation des scènes pour l'optimisation et gestion de l'audio (Kévin)	28
8.4	Dernières maps : maps de combat, nouveau donjon, labyrinthe, boss et village (Kévin)	30
8.5	Animations du boss et des villageois (Kévin)	34
8.6	Animations du sorcier et des squelettes (Victor)	35
8.7	Création du script d'affichage des dialogues (Victor)	35
8.8	Game Over (Kévin et Victor)	36
8.9	Refonte de l'écran titre et des menus (Victor et Kévin)	37
8.10	Mise à jour finale du site (Kévin)	38
8.11	Réalisation de l'installateur (Kévin et Victor)	39
9	Récit de la réalisation	40
9.1	Modification importante du groupe lors de la première période	40
9.2	Première période	41
9.3	Seconde période	43
9.4	Dernière période	48
10	Conclusion	50
11	Annexes	51

1 Introduction

Nous en venons à la fin de ce projet. Son développement fut plus ardu que prévu, avec une constitution de groupe très changeante dès le départ. Mais nous avons malgré tout réussi à offrir un jeu fonctionnel et relativement complet. Le cahier des charges était probablement un peu trop ambitieux dès le départ et l'était pour sûr après les changements de groupe. Cependant, les lignes principales de ce dernier sont respectées. Nous avons produit un jeu d'aventure dans un style pixel-art comme prévu dès l'origine du projet.

Vous verrez également dans la partie sur le Récit de la réalisation de cette dernière partie que les problèmes internes au groupe se sont finalement répétés et que la situation ne s'est pas améliorée. Le développement a donc été réalisé à deux encore une fois avec une absence de retour de travail (malgré plusieurs relances une fois de plus).

Quoi qu'il en soit, nous sommes dans l'ensemble fiers du travail accompli. Le jeu offre une expérience relativement complète et stable. Ce rapport retrace l'ensemble de l'évolution du projet, de ses prémisses à son rendu final. Il reprend des éléments du cahier des charges et des rapports de soutenances précédemment rendus afin d'offrir une revue complète de l'ensemble du développement.

2 Origine et nature du projet

Plus jeunes, notre enfance a été marquée par de nombreux jeux. Mais parmi tous ces jeux, les RPG et les jeux d'aventure en 2D vue du dessus, partageant un style graphique proche, nous sont particulièrement chers. Nous pouvons citer par exemple la saga *Pokémon*, qui en était à ses opus sur Nintendo DS pendant nos plus jeunes années. Puis en grandissant et en apprenant plus sur l'histoire du jeu vidéo, nous avons pu découvrir les jeux du passé qui ont inspiré les jeux d'aujourd'hui : *The Legend Of Zelda : A Link To The Past*, *EarthBound* ou encore les opus Super Nintendo de *Final Fantasy*. Bien que leurs mécaniques diffèrent en partie, ces jeux ont en commun ce style graphique pixel-art chatoyant et coloré que nous apprécions tous les quatre.

En plus de cela, nous nous sommes déjà intéressés plus jeunes à l'utilisation de *RPG Maker*, logiciel permettant la création de notre propre RPG tout en évitant d'avoir directement recours à la programmation. Une interface graphique permet de facilement gérer son level-design et de prévoir les évènements à déclencher selon les interactions du sprite du joueur avec son environnement, facilitant grandement la création d'un jeu aux non-initiés de la programmation. Mais nous ne sommes désormais plus des non-initiés...

3 Etat de l'art

3.1 L'histoire des RPG/Aventure

Les RPG, sigle de *Role Playing Game* (jeu de rôle en français), ont d'abord fait leur apparition comme jeux de plateau. *Donjons et Dragons*, créé en 1974, est l'un des précurseurs du genre. Chaque joueur crée son personnage et le fait évoluer dans une histoire narrée par un maître du jeu qui a pour charge l'organisation de la partie et son bon déroulement. L'appréciation d'une partie de RPG repose alors grandement sur le talent de narration du maître du jeu. Les parties s'organisent généralement autour de lancers de dés qui déterminent le sort des joueurs (dans un combat par exemple) selon leur résultat.

Le premier RPG sorti en jeu vidéo est *dnd*, un jeu s'inspirant des mécaniques de *Donjons et Dragons* développé en 1975 par deux étudiants américains sur le système informatique de leur université du sud de l'Illinois. L'ordinateur joue alors le rôle du maître du jeu et les lancers de dés sont simulés par des algorithmes de génération de nombres aléatoires.

Les RPG publiés sur PC et consoles commencent dès lors à fleurir et se divisent progressivement en deux branches :

- les RPG occidentaux, laissant une assez grande part de liberté au joueur et d'abord publiés majoritairement sur PC
- les RPG japonais, abrégés en JRPG, dans lesquels le scénario occupe une place plus importante et qui guident donc plus le joueur au long de son aventure, publiés dans un premier temps sur consoles

En ce qui concerne les RPG actuels, on peut citer plusieurs grands noms :

- la saga *Final Fantasy*, dont le dernier opus est sorti cette année
- *Undertale*, jeu indépendant acclamé par la critique sorti en 2015

- *Fallout*, du genre Action-RPG, et son dernier opus *Fallout 76* sorti en 2018
- la saga *Pokémon*, débutée en 1996 au Japon et toujours éditée aujourd'hui avec l'opus *Pokémon Ecarlate et Violet*

Du côté des jeux d'action-aventure qui ont débuté avec une vue 2D du dessus similaire à celle de RPG mentionnés précédemment, on peut également citer :

- la saga *The Legend Of Zelda*, débutée en 1986 sur NES (Famicom au Japon) et en 2D jusqu'en 1998 avec la sortie d'*Ocarina Of Time* en 3D sur N64
- *Secret Of Mana*, édité en 1993 sur Super Nintendo, qui partage à la fois les caractéristiques d'un RPG et celles d'un jeu d'action-aventure

3.2 Notre idée de jeu

Notre jeu serait donc comme résumé précédemment un jeu du style RPG/Aventure dans lequel le joueur évoluerait avec une vue 2D du dessus.

Le jeu reposera sur un monde rempli de plusieurs biomes (forêt, désert, neige, jungle, montagne...) dans lesquels seraient répartis différents donjons. Avant d'entrer dans un donjon, le joueur devra choisir un outil parmi la sélection proposée (arc, épée, lance...). Au coeur de ces donjons se trouveront des énigmes ou des épreuves à passer, dont la résolution différera selon l'objet spécial choisi. Certains objets seraient alors plus adaptés que d'autres à la résolution de certains donjons.

Dans ces donjons se trouveront également des ennemis à vaincre pour pouvoir poursuivre son chemin. L'affrontement différera lui aussi dans son style selon l'outil choisi précédemment, qui pourra être utilisé pour combattre. De plus, des évènements aléatoires pourront avoir lieu en combat selon le biome dans lequel évolue alors le joueur : tempêtes de sable dans le désert, tempêtes de neige dans le biome hivernal, chutes de pierres en

montagne... Chacun de ces évènements aura une influence sur l'issue du combat, pouvant handicaper ou avantager le joueur et/ou son adversaire.

Une fois le "donjon" achevé, le joueur recevra une récompense différente selon l'objet spécial sélectionné : cela permettrait ainsi de rendre unique chaque partie, l'issue dépendant de la manière de jouer employée.

En dehors de ces donjons, le monde serait par ailleurs constitué de villages dans lesquels le joueur pourra interagir avec des PNJ (personnages non joueurs) lui permettant d'accomplir certaines actions exclusives à la zone du village.

Le jeu serait d'ailleurs plus proche du style d'un JRPG, avec un gameplay relativement guidé pour faciliter la progression du joueur et des combats se déroulant au tour-par-tour dans une interface dédiée et non pas "en direct".

L'implémentation du multijoueur permettrait au joueur de former un duo avec un autre joueur en réseau pour combattre les ennemis en équipe. Le second joueur pourra également choisir un autre objet spécial que son coéquipier et ainsi permettre une résolution facilitée des donjons : la combinaison des capacités des deux objets ouvre alors au duo plus de possibilités dans leur aventure.

4 Moyens logiciels

Grâce à ce projet, nous allons pouvoir découvrir comment programmer concrètement un jeu vidéo de A à Z. L'expérience que nous vivions jusqu'à présent en tant qu'utilisateurs se trouve renversée : nous devenons les créateurs. Malgré des compétences que nous possédons déjà dans quelques domaines, nous savons aussi que nous partons de zéro dans plusieurs facettes du développement.

La réalisation de projet nous permettra ainsi à tous de gagner en compétence dans le maniement de plusieurs outils informatiques, chacun remplissant un rôle spécifique dans la réalisation du jeu :

- **C#**, langage orienté objet qui permettra de programmer le jeu et de lier les éléments en un tout cohérent
- **Unity**, framework qui sera utilisé comme exigé pour notre projet de type jeu, nous permettant de poser les fondations de notre programme
- **Tiled**, logiciel libre permettant de créer des niveaux bitmap
- **Rider** ou **Visual Studio Code** selon les préférences de chacun comme environnement de programmation
- **HTML5, CSS3** et **PHP** pour la création du site web présentant notre projet
- **MySQL** pour la base de données de notre site
- **Paint.NET** pour l'édition des assets graphiques
- **Audacity** pour l'édition audio (musique et effets sonores)
- **Overleaf**, éditeur LaTeX (utilisé notamment pour la rédaction de ce cahier des charges et pour les futurs rapports)

En cas de recours à des solutions logicielles non-mentionnées ici en cours de développement, cette liste sera actualisée.

5 Découpage du projet

La réalisation de ce projet éprouvera nos compétences de coopération et de travail en équipe, indispensables dans le domaine informatique. La planification et la répartition sont alors des points essentiels pour assurer la réussite de ce projet et l'entente du groupe tout au long de la phase de développement.

5.1 Les différents axes

Pour une répartition des tâches simplifiée, le projet a été découpé en huit axes :

- Gameplay
- Intelligence artificielle
- Multijoueur/Réseau
- Graphismes/design
- Audio
- Menus et interface
- Installateur
- Site Web

Parmi ces huit axes, trois concernant tout particulièrement la partie code du projet ont été redécoupés en sous-tâches spécifiques : ces axes sont les axes Gameplay, IA et Multijoueur/Réseau.

Gameplay (basé en grande partie sur de la POO) :

- *Joueur* : gestion des déplacements du joueur, de ses attributs/caractéristiques, de son inventaire, de ses interactions avec la map et des actions qu'il est capable de réaliser selon les conditions remplies (ou non) par ses attributs et son environnement

- *Ennemis* : mise en place des différents attributs des ennemis qui seront ensuite réutilisés dans la partie IA pour la gestion des combats, comme les caractéristiques d'attaque, de défense ou encore des coups spéciaux
- *Objets* : implémentation des différents items qui peuvent être récupérés par le joueur, implémentation de leur comportement si leur utilisation est possible directement sur la map (armes par exemple) ou de leur effet sur les statistiques et compétences du joueur (dans le cas de potions notamment)
- *Map* : mise en place des bornes de la map par rapport aux graphismes créés, placement et gestion des évènements déclenchables par le joueur, élaboration et implémentation des énigmes présentes sur la carte grâce à la création de ces évènements

Intelligence artificielle :

- *Comportement dans les niveaux* : déplacements éventuels des ennemis, ainsi que leur placement et leur répartition sur la map par rapport au joueur
- *Comportement en combat* : manière des ennemis de répondre aux attaques de façon plus ou moins violente/compétente selon des facteurs d'expérience et d'environnement notamment, adaptabilité relative aux coups du joueur

Multijoueur/Réseau :

- *Interactions en jeu* : implémentation du fonctionnement du multijoueur dans les maps des niveaux, interactions possibles entre les joueurs, déplacement de l'affichage de la map selon la position des deux joueurs l'un par rapport à l'autre
- *Combats en duo* : fonctionnement du combat à deux joueurs, interaction avec la partie IA pour expliciter à l'IA l'évolution de son comportement face à deux joueurs plutôt qu'un

- *Connexion* : mise en relation à distance des machines des deux joueurs sur le réseau, gestion des données à envoyer en permanence pour actualiser l'état du jeu

5.2 Tableau de répartition

Répartition	Victor	Rayane	Kévin
Gameplay			
<i>Joueur</i>	R		<i>S</i>
<i>Ennemis</i>	R		<i>S</i>
<i>Objets</i>	R	<i>S</i>	
<i>Map</i>	R		<i>S</i>
IA			
<i>Niveaux</i>	<i>S</i>	R	
<i>Combats</i>	<i>S</i>	R	
Multijoueur/Réseau			
<i>Interactions en jeu</i>		<i>S</i>	R
<i>Combats en duo</i>		<i>S</i>	R
<i>Connexion</i>	<i>S</i>		R
Graphismes/design	<i>S</i>	R	
Audio	R		<i>S</i>
Menus/interface		<i>S</i>	R
Installateur		R	<i>S</i>
Site Web		<i>S</i>	R

R : responsable de la tâche

S : suppléant

Cette répartition a été élaborée selon les compétences et les préférences de chacun. Vous trouverez comme demandé pour chaque tâche le responsable et le suppléant assignés. Nous avons essayé de garder une relative

équité dans ce partage des tâches vis-à-vis de la charge de travail à prévoir.

5.3 Tableau d'avancement

Planning	Soutenance 1	Soutenance 2	Soutenance 3
Gameplay			
<i>Joueur</i>	50%	80%	100%
<i>Ennemis</i>	50%	80%	100%
<i>Objets</i>	40%	70%	100%
<i>Map</i>	40%	70%	100%
IA			
<i>Niveaux</i>	20%	60%	100%
<i>Combats</i>	30%	70%	100%
Multijoueur/Réseau			
<i>Interactions en jeu</i>	30%	80%	100%
<i>Combats en duo</i>	20%	60%	100%
<i>Connexion</i>	60%	90%	100%
Graphismes/design	40%	70%	100%
Audio	0%	50%	100%
Menus/interface	25%	75%	100%
Installateur	0%	0%	100%
Site Web	40%	70%	100%

Nous avons priorisé le gameplay pour la première soutenance, ce dernier restant la fondation de n'importe quel jeu. Les graphismes, les menus, l'IA et le multijoueur devraient quant à eux suivre une évolution relativement linéaire. Le site Web évoluera progressivement avec le jeu, mis à jour au fil des changements.

L'audio pourra être débuté plus tard, le jeu pouvant se passer de son pour la première soutenance. Enfin, l'installateur ne sera réellement nécessaire que lors de la dernière soutenance.

6 Réalisations de la première période

6.1 Création de la map (Victor)

Le projet a débuté avec la création de la carte de l'overworld par Victor. D'abord passés par le système de tuiles d'Unity, nous avons remarqué de nombreux bugs visuels. Nous avons alors migré vers une autre application nommée Tiled et avons assemblé cette carte à partir d'assets trouvés sur le web, notamment OpenGameArt ou encore Itch.io. Comme son nom l'indique, l'application Tiled permet l'assemblage de tuiles bitmap pour former un niveau. La dimension finalement retenue pour les tuiles de notre jeu est de 16 pixels par 16.

Les collisions et autres bordures des éléments de la map ont alors été réalisées avec l'aide du plugin SuperTiled2Unity. Ce plugin permet de créer un masque de collisions associé à la map qui sera directement importé dans Unity. Ainsi le sprite du joueur ne peut pas passer à travers les troncs des arbres ou les pierres.



6.2 Animation du joueur (Victor)

Victor s'est ensuite attelé à l'animation du sprite du joueur et à son déplacement sur la map. Le sprite du joueur peut donc être aperçu remuant des jambes lorsqu'il reste sur place. Il se réoriente vers la gauche ou la droite selon la direction de sa course et dispose d'une animation d'attaque avec son épée lorsque la barre espace est pressée. Le personnage ne possédant que des animations vers la gauche et la droite, nous avons dû ruser et fixer des animations par défaut sur les déplacements verticaux. Dans ce cas, animation gauche quand il monte, et droite quand il descend. De même pour les attaques orientées vers le haut ou le bas. Enfin, les animations d'attaque ont été réalisées par Victor. Ces sprites sont une combinaison des différents assets récupérés.



6.3 Suivi de la caméra (Victor)

Il a fallu programmer la caméra de manière à ce qu'elle suive le joueur dans ses déplacements sans pour autant laisser apercevoir les bordures de la carte. A l'approche des extrémités, la caméra n'est alors plus centrée sur le joueur et stoppe son suivi avant de déborder de la map. Le déplacement de la caméra gère également le passage entre les différents tableaux d'une même carte. La caméra suit le joueur avec un léger délai ce qui permet de fluidifier le gameplay.

6.4 Ennemis et combat tour par tour (Victor)

Un premier type d'ennemi a été implémenté par Victor. Il est animé dans les quatre directions : haut, bas, gauche et droite. Il est aussi doté d'une simple intelligence artificielle qui lui permet de détecter la présence

d'un joueur à proximité dans un certain rayon. Si un joueur est reconnu, l'ennemi commence alors à le suivre jusqu'à ce qu'il ressorte du rayon de détection. Si les deux rentrent en contact le combat tour par tour se lance. Il a été nécessaire d'adapter ce système au multijoueur et de créer ainsi une liste de joueurs pour que l'ennemi comprenne qui attaquer en priorité.



Enfin, l'interface de combat au tour par tour a été mise en place. Pour l'instant, il est simplement possible d'attaquer l'ennemi avec la barre espace. Les jauge de vie du joueur et de l'ennemi sont visibles. Le système complet de combat sera intégré d'ici la prochaine soutenance et intègrera le combat à deux ainsi que l'inventaire du joueur.



6.5 Connexion au multijoueur Photon (Kévin)

Kévin quant à lui s'est occupé de la gestion du plugin Photon. Il s'est chargé de la connexion des deux joueurs au même serveur Photon et des problématiques de synchronisation des deux joueurs. Il a alors fallu intégrer au préfabriqué du joueur différents composants Photon :

- Photon Transform Classic View pour synchroniser la position des joueurs
- Photon Animator View pour synchroniser les animations des sprites
- Photon View pour que les commandes effectuées par un des deux joueurs n'influent pas sur son coéquipier
- Photon Rigidbody 2D View pour que les collisions entre les joueurs n'entraînent pas un glissement sur le terrain et une désynchronisation des positions

Il a également fallu gérer lors de la connexion l'instanciation d'un joueur et de sa caméra associée. En effet, le GameObject du joueur ne pouvait plus être laissé en attente sur la scène de l'overworld et devait être généré à partir d'un préfabriqué lors de chaque connexion à la même partie. Contrairement au joueur, la caméra ne devait pas être synchronisée sur Photon, chaque joueur n'ayant à voir que sa propre caméra.

6.6 Interfaces et menus (Kévin)

Le menu du lobby a été conçu dans le même temps par Kévin. Il dispose très simplement d'une option de création de partie pour le premier joueur et de connexion à une partie pour le second joueur cherchant à rejoindre son ami. Il suffit alors au premier joueur de transmettre son nom de partie au deuxième.

Il lui a alors fallu créer le reste des menus pour l'expérience de navigation du joueur. Un écran titre accueille le joueur avec une courte vidéo diffusée en boucle en arrière-plan, montrant le joueur s'animant. Cet écran propose au joueur d'accéder au lobby ou de quitter le jeu. Il pourra dans le futur proposer également l'accès à quelques options, telles que la résolution du jeu par exemple ou le volume du son. Si le joueur choisit de lancer une partie, l'écran de connexion d'affiche alors. C'est durant cette simple scène que s'exécute en fond le script de connexion au serveur Photon. Une fois l'accès au lobby effectué, le joueur peut décider de revenir au menu principal, ce qui le déconnecte alors du serveur Photon.



6.7 Site web (Kévin)

Kévin s'est également occupé de la conception de la base du site web présentant le projet. Ce site n'est pas encore complet et s'étoffera au fil des soutenances. Il intègre cependant déjà une page d'accueil, une page de présentation des premières phases du projet, une foire aux questions et une page de contact. Des captures des différents tableaux du jeu servent de fond aux pages du site. Le site étant pour l'instant présenté uniquement localement, la page de contact est "fictive" et n'envoie pas réellement de message. Le site est pour l'instant constitué de code HTML et CSS. Il intégrera dans le futur un lien de téléchargement de l'installateur du jeu.

6.8 Combat au tour par tour et IA (Rayane)

Nous nous étions mis d'accord avec Rayane pour qu'il réalise cette tâche. Malheureusement, Rayane ne nous a fait parvenir aucune réalisation concrète malgré nos multiples relances. Nous avons reçu ce paragraphe de sa part, décrivant sa tâche à 22h, la veille de la soutenance :

"J'ai donc travaillé sur les IA et les combats tour par tour.

Dans un premier temps, j'ai dû définir le comportement de l'IA : dans ce cas un comportement neutre se baladant aléatoirement et courant sur le joueur dès qu'il rentre dans la portée de l'IA. Deuxièmement, j'ai dû donc choisir une méthode d'IA et c'est là que tout se complique : pour le moment je n'ai toujours pas testé la bonne méthode après l'essai des arbres de comportement et des réseaux de neurones, mais je compte en tester d'autres jusqu'à trouver celle qui me plaît. Troisièmement, je compte donc programmer l'IA en C# pour enfin l'intégrer et la tester.

Combat tour par tour :

Après avoir trouvé l'IA et que celle-ci fonctionne, je compte donc créer un système de combat tour par tour. Pour cela, je dois d'abord définir les éléments du combat tels que les personnages, les ennemis, les attaques et les défenses. Puis je dois créer un système de tour où chaque personnage peut prendre une action à son tour. Je peux aussi utiliser une variable pour suivre le personnage. Mais rien n'est encore fait, je ne garantis pas que le résultat final sera identique."

7 Réalisations de la seconde période

7.1 Refonte du système de combat au tour par tour (Victor)

La première version naïve de combat au tour par tour présentait plusieurs défauts et bugs. Par exemple, la disparition de l'ennemi n'était pas encore prise en compte une fois celui-ci vaincu par le joueur. La réapparition du joueur sur l'Overworld était aussi défectueuse, celui-ci revenant au point de spawn plutôt qu'à sa dernière position. Victor s'est donc attelé à la refonte de ce système de combat. Il prend désormais mieux en compte les statistiques des joueurs et de ses ennemis lors de l'application des dégâts. Ce nouveau système corrige également les bugs mentionnés précédemment. A présent, l'ennemi est bel et bien détruit et le joueur revient sur l'Overworld à sa dernière position. Quelques changements ont aussi été apportés à la scène pour mieux la préparer au multijoueur, qui rencontre encore des bugs handicapants lors du lancement d'un combat.



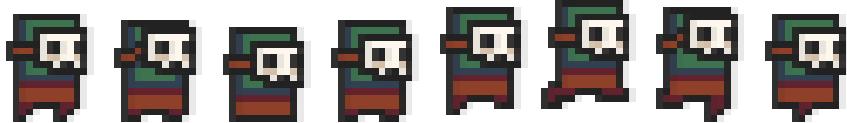
7.2 Remote Procedure Calls de Photon (Victor)

Victor s'est également penché dans le même temps sur les Remote Procedure Calls du network multijoueur Photon. En effet, l'implémentation des RPC, effectuée dans le même temps que la refonte du combat au tour par tour, permet de mieux préparer le terrain pour les prochaines fonctionnalités multijoueur. Les RPC sont un système de messages entre les clients Photon des différents joueurs. Ces messages permettent de synchroniser l'état des clients entre eux avec plus de facilité et plus de flexibilité que nos scripts le permettaient auparavant.

Ces ajouts n'ont pas été sans problèmes : il a fallu faire face à de nouvelles problématiques soulevées par cette nouvelle conception. Un problème rencontrée a été un dysfonctionnement des méthodes de connexion à une Room de jeu Photon, ce qui a partiellement ralenti le développement le temps de trouver la solution appropriée après de longues recherches en ligne.

7.3 Animation d'ennemi de donjon (Victor)

Enfin, Victor a réalisé l'animation du nouvel ennemi de donjon, visible ci-dessous. Le processus a été relativement similaire à l'animation du premier ennemi lors de la première phase de développement. Comme la bûche ennemie de la map Overworld, un suivi du joueur s'active lorsque celui-ci rentre dans sa zone d'influence, marquée par un collider circulaire. Ce nouvel ennemi possède en revanche des statistiques différentes qui lui permettront de se distinguer en combat. Cet ennemi est intégré au premier donjon réalisé par Kévin, visible dans la section qui suit.



7.4 Tilemaps des donjons et transitions entre les scènes (Kévin)

De son côté, Kévin s'est d'abord chargé de la création des deux premiers donjons, accessibles depuis l'Overworld. Ces deux nouvelles maps ont été conçues comme les précédentes à l'aide du logiciel Tiled et du plugin SuperTiled2Unity. Les tilemaps ont d'abord été assemblées dans l'éditeur de Tiled grâce à des tilesets 16x16 libres, trouvés sur OpenGameArt.org. L'utilisation du plugin SuperTiled2Unity a ensuite permis d'importer sans problème la totalité des calques et des masques de collisions dans le jeu.

Il a alors fallu ajouter en plus de cela les transitions nécessaires entre les scènes, permettant de passer de l'Overworld à chaque donjon respectif et inversement. Cela a été réalisé à l'aide de sprites 2D sans rendu, contenant simplement un Collider servant d'interrupteur déclenchant le changement de scène lors de la rencontre avec le sprite d'un joueur. L'appel d'un DontDestroyOnLoad s'est alors avéré nécessaire pour permettre la conservation du joueur et de sa caméra entre les scènes. Un simple script gère donc

le déplacement de la position relative du joueur et de sa caméra lors des chargements des donjons afin d'apparaître au point de spawn souhaité.



Tilemap du premier donjon avec masques de collision visibles



Deuxième donjon

7.5 Scripts audio : musique et effets sonores (Kévin)

La partie audio a été assurée par Kévin. Les musiques de fond des menus et des différents niveaux ont été sélectionnées sur Incompetech, un site du compositeur de musiques libres de droit Kevin MacLeod. Il a alors fallu chercher des musiques à l'ambiance relativement cohérente avec le style de chaque map. Ces musiques devaient aussi rester discrètes dans le cas des menus. Le son a été assez rapide à implémenter grâce aux composants AudioSource de Unity. La seule problématique à gérer a été la conservation de la même musique sans redémarrage entre les différentes scènes composant le menu principal. Il a donc encore été question de DontDestroyOnLoad pour permettre la sauvegarde du GameObject contenant la musique dans son état actuel.

En plus de la musique, quelques effets sonores ont été ajoutés aux différents personnages constituant le jeu. L'épée du chevalier contrôlé par le joueur fait entendre un son de lame filant dans l'air lorsque l'animation d'attaque est activée. Les bûches ennemis dans l'Overworld font également entendre un petit hoquet lorsque le joueur sort de leur zone d'influence, permettant ainsi de clarifier l'état alerte ou non de l'ennemi à l'utilisateur.

7.6 Mise à jour du site web (Kévin)

Kévin s'est enfin occupé de la mise à jour du site web afin que ce dernier présente toutes les caractéristiques attendues à la deuxième soutenance. L'interface complète du site web est disponible en annexe à la fin de ce rapport. Par rapport à la première soutenance, la section de présentation du projet mentionne désormais les nouvelles réalisations de cette deuxième période.

La section Liens est venue remplacer la FAQ, peu utile. Dans cette section Liens, on peut retrouver les liens menant aux sites dont les solutions logicielles et les ressources ont été utilisées (Tiled, SuperTiled2Unity, Open-GameArt, Incompetech...). La section contient également le lien de téléchargement du dossier de Build du projet et du rapport de soutenance. A

A l'heure actuelle, ces dossiers sont hébergés dans un repo GitHub.

La section Contacts a également gagné en partie en interactivité. Les adresses mails de l'équipe de développement ne sont plus de simples blocs de textes mais sont désormais des liens HTML *mailto*, permettant d'ouvrir directement le logiciel de messagerie par défaut de l'utilisateur sur un courriel adressé au développeur sélectionné.

7.7 Barre de vie et inventaire (Rayane)

Rayane s'est chargé de la réalisation d'une barre de vie pour le joueur, visible dans l'Overworld. Cette dernière utilise un Canvas affiché sur la caméra du joueur et s'actualise selon l'attribut HP contenu dans le script associé au GameObject du joueur. La création d'un système d'inventaire a également été débutée par Rayane. Ce système d'inventaire permet au joueur d'accéder à tous les objets et items interactifs qu'il a collectés, tels que des potions par exemple. A l'heure actuelle, cet inventaire n'est pas encore complètement fonctionnel.



8 Réalisations de la dernière période

8.1 Combat en multijoueur fonctionnel (Victor)

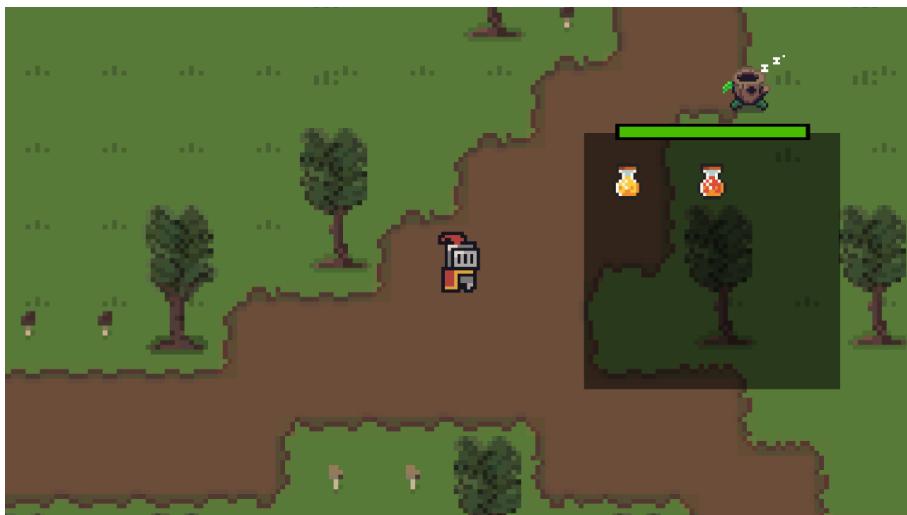
Après l'implémentation des Remote Procedure Calls réalisée durant la période précédente, la réalisation d'un système de combat fonctionnel multijoueur a été possible grâce à Victor.

Lorsque l'un des deux joueurs entre en collision avec un ennemi qui le poursuit, les deux joueurs sont téléportés dans la scène de combat en cours de chargement. Ils se retrouvent alors tous deux face à l'ennemi à vaincre. Le premier joueur (celui qui a créé la partie) attaque en premier, ce qui est signalé par sa barre de vie qui clignote ; Puis vient ensuite le tour du second joueur (celui qui a rejoint la partie). L'ennemi attaque en dernier : son attaque touche aléatoirement le premier ou le second joueur selon un appel à un objet Random à chaque tour. C'est donc en 3 étapes que ce combat se déroule en multijoueur : joueur *Create*, joueur *Join* et enfin l'ennemi.

Dès que l'ennemi est vaincu, les deux joueurs sont retéléportés à leurs positions dans la scène où l'ennemi vaincu a été précédemment rencontré. En revanche, si l'un des deux joueurs est vaincu par l'ennemi et se retrouve donc avec 0 PV, l'écran de Game Over est déclenché (voir la sous-section consacrée au Game Over).

8.2 Finalisation de la barre de vie et de l'inventaire (Victor)

Victor a réimplémenté de manière complète la barre de vie en jeu et l'inventaire. Ils sont tous les deux accessibles à n'importe quel moment en jeu (que ce soit en combat ou dans l'Overworld) par une pression de la touche E. Un Canvas s'ouvre alors et affiche les 9 emplacements de l'inventaire, ainsi que la barre de vie au dessus de cet inventaire.

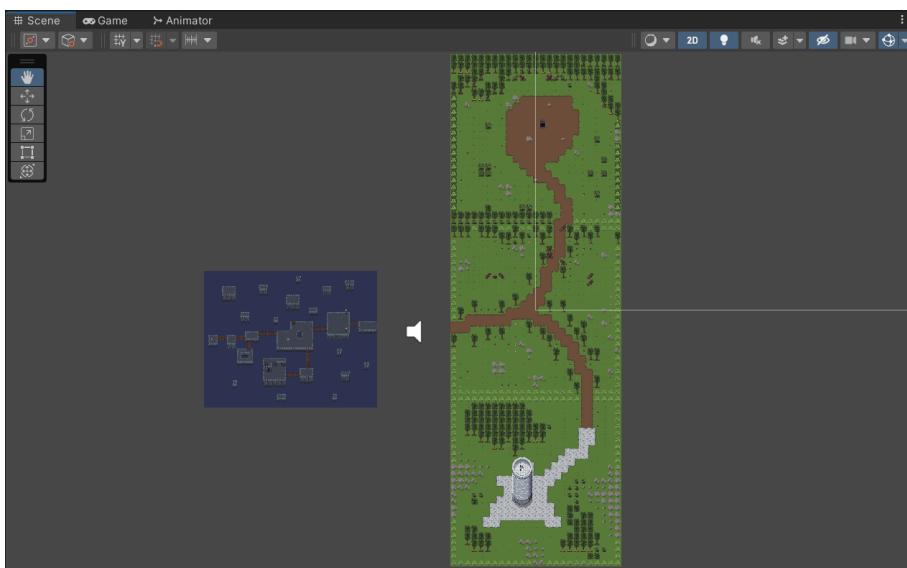


L'inventaire permet de contenir des potions disséminées tout au long de l'aventure. Il existe deux types de potions : des potions de vie, qui régénèrent une partie de la vie du joueur, et des potions de force, qui augmentent la puissance de l'attaque en combat de manière permanente. Les potions de vie sont affichées en rouges, tandis que celles de force sont jaunes. Leur utilisation se fait pas un simple clic gauche de souris sur la case de l'inventaire contenant la potion voulue.

8.3 Réorganisation des scènes pour l'optimisation et gestion de l'audio (Kévin)

Chaque map était auparavant dans sa propre scène, ce qui causait de nombreux temps de chargement d'une map à une autre. Pour supprimer ces chargements, il a été dans un premier temps envisagé de mettre toutes les maps dans la même scène et de simplement téléporter le joueur d'un endroit à l'autre de la scène. Face au manque d'optimisation de cette méthode, causant de nombreux ralentissements et rallongeant les temps de chargement entre les scènes de combat et l'Overworld, un compromis a été retenu.

Chaque scène (excepté le Labyrinthe) contient désormais 2 cartes. Cela a permis de réduire le nombre de chargements en remplaçant certains changements de scène par de simples téléports du joueur et de sa caméra tout en conservant une fluidité d'exécution raisonnable.



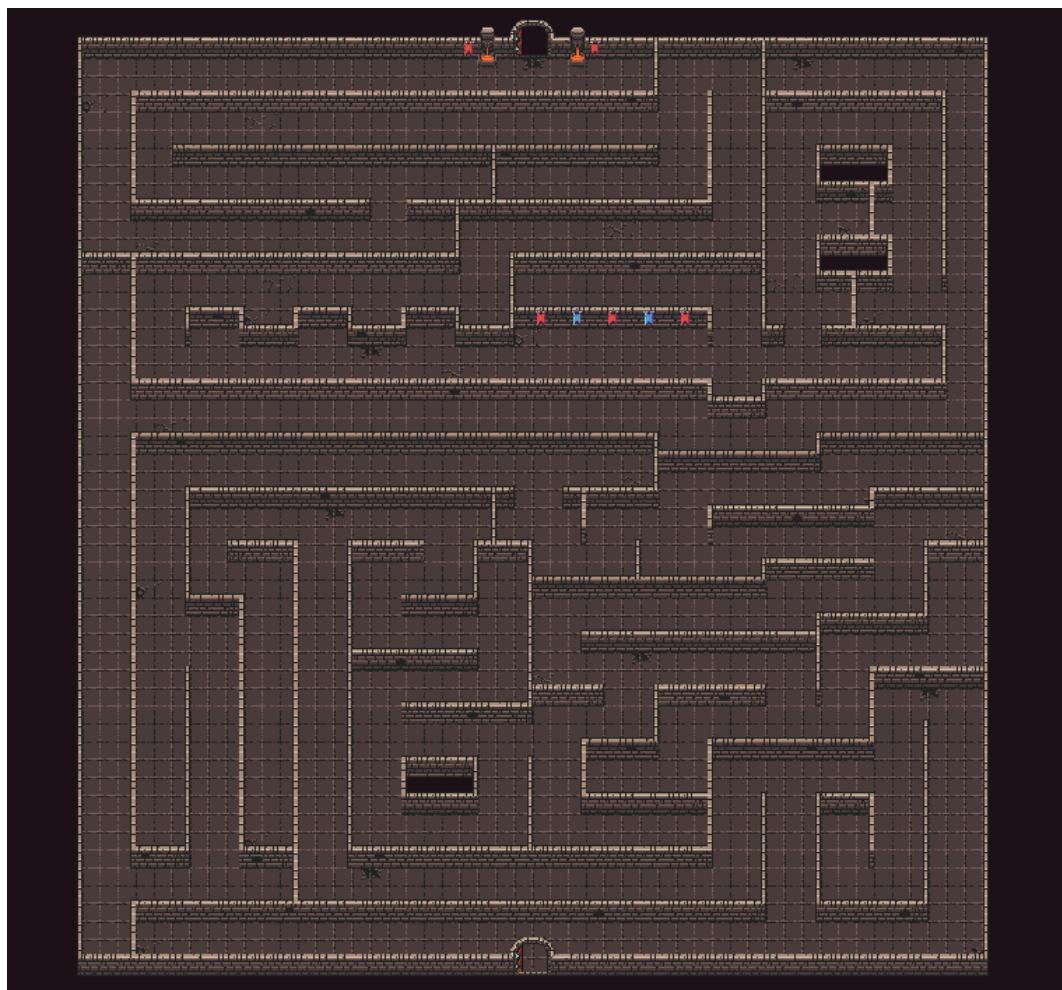
Cette réorganisation de l'agencement des scènes a aussi impliqué dans le même temps une gestion des changements de musique. En effet, chaque scène possédait avant sa propre source audio contenant la musique à déclencher dès le lancement de la scène. Après réagencement, il a fallu modifier ce fonctionnement pour permettre le changement de musique au sein d'une même scène en cas de téléport et pour lancer la bonne musique selon la position d'arrivée après un chargement sur une nouvelle scène.

8.4 Dernières maps : maps de combat, nouveau donjon, labyrinthe, boss et village (Kévin)

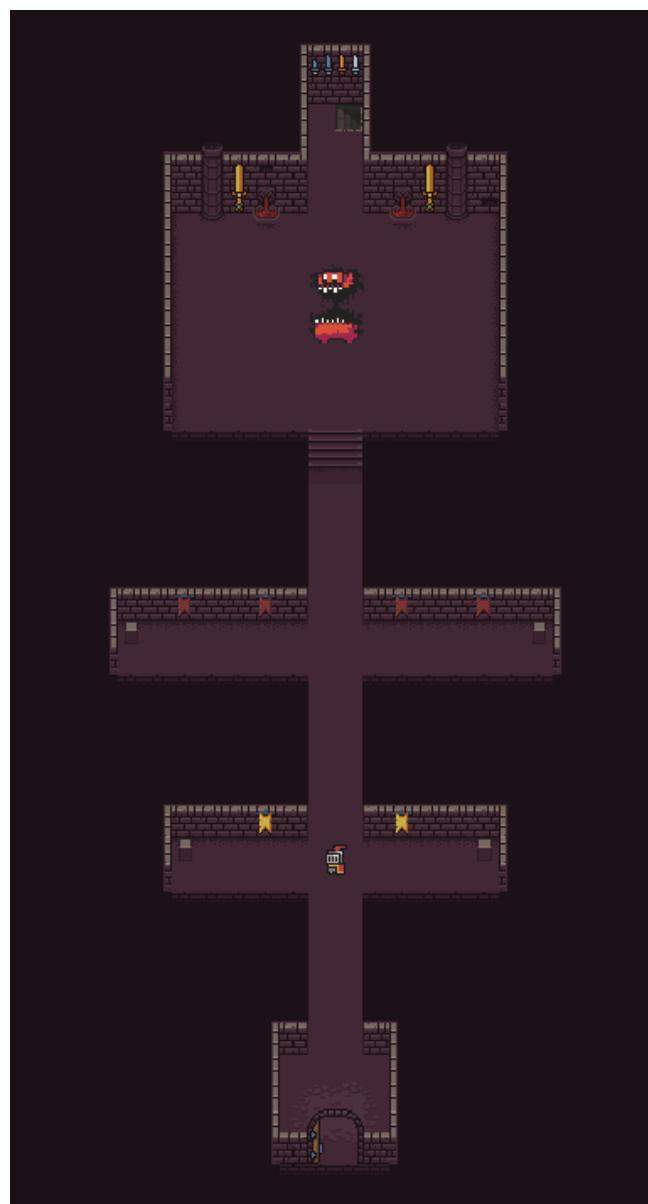
Le jeu avait besoin de nouvelles maps pour être plus complet. Kévin s'est donc chargé de la création des dernières tilemaps. Un nouveau donjon a été ajouté et permet de faire la liaison entre les deux donjons précédemment créés.



Un labyrinthe a ensuite été concu pour précéder l'arrivée dans la salle du boss final. Ce labyrinthe contient plusieurs potions de force pour préparer le joueur à l'affrontement du boss final.



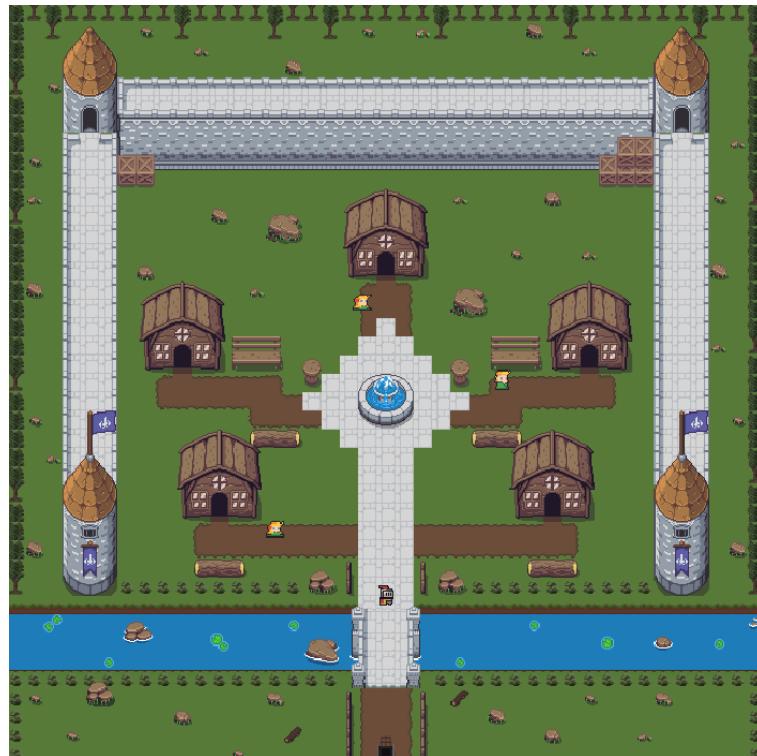
Le donjon final contient alors le boss à vaincre (voir sous-section animation du boss ci-dessous). C'est évidemment l'ennemi aux statistiques de vie et d'attaques les plus élevées du jeu, tout en restant un combat accessible et loin d'être punitif.



La map de combat associée a également été modélisée à cette occasion.



Enfin un village a été créé pour accueillir différents PNJ remerciant le joueur pour le service rendu (voir les sous-sections concernant leur animations plus bas). Ce village permet également d'accéder à une scène de remerciement final qui stoppe la partie et renvoie le ou les joueur/s à l'écran titre.



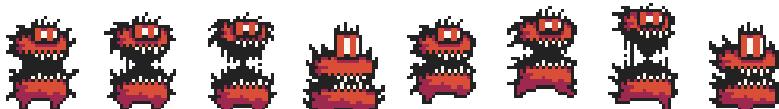
Comme pour les niveaux précédents, les tuiles utilisées viennent du site OpenGameArt.org, proposant du contenu libre d'utilisation.

8.5 Animations du boss et des villageois (Kévin)

Le boss visible dans le donjon final ainsi que les villageois en fin du jeu ont été animés par Kévin. Ces animations ont été réalisées à l'aide de l'Animator de Unity. Il a fallu créer des arbres d'état sélectionnant le mouvement des personnages selon les variables de direction X et Y dans un repère cartésien. L'animation en elle-même est rendue facile à réaliser grâce à la fenêtre d'animation de Unity, permettant de mettre les sprites correspondant à un mouvement particulier bout à bout et de choisir la durée totale de l'animation.



Le boss a également été implémenté avec son script dédié, descendant du script père de gestion de l'ensemble des ennemis. Cela a permis d'ajuster la vie du boss, sa vitesse de déplacement et sa force d'attaque notamment.

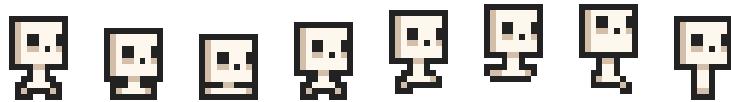


8.6 Animations du sorcier et des squelettes (Victor)

Le sorcier qui agit comme un PNJ a été animé par Victor, ainsi que les squelettes qui sont un nouveau type d'ennemis. L'animation en elle-même a suivi un procédé similaire à celui décrit précédemment, avec des sprites récupérés sur OpenGameArt une fois de plus.

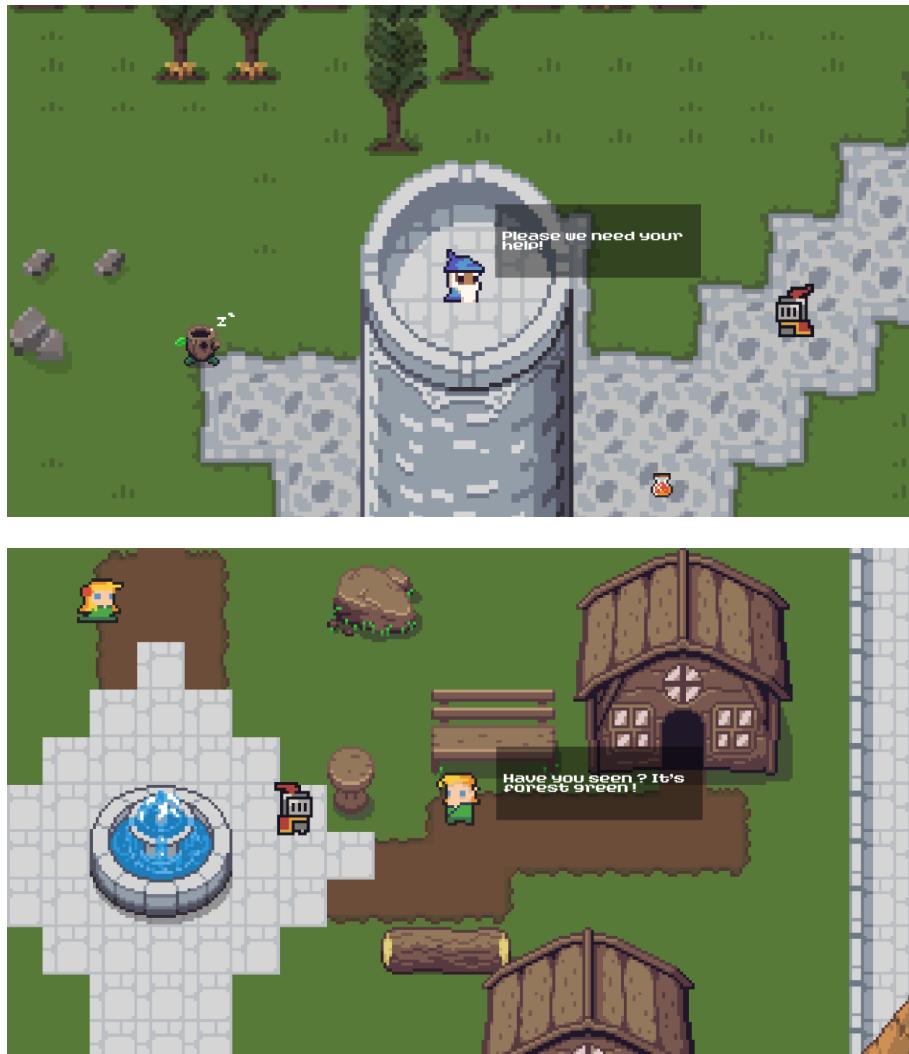


Comme pour le boss, le squelette a du être implémenté en tant que nouvel ennemi, avec ses caractéristiques de vie, de vitesse et d'attaque dédiées. Il possède lui aussi son script fils du script de gestion des ennemis.



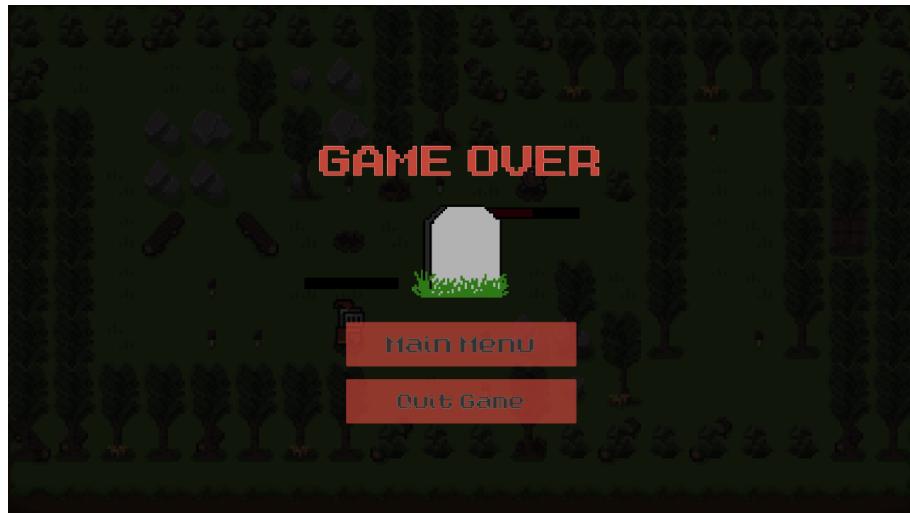
8.7 Création du script d'affichage des dialogues (Victor)

Une fois la gestion des animations des différents personnages non-joueurs mise en place par Victor (sorcier) et Kévin (villageois), Victor s'est chargé de la création d'un script qui permettrait aux PNJ de réciter un dialogue lorsque le joueur s'approche d'eux. Ce script prend en compte un collider circulaire placé tout autour d'un PNJ. Lorsque le joueur rentre dans la zone de ce collider, la caméra est transposée sur le PNJ et affiche à sa droite son dialogue animé. Si le joueur revient voir le personnage une fois le premier dialogue du PNJ déjà énoncé, celui-ci prononcera un second message différent.



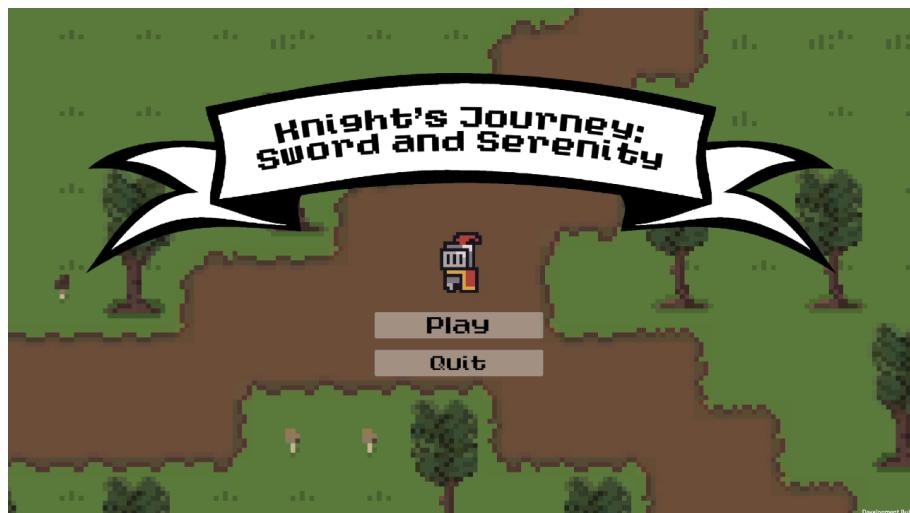
8.8 Game Over (Kévin et Victor)

L'écran de Game Over a été réalisé par Kévin. Son affichage se déclenche une fois que la vie du joueur tombe à 0 lors d'un combat. Ses fonctionnalités sont semblables au menu pause, avec possibilité de quitter directement le jeu ou de revenir au menu principal afin de retenter sa chance en relançant une partie. Victor a ensuite aidé à la gestion de ce menu dans le contexte du multijoueur.



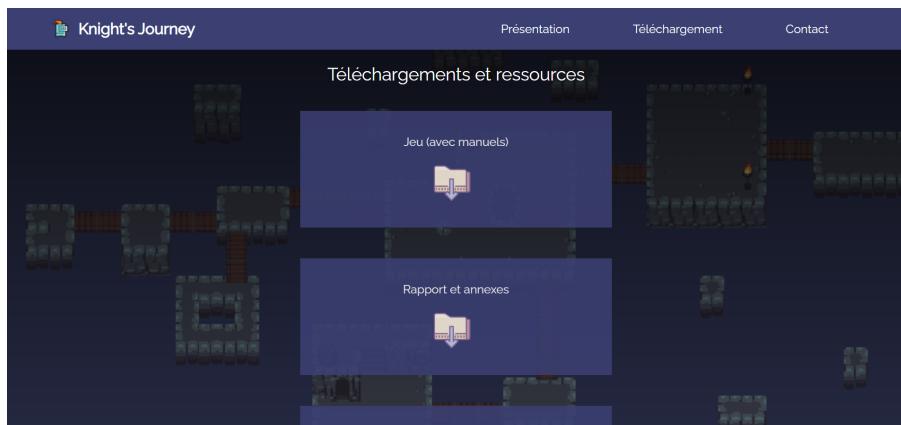
8.9 Refonte de l'écran titre et des menus (Victor et Kévin)

Pour mieux correspondre à la direction artistique de l'ensemble du jeu, la police, auparavant police par défaut des TextMeshPro de Unity, a été remplacé par une police au style "pixelart". Le principal accueille désormais le titre final du jeu dans une présentation similaire à celle visible sur la jaquette du jeu.



8.10 Mise à jour finale du site (Kévin)

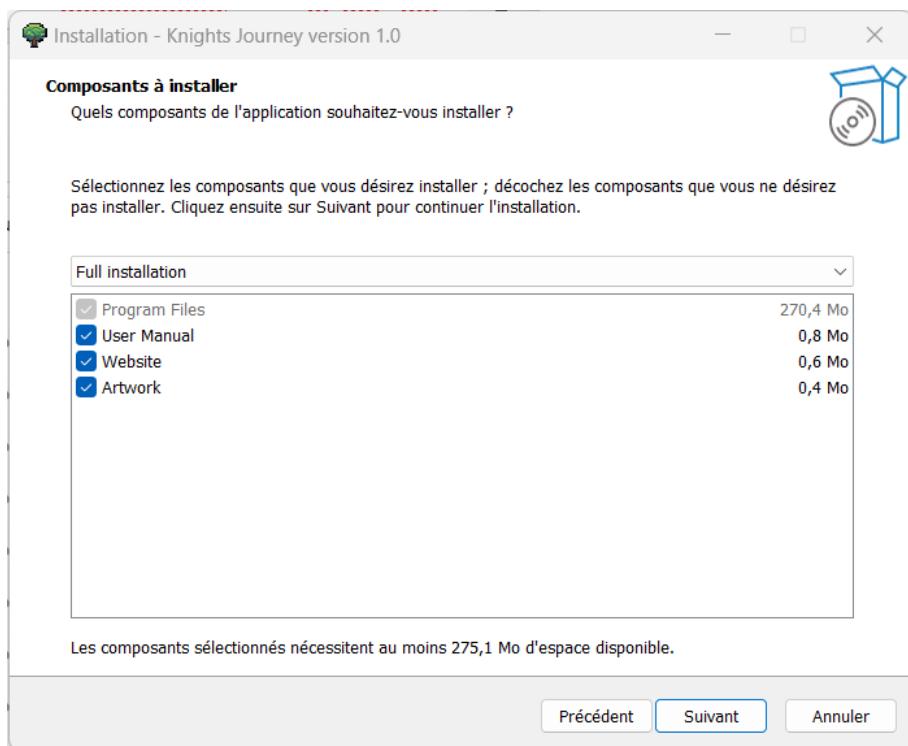
Kévin s'est chargé comme pour les soutenances précédentes de la mise à jour du site. Les liens de téléchargement ont été mis à jour pour permettre de télécharger sur GitHub l'installateur final du jeu, les manuels et les autres documents concernant le projet (rapport, jaquette...). La page de présentation du projet présente maintenant les derniers ajouts réalisés lors de cette période finale de développement. Enfin, plusieurs éléments graphiques ont été mis à jour pour correspondre à la direction artistique finale du jeu : l'icone de manette a été remplacée par le sprite du joueur, le titre final du jeu a remplacé le nom du groupe et les captures d'écran en fond permettent de voir des maps plus diversifiées.



Le reste de l'interface du site est visible en annexe.

8.11 Réalisation de l'installeur (Kévin et Victor)

L'installateur a été la dernière réalisation. La tâche a été réalisée de manière commune par Victor et Kévin. L'installateur a été réalisé avec l'aide de Inno Setup. Ce programme a permis de créer rapidement et efficacement un installateur qui télécharge l'exécutable du jeu, les données associées ainsi que les manuels sur l'ordinateur de l'utilisateur. L'utilisateur peut bien évidemment sélectionner le dossier dans lequel il souhaite installer le jeu et dispose d'un exécutable de désinstallation situé dans ce même dossier sous le nom *unins000.exe*. Il peut n'installer que le jeu avec l'installation minimale ou bien le jeu ainsi que le site web, le manuel et l'artwork du jeu avec l'installation complète.



9 Récit de la réalisation

9.1 Modification importante du groupe lors de la première période

Le groupe, initialement constitué de Victor Epinette, Séraphin Outrebon, Kévin Petit et Daniel Prietto-Ouezzanni, a connu deux défections entre les mois de décembre et février. Il y a tout d'abord eu le départ de Séraphin Outrebon mi-janvier, lorsqu'il a présenté sa démission à l'école. Le groupe a ensuite appris début février la césure de Daniel Prietto-Ouezzani pour ce second semestre, après plusieurs semaines sans nouvelles de sa part. Le groupe a donc pendant une certaine période évolué avec seulement deux membres "actifs", sans possibilité de changement ou de fusion de groupe sans signalement administratif de la césure/désinscription de Daniel pour cette année scolaire.

Rayane Gomri a alors rejoint le groupe le 10 février, portant le nombre final de membres à trois. Cette évolution relativement mouvementée du groupe n'a donc pas été sans causer quelques retards dans certaines portions du développement du jeu. Contrairement au groupe original, où tout le monde était au moins l'ami d'un des autres membres, Rayane était inconnu de Victor et Kévin. Il a ainsi fallu rapidement faire connaissance et présenter le concept du projet, les fonctionnalités qui avaient déjà été implémentées et les tâches qu'il restait à faire. La mission de réaliser l'interface de combat au tour par tour a été celle assignée en priorité à Rayane, après discussions sur les préférences de chacun.

Comme demandé, le cahier des charges a également été révisé en conséquence depuis l'arrivée de Rayane, en ce qui concerne plus particulièrement les tableaux de répartition et d'avancement du projet. Les tâches que devaient notamment réaliser Séraphin et Daniel ont ainsi été redistribuées et certains pourcentages d'avancement ont été légèrement revus à la baisse.

En ce qui concerne le tableau de répartition, nous avons tenté de redistribuer les tâches sans compromettre l'interdépendance des sections réalisées par chacun des membres. En ce qui concerne le tableau d'avancement, la progression des sections en rapport avec le système combat ont été réduites pour la première soutenance.

9.2 Première période

Avant les deux départs cités précédemment, les premières bases du projet avaient été posées. La solution de stockage et de partage des fichiers retenues a été GitLab plutôt que GitHub. En effet, le plan proposé sur GitLab par notre compte scolaire était bien plus avantageux en terme de stockage que le plan gratuit de GitHub. Les premiers fichiers ajoutés au répertoire GitLab ont été la template Unity 2D sur laquelle tous les membres allaient effectuer la majeure partie de leur travail, ainsi qu'un `.gitignore` permettant de réduire drastiquement la taille de ce dossier. Beaucoup des fichiers présents par défaut dans le dossier de la template nous sont en réalité dispensables, puisqu'ils sont automatiquement regénérés par l'éditeur Unity lors du chargement de la solution.

Malgré la mise en place des outils nécessaires au développement du projet, le jeu a mis du temps avant de réellement évoluer. En effet, Séraphin et Daniel semblaient déjà avoir perdu beaucoup de leur envie et de leur motivation suite à leurs différents déboires en rapport avec leurs résultats. Le projet a donc mis un certain temps avant de prendre enfin son envol.

La première phase de la réalisation a finalement été entachée par les départs de Séraphin et Daniel. Le départ de Séraphin mi-janvier s'est produit alors que Daniel était absent de l'école depuis déjà deux semaines. Nous avons alors fait part de notre problème pour savoir si un transfert de groupe était possible. Malheureusement, la désinscription de Daniel n'a été effectuée que des semaines plus tard, peu de temps après l'ajout de Rayane le 10 février. Le groupe a donc traversé une phase de doute quant à la poursuite du projet suite à l'absence de Daniel, ne laissant que deux membres réels. Cette période de doute a causé pendant un temps un ralentissement du projet, à cause de la perte de motivation qu'elle a pu entraîner chez les deux membres restants, Victor et Kévin.

La majeure partie de la réalisation a été grandement aidée par la quantité importante de documentation concernant Unity présente en ligne. Elle nous a été d'une aide précieuse pour faire nos premiers pas dans ce moteur de jeu, qui était pour chacun de nous un outil entièrement nouveau à prendre en main.

Cela ne nous a pas empêché d'avoir à faire face à différents problèmes au fil de cette première phase de développement. L'identification de l'origine de ces problèmes a fréquemment pu être facilitée grâce aux forums Unity couvrant déjà de nombreux problèmes récurrents. Malgré la nouveauté de cet outil pour nous, nous avons réussi à nous adapter aux contraintes liées aux différents bugs rencontrés.

Une des premières difficultés à laquelle il a fallu faire face a été l'intégration du multijoueur. Les premières mécaniques conçues uniquement pour le solo dans un premier temps pour faciliter le début du développement se sont heurtées à l'arrivée du réseau et à son lot de problèmes avec. On peut notamment citer les problématiques de synchronisation des sprites des joueurs et de leurs animations pour chacun des joueurs connectés au serveur Photon.

Des problèmes similaires de non-compatibilité ont fréquemment dû être réglés lors des fusions des branches de travail Git de chaque membre. La manipulation du dépôt Git ne s'est d'ailleurs pas fait sans quelques hésitations, avec la peur d'avoir commis une mauvaise manipulation lors des opérations de merge. Malgré tout, nous avons l'impression d'avoir progressé dans la manipulation de ce puissant outil qu'est Git et notre confiance lors de son utilisation commence à s'améliorer.

Durant la semaine précédant la soutenance, la majorité des communications du groupe se sont faites sur Discord. En effet, la distance entre les domiciles de chacun et l'école rendait plus efficace et moins coûteux en temps de travailler à distance. Plusieurs meetings vocaux ont été organisés, ainsi qu'une dernière rencontre en présentiel quelques jours avant la soutenance afin de fixer les derniers points sur lesquels des doutes subsistaient. Cette préparation s'est effectuée sans la partie de Rayane, qui n'avait alors toujours pas été préparée malgré les demandes formulées par le reste du groupe.

9.3 Seconde période

La deuxième phase de développement nous a mis face à des problèmes et bugs encore plus ardu à résoudre que les précédents. Ces bugs ont surtout été rencontrés lors du développement de la partie multijoueur. Des incompatibilités qui jusque là n'avaient pas encore posées problèmes se sont progressivement dévoilées. La vitesse de progression escomptée a donc été en partie ralentie par l'imprévisibilité de ces bugs.

S'il y avait quelque chose à retenir de cette période ce serait surtout cela, qu'il faut savoir s'attendre à affronter l'imprévu plus souvent qu'on ne le pense. Nous pensions pouvoir arriver à nos objectifs en temps et en heure sans embûches, mais le timing ne s'est pas révélé aussi maîtrisé que nous le pensions. Cela devrait nous servir d'enseignement pour la période finale de développement. L'implémentation de certaines fonctionnalités sera probablement réalisé plus en amont que cela a pu être le cas par le passé, et ce afin de gagner du temps de marge de manœuvre en cas de gros bug à régler.

Cela nous fait aussi regretter quelques choix de conception qui ont été avancés dans le cahier des charges. Nous aurions peut-être dû finalement préférer un combat direct dans les maps, nous rapprochant plus de la facette Aventure que de la facette RPG. Ce système nous aurait économisé des changements de scène qui s'avèrent compliqués à gérer en multijoueur lorsqu'il s'agit de synchroniser correctement tous les joueurs connectés au serveur. Mais rien ne sert de s'apitoyer sur cette situation, nous devons maintenant assumer ce choix, avec toutes les contraintes qu'il entraîne. En revanche si ce choix était à refaire, nous y réfléchirions peut-être à deux fois avant de s'engager dans cette direction. C'est donc aussi un des enseignements à tirer de ce projet : nous ne savions pas entièrement vers quoi nous nous dirigions lors de la génèse du projet. Cependant les choses nous apparaissent plus claires désormais grâce à l'expérience acquise sur Unity. Nous sommes capables de faire des choix plus éclairés grâce à une meilleure connaissance des contraintes et difficultés qui les accompagnent.

Des coups de stress sont aussi à mentionner : à quelques jours de la soutenance, la connexion au multijoueur Photon s'est mise à dysfonctionner, sans message d'erreur apparent pour nous guider un minimum dans la résolution de ce problème. Il a donc fallu écumer les pages de forums d'aide en ligne, que ce soient les forums Photon, Unity, ou encore l'éternel classique StackOverflow. Heureusement, une solution a finalement été trouvée, permettant de restaurer les fonctionnalités principales à présenter lors de la seconde soutenance.

Quelques retards sont donc à constater. L'implémentation des mini-énigmes ne s'est pas encore complètement faite. Le retard pris sur la réalisation de fonctionnalités interactives annexes rendait compliqué la création de celles-ci. Le combat en multijoueur souffre également d'un léger retard, le fonctionnement des RPC Photon implémentés aux scripts n'étant pas encore entièrement assuré. Mais en dehors de cela, le reste des objectifs que nous nous étions fixés (que ce soit l'amélioration du système de combat, l'amélioration de l'interface du joueur, l'arrivée de l'audio et des premiers donjons) sont remplis et les autres objectifs sont en voie de finalisation.

Nous sommes d'autre part très satisfaits de l'aspect graphique du jeu, qui correspond bien à ce que nous avions visés. Il respecte nos attentes en ayant une palette de couleurs chatoyantes, tout en restant relativement simple avec ses tiles en 16x16. Le mix entre les deux genres du jeu, RPG et Aventure, évolue au fil des nouvelles fonctionnalités et l'importance accordée à chacun d'eux n'est pas fixe. Cela nécessite parfois des ajustements pour assurer la cohérence de l'ensemble du gameplay, des ajustements qui peuvent donner la priorité à l'un ou à l'autre des deux genres selon les choix conceptionnels faits dans différentes fonctionnalités. Par exemple, tandis que le combat au tour par tour et les items tels que les potions sont des marqueurs du genre RPG, l'aspect simplifié des combats tendrait plus vers le genre Aventure.

Pour continuer sur les points positifs, le sentiment d'accomplissement ressenti une fois un bug corrigé ou une fonctionnalité ajoutée sans problème était très gratifiant. Comme souvent en informatique, des heures sont passées à souffrir sur un problème qui semble trivial au premier abord, mais qui se révèle être une voie sans issue apparente. Heureusement, ses heures de tourmente sont généralement couronnées par un instant de bonheur et d'accomplissement, lorsqu'enfin le code tourne comme escompté. D'une certaine manière, c'est aussi quelque chose qu'est en train de nous apprendre ce projet, plus que les TP n'ont déjà pu nous l'apprendre. On y découvre ce que c'est concrètement de programmer un projet d'envergure plus importante qu'un simple TP hebdomadaire, avec ses joies et ses peines. Tout ne se passera pas comme prévu et il faudra savoir persister, même si la tâche semble insurmontable. Une fois que les choses se passent enfin comme prévu, le sentiment n'en est que plus grisant.

En ce qui concerne l'entente du groupe, celle-ci est en voie d'amélioration. La lecture du premier rapport faisait état d'une absence de travail de la part de Rayane. Cette fois-ci, Rayane a réalisé les tâches qui lui ont été demandées et la communication avec le reste du groupe s'est relativement améliorée par rapport à la période précédente. On ne peut que espérer que les choses continuent dans cette direction et que cela dure.

Il faut cependant noter une fois de plus un rendu de travail tardif du côté de Rayane. Cela a causé plusieurs mésaventures avec la gestion du repo Git à moins de 24h de la soutenance, des mésaventures dont on aurait préféré se passer. En effet, le groupe s'est rendu compte trop tardivement que Rayane ne travaillait pas sur une version clonée du repo liée au Git, mais sur un download local du zip récupéré à partir du lien de téléchargement web GitLab. Son dossier de travail n'était donc pas contenu dans une branche, comme le reste des fonctionnalités développées par Victor et Kévin l'avaient été avant des commandes de merge pour actualiser le main. Le manque de connaissances de Rayane en Git s'est avéré handicapant pour le groupe, tout comme pour lui. De mauvaises manipulations du repo pour intégrer son travail, sans autre copie locale, l'ont obligé à récupérer périlleusement les scripts qu'il avait codé pour les réintroduire manuellement dans la version la plus récente du code. Le manque de copies locales de son travail couplé à une manipulation hasardeuse du repo Git a entraîné un stress à quelques heures de la soutenance qui aurait pu être facilement évité si plus de soin avait été apporté aux méthodes de travail.

Enfin, le groupe va peut-être pour la dernière période connaître l'arrivée d'un quatrième membre. Cette arrivée ne pourrait être que bénéfique pour le rythme de progression du travail. La stagnation qu'a pu connaître le groupe lorsqu'il n'était constitué de plus que deux membres (voir rapport de soutenance 1) ne serait plus qu'un lointain souvenir. Nous espérons que cette arrivée sera bel et bien possible, et que si tel est le cas, les objectifs pourront être remplis avec plus de facilité grâce à une collaboration renforcée.

9.4 Dernière période

C'est finalement probablement la dernière période qui aura été la plus intense. Avec en plus du projet les révisions de partiels à gérer, il faut avouer qu'une bonne partie de la finalisation du développement a eu lieu seulement après la fin des épreuves de partiels.

Comme vu dans la section de réalisation des tâches, le multijoueur est enfin fonctionnel en combat, permettant d'offrir enfin un intérêt au jeu à 2. Il a également fallu particulièrement débugger de nombreux problèmes de fonctionnement liés au multijoueur et à la gestion synchronisée de Photon.

Les dernières maps ont également été rapidement ajoutées afin d'avoir un jeu avec une aventure à la durée respectable. En effet, dans ses versions précédentes, le jeu n'avait qu'un très faible nombre de cartes, ce qui ne permettait pas d'avoir une aventure "complète". Désormais, l'aventure reste bien évidemment courte face à ce qu'offrirait un jeu professionnel, mais elle reste d'une durée respectable.

Il est encore une fois de plus à noter que malheureusement Rayane n'a fourni aucun travail, ce qui est explicitement visible dans la section réalisation des tâches. Sans nouvelle de Rayane pendant un bon moment, une tâche lui avait été confiée. Plusieurs relances plus tard, Rayane n'est revenu vers nous que la veille de la soutenance à 16h (donc à moins de 24h de la soutenance) pour nous demander s'il était possible d'implémenter ses réalisations, alors que nous étions dans une phase de déboggage, de peaufinage des derniers détails et de préparation de la soutenance. Nous avons alors dû refuser, au risque d'introduire des bugs et de compromettre l'intégrité du multijoueur à quelques heures de la soutenance. Rayane ne semblait pas réaliser qu'il fallait proposer un produit fini et qu'introduire de nouvelles fonctionnalités la veille n'était tout simplement pas envisageable.

Le projet s'est donc encore une fois malheureusement déroulé à deux. Les attentes d'origine mentionnées dans le cahier des charges n'ont donc pas pu entièrement être atteintes. Cependant, nous proposons un jeu jouable et relativement stable après plusieurs débuggages.

10 Conclusion

Nous sommes donc fiers d'avoir réussi à proposer un jeu jouable et relativement stable, répondant aux grandes lignes du cahier des charges. En revanche, la manière dont s'est déroulée l'organisation du projet laisse un goût amer aux membres du groupe.

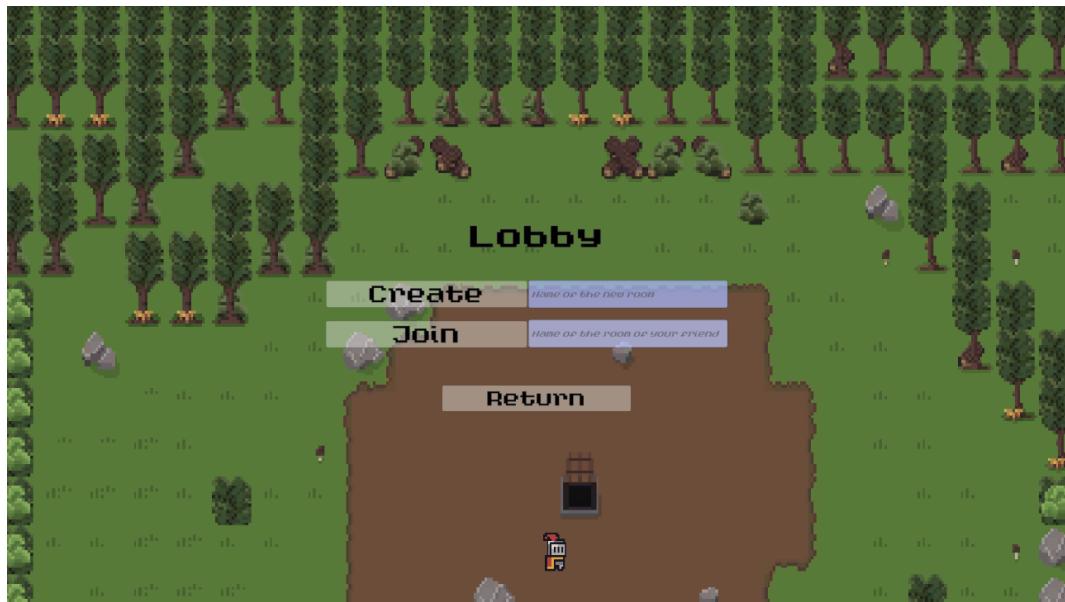
Cette dernière phase de développement, comme mentionné précédemment, a finalement été la plus rude avec la gestion des révisions de partiels durant un temps. La semaine précédant la soutenance a été plus que jamais chargée en travail et surtout en déboggage, avec la gestion du multijoueur Photon donnant beaucoup de fil à retordre.

C'est ce que nous retiendrons probablement de plus dur dans les pré-requis de ce projet : l'implémentation d'un multijoueur. La création d'un jeu solo Unity, maintenant que nous en connaissons les contraintes principales, semble en comparaison bien plus simple, éliminant de nombreuses problématiques de synchronisation. Cela nous a paru quelquefois presque insurmontable en tant qu'étudiants de première année, avec un challenge très élevé dès le départ.

Mais le jeu est là, fonctionnel et jouable et les épreuves que nous avons eu à affronter en tant que développeurs débutants n'ont fait que renforcer grandement nos compétences en informatique. Nous espérons que vous saurez apprécier le jeu et l'ensemble du travail fourni pour la réalisation de celui-ci.

11 Annexes

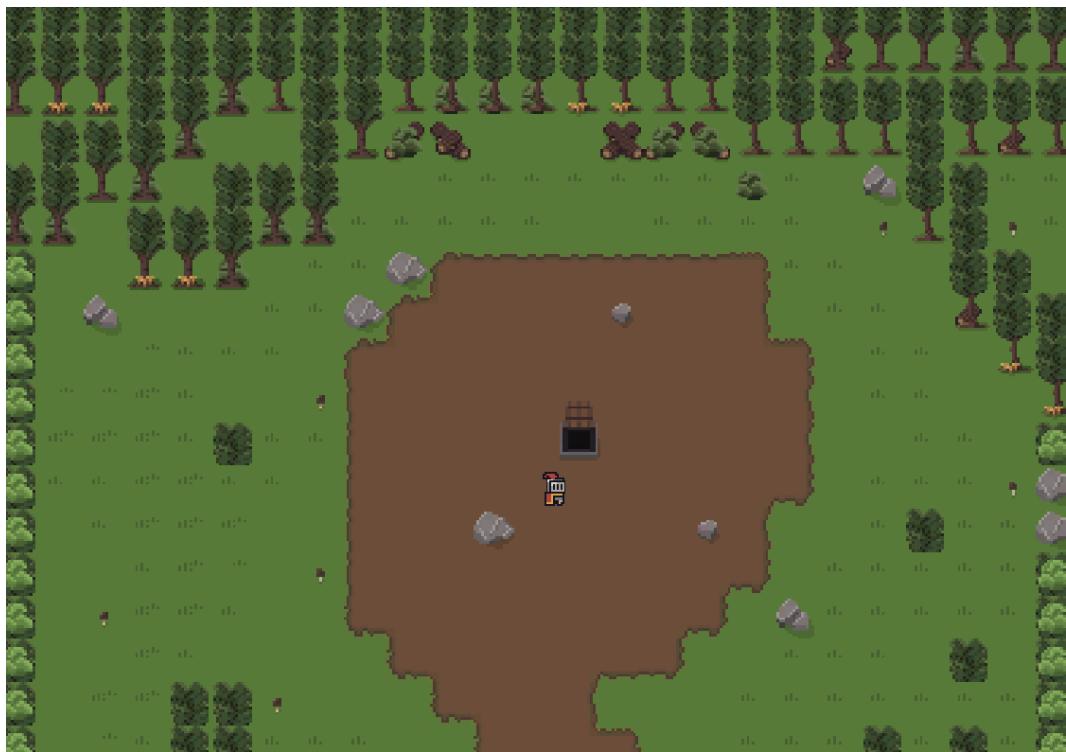
Menus et interface du jeu











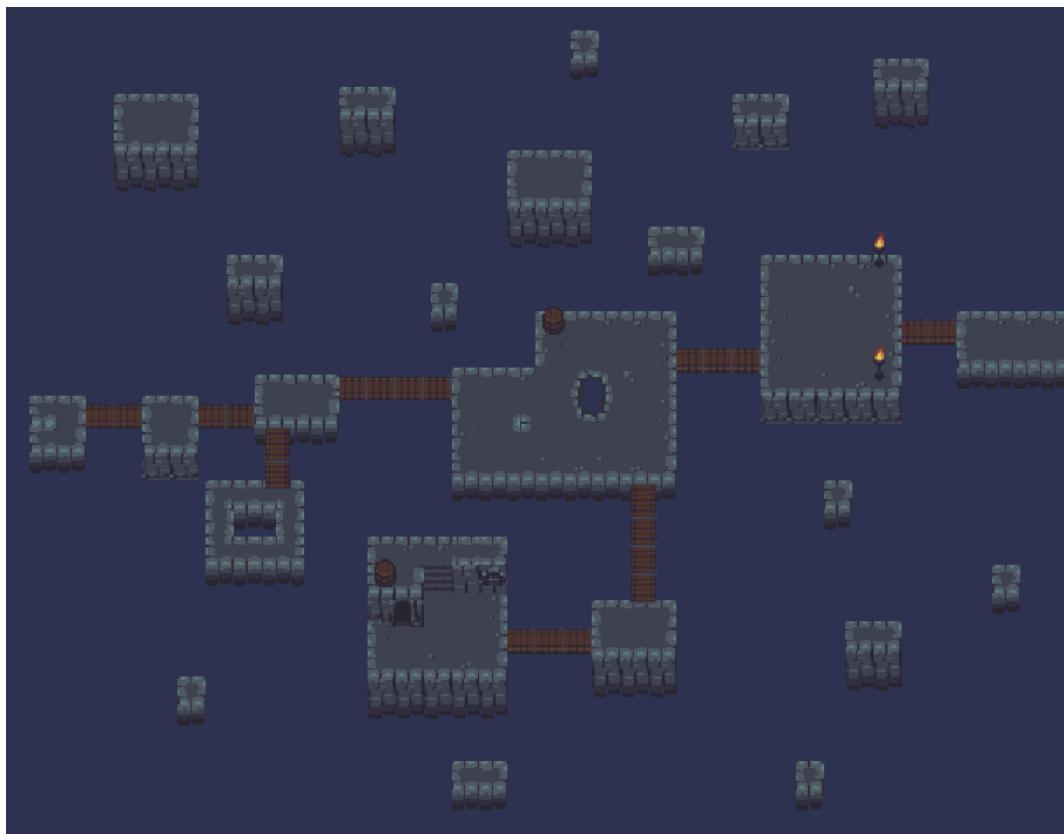
Segment de l'Overworld (nord)



Segments de l'Overworld (centre et sud)



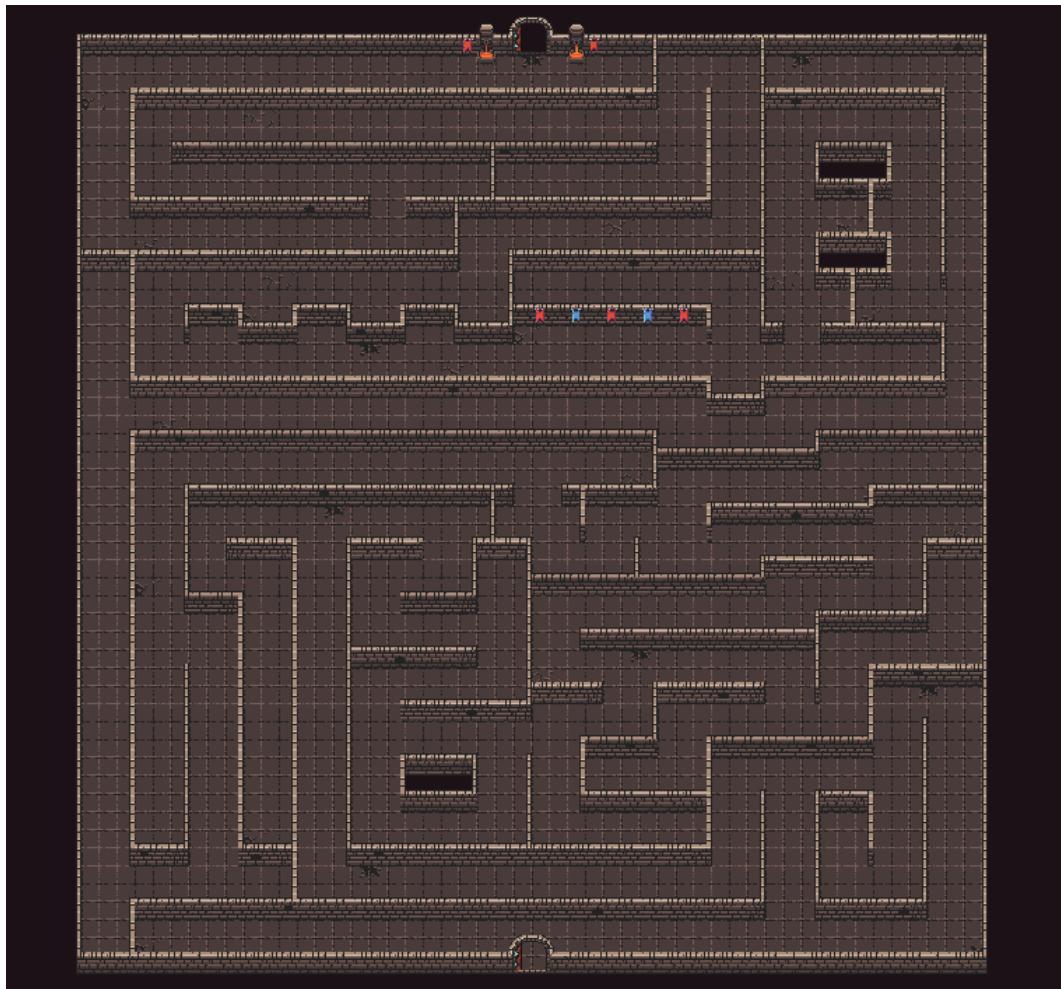
Donjon 1



Donjon 2



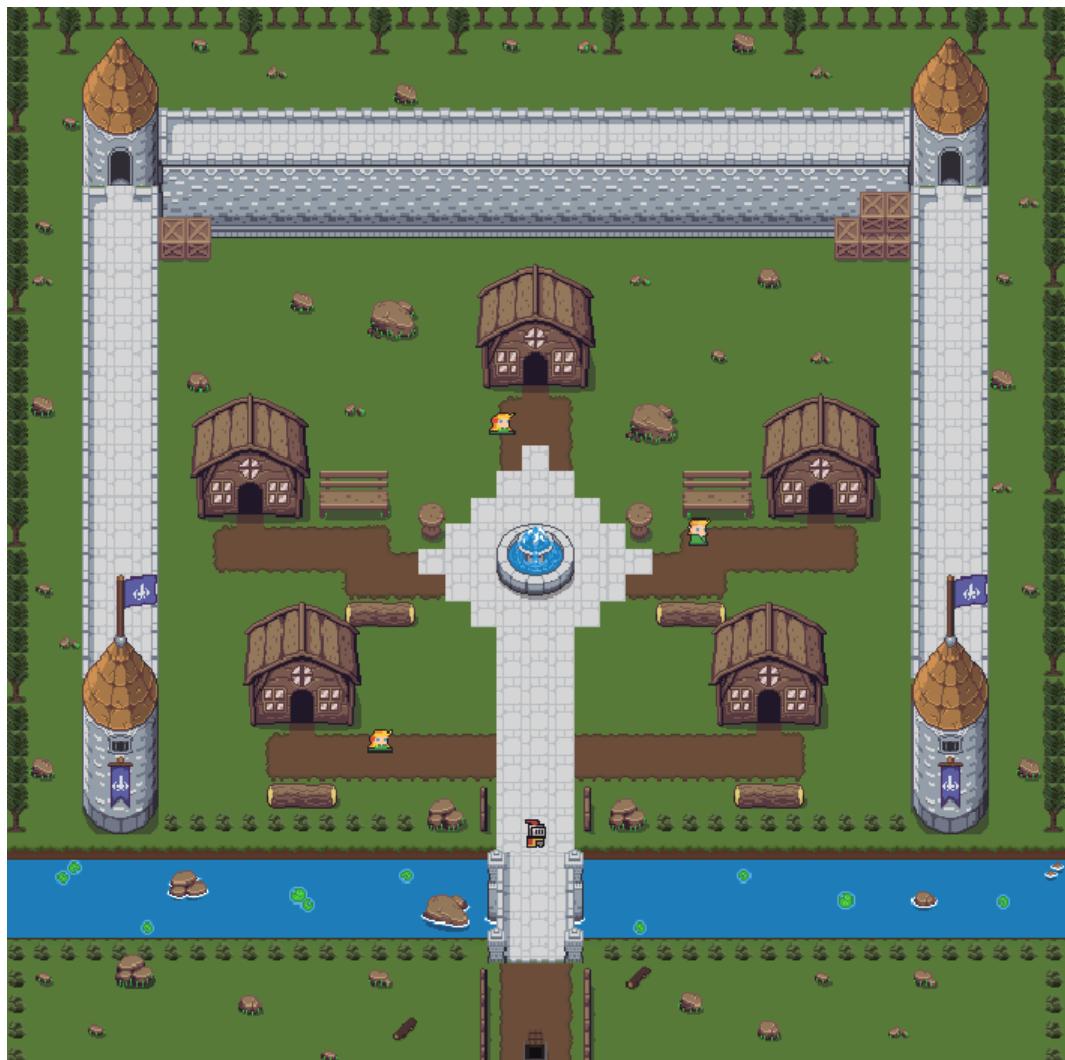
Donjon 3



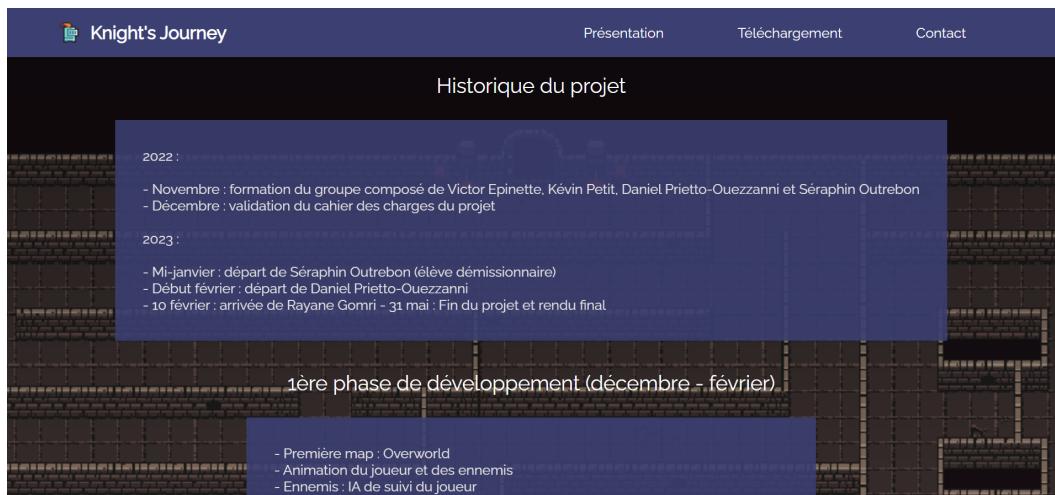
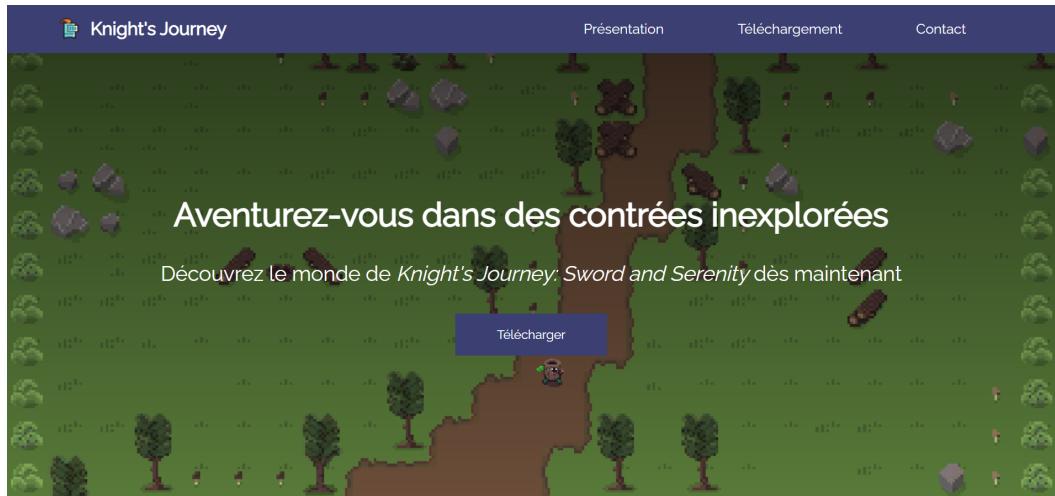
Labyrinthe



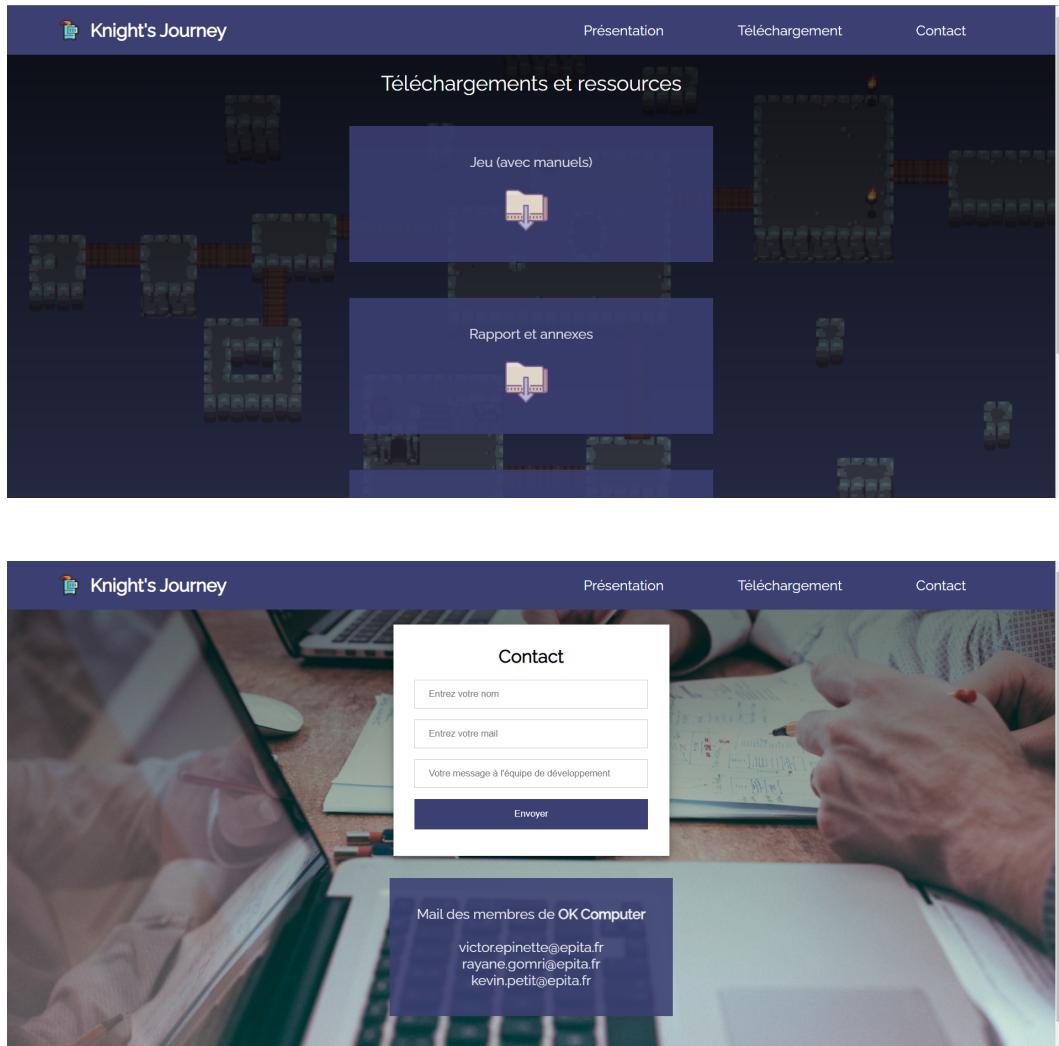
Donjon du boss



Village



Site web du projet (accueil et présentation)



Site web du projet (téléchargements et contact)