



Cahier des Charges

un projet réalisé par :

GAUTIER Alexandre

LORGE Adrien

FERRETTI Leo

GUILBAUD Mathis

supervisé par

BLET Loïc, DERRODE Lamia et DIACONU Christian

Table des matières

1	Introduction	3
1.1	Présentation de l'équipe	3
1.1.1	Nom de l'équipe	3
1.1.2	Projet X	3
1.2	Contenu du cahier des charges	4
2	Présentation du Jeu	5
2.1	Gameplay	8
2.1.1	Système d'ombre/lumière	8
2.1.2	Monstre	8
2.1.3	Boss	8
2.1.4	Economie	9
2.1.5	LeaderBoard	9
2.1.6	Pièges	9
2.2	Multijoueur	10
2.3	Site Internet	11
2.4	Menu et Interface	12
2.4.1	Menu	12
2.4.2	Interface	12
2.5	Graphisme et Sound Design	12
2.5.1	Graphismes :	12
2.5.2	Présentation générale	12
2.5.3	Technique	12
2.5.4	Sound design :	13
2.5.5	Présentation générale	13
2.5.6	Technique	13
2.6	Level Design	13
2.6.1	Présentation générale	13
2.6.2	Technique	13
3	Répartition des tâches et avancement	14
3.1	Répartition des tâches	14

3.2 Avancement 14

4 Budget 15

5 Conclusion 16

Chapitre 1

Introduction

1.1 Présentation de l'équipe

1.1.1 Nom de l'équipe

Le nom de l'équipe : Poule a pour origine l'histoire d'un guerrier Sparte au Ve siècle, et plus précisément lors de la guerre du Péloponnèse. Celui-ci était un soldat puissant, charismatique, musclé et redouté de tous. Sa réputation était telle que personne ne voulait le côtoyer. Sa seule amie était donc une Poule qu'il chérissait plus que tout au monde. Lors de la guerre, il avait emmené sa poule qui se battait à ses côtés. Malheureusement, un soldat Athénien la transperça d'un coup de lance et elle mourut au pied du Spartiate désormais fou de rage. Nous avons donc choisi Poule comme nom d'équipe pour rendre hommage à cet homme qui a tué 10.000 ennemis Athénien pour venger sa poule.

Dans cette équipe, nous nous sommes chacun assigné une tâche principale dans le jeu, disponible dans le Tableau de répartition des tâches.

1.1.2 Projet X

Notre Groupe c'est formé naturellement : nous sommes des amis proches qui aimons les jeux vidéo. Nous avons donc décidé de créer ensemble un Roguelike, un sous-genre de jeu vidéo de rôle dans lequel le joueur explore un donjon infesté de monstres qu'il doit combattre pour gagner de l'expérience et des trésors..

Cela nous est venu car nous aimons tous jouer à ce genre de jeu dont le style nous plaît particulièrement. Mais ce n'est pas le seul critère qui nous a fait choisir ce type de jeu, c'est surtout la liberté de scénario, de gameplay, d'environnement, de telle sorte qu'il y ai autant de plaisir pour produire le jeu que pour y jouer.

Via ce projet, nous allons apprendre à travailler en équipe sur le long terme tout en utilisant de nouveaux outils : Unity, logiciel mixage son, logiciel montage ... Nous allons également approfondir nos connaissances en C# et utiliser ces connaissances pour créer quelque chose de concret.

1.2 Contenu du cahier des charges

Nous allons donc créer un roguelike en prenant soin de développer chaque facette du jeu :

- Site Internet lié au jeu
- Multijoueur
- Gameplay
- Level Design
- Menus et HUD
- Graphisme
- Sound design

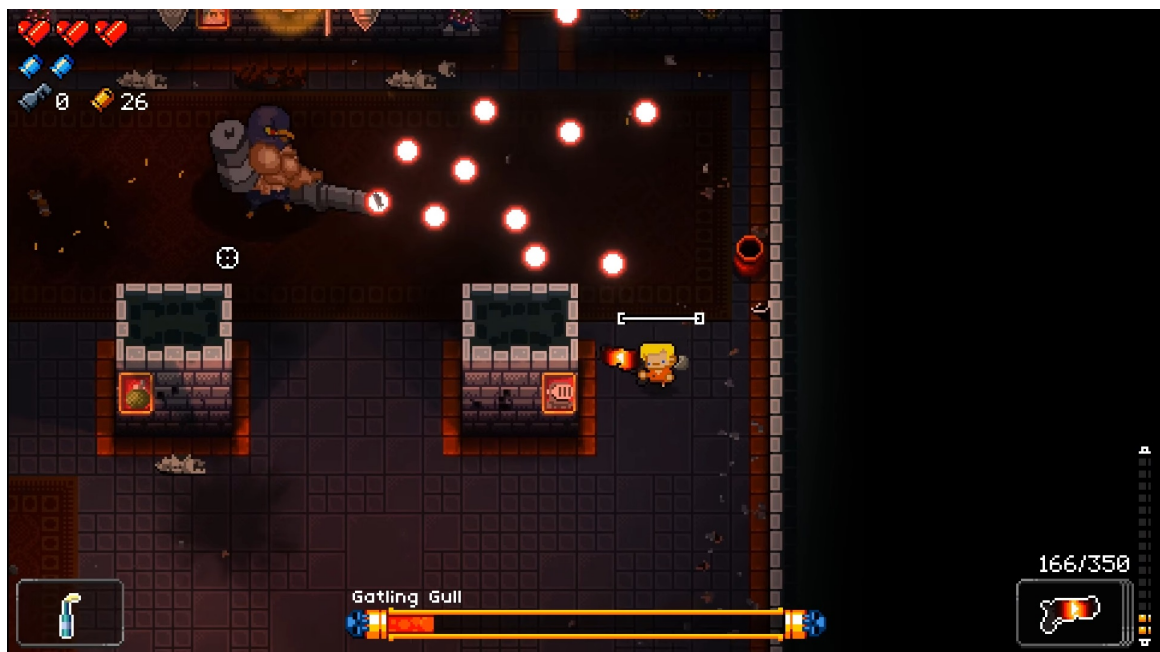
Ce cahier des charges est dédié aux explications point par point des éléments qui composent le jeu ainsi que la manière dont ils seront implémentés.

Chapitre 2

Présentation du Jeu

Notre jeu sera un roguelike dans lequel un personnage va évoluer dans un univers sombre : un donjon. Il sera mené à éliminer diverses créatures grâce à de nombreuses armes. Il devra également affronter des boss qui jalonneront son parcours . Le personnage possèdera ses propres caractéristiques comme des points de vie et de l'expérience . Ce jeu sera inspiré d'autres jeu du même genre :

- Enter the Gungeon, également un roguelike, très dynamique avec de très beau graphismes et dans lequel on retrouve de nombreuses interactions avec les objets du décors. On peut par exemple retourner des tables pour les utiliser pour sa defendre, ses interaction avec les objets sont une source d'inspiration :



- Binding of Isaac, qui a une atmosphère très sombre digne d'un vrai jeu de donjon. Nous cherchons dans notre jeu à créer une atmosphère similaire, angoissante et qui tiens en haleine tout en gardant une simplicité dans les graphismes. :



- Heroe siège, qui possède plusieurs difficultés, ce qui le rend intéressant pour toute personne souhaitant du challenge. De plus le point de vu de la camera est très intéressant. Celle-ci se tient en plongée, de manière à apercevoir à la fois le haut et la face des bâtiments. Ce point de vu rend le jeu visuellement plus dynamique. Nous allons reprendre cette idée là. :



Ce jeu sera similaire à un jeu de borne d'arcade. Cela implique un jeu dynamique, assez dur possédant un leaderboard et dans lequel toute mort est définitive. Ainsi, à chaque nouvelle partie, tout sera réinitialisé. De plus, notre but est tout d'abord de créer un jeu de base, un squelette auquel nous pourrons ajouter très facilement du contenu pour le rendre plus "riche", comme de nouvelles armes, ennemies, boss ou encore objets.

2.1 Gameplay

Le joueur incarne un personnage qui possède ses propres caractéristiques : attaque, vitesse, vie ... Ces caractéristiques varieront en fonction de l'équipement, des bonus sur ses statistiques achetés, mais également des attaques des différents monstres qui pourront par exemple ralentir le personnage ou encore l'empoisonner.

Les points de vie seront implémentés sous forme de barres de vie en haut de l'écran. Les points de vie seront actualisés à chaque fois que le joueur se fera toucher par un monstre ou un piège. Si les points de vie tombent à zéro, alors la partie s'arrête.

En multijoueur, si les deux joueurs meurent, alors la partie s'arrête, si l'un des deux meurt, alors il a toujours une possibilité de faire ressusciter son partenaire si il arrive à finir le niveau, le joueur qui se fait ressusciter recevra des malus.

2.1.1 Système d'ombre/lumière

Nous prévoyons également de mettre en place un système inspiré de *Darkest Dungeon* qui permettrait de réguler la difficulté et qui affecterait les loots et la puissance des bonus : Le joueur possède et ramasse des torches qui lui servent à éclairer le donjon et augmenter le niveau de luminosité. Plus le niveau de luminosité est élevé, moins les monstres sont puissants mais ils donneront par conséquent des objets moins puissants/rare. Au contraire, plus la luminosité est faible, plus les monstres seront puissants et plus les objets obtenus en les tuant seront rares. Ce jeu de lumière rajoute une mécanique intéressante au jeu qui accentue également l'aspect angoissant et sombre du jeu. Bien entendu lorsque qu'aucune torche n'est utilisée l'écran ne sera pas tout noir, les couleurs seront juste ternies (nuances de gris, de sombre et de clair) de façon à donner cette impression de sombre. Pour les effets de lumière, nous utiliserons les outils fournis par Unity.

Le joueur peut régler sa difficulté tout au long de sa session de jeu, les caractéristiques des monstres seront proportionnelles avec la luminosité du joueur. Il est impossible de modifier la barre de luminosité quand un joueur a engagé le combat.

En multijoueur : Chacun aura sa propre barre de luminosité et la difficulté choisie sera la moyenne des deux valeurs pour éviter des conflits avec des niveaux de difficulté différents.

2.1.2 Monstre

Les monstres essaient de tuer le joueur au corps à corps ou à distance, chaque monstre aura ses caractéristiques et ses attaques, ses caractéristiques seront boostées selon la luminosité du joueur.

2.1.3 Boss

À la fin du donjon, le joueur devra affronter un boss, les boss auront des plusieurs attaques, qui seront tirées aléatoirement.

2.1.4 Economie

Présentation Générale

Il y aura un système d'économie dans le jeu qui prend une place plus qu'importante dans le gameplay du Jeu. L'argent s'obtient en éliminant des monstres, en finissant les niveaux et en vendant des objets. Cet Argent pourra être utilisé dans une boutique présente dans les niveaux afin d'améliorer le personnage, ses statistiques, ses éventuelles compétences. Il y aura également possibilité d'acheter des objets utilisables comme par exemple des potions de vie ou d'autres qui permettraient d'améliorer temporairement les statistiques du joueur ou encore de récupérer la vie perdue. Encore d'autres objets pourront être achetés comme des équipements (différentes armes, armures) ou des torches, utile pour les mécaniques de lumière que nous allons mettre en place. Les objets pourront également être vendu.

Technique

Tous les objets appartiendront à la classe objets, qui définit leurs propriétés communes. Ensuite, une classe héritant de la classe objet sera créé pour chaque grand type d'objet (équipement, consommable ...). Puis enfin, chaque objet hérite de la classe lié à son type et possède ses propres caractéristique.

Une boutique n'est en fait qu'une liste d'objets que le joueur peut acheter si la variable "argent" qui lui est associé est supérieur à l'argent nécessaire pour acheter l'objet. En Outre, le joueur pourra aussi vendre des objets et gagner de l'argent.

2.1.5 LeaderBoard

Le jeu aura un tableau de scores qui sauvegardera les meilleurs score sur dans le jeu.

Le score sera calculé en fonction de la difficulté choisi par le système de luminosité, du nombres de monstre tués, temps passé pour finir la partie.

Les scores seront affiché dans une liste trié de taille fixe.

2.1.6 Pièges

Les niveaux que le personnages aura a parcourir seront piégés, il y aura différents type de pièges, des pièges mortels, des piège infligeant des blessures, des pièges qui placerons le personnage dans un état handicapant (par exemple empoisonnement ou réduction de la vitesse, etc).

Pour les pièges mortel, nous créeront un box collider avec l'option de détection "is triggered" sur Unity qui lorsque le personnage sera détecté déclenchera l'animation spécifique au piège (par exemple, la fermeture des "mâchoires" d'un piège a loup) puis l'animation de mort du personnage. Pour les autres pièges, il n'y aura que l'animation du piège et les effet associé, comme de l'empoisonnement, un saignement ou un ralentissement, seront appliqués au joueur qui est entré dans le collider.

2.2 Multijoueur

Le jeux possèdera 2 modes : un mode où le joueur joue seul et un mode multijoueur local, ou les joueurs jouent côte à côte sur la même machine. Ce jeu disposera d'un systeme de point. Selon le nombre d'ennemies tués, de boss tués, d'objets apportant des points ou encore si le joueur atteint la fin du jeu, des points seront attribué. L'implémentation d'un leaderboard est nécessaire. Il y aura donc 2 classements distincts : Le leaderboard local et le leaderboard pour le multijoueur. Ils apparaîtront à la fin de la partie une fois que celle-ci est terminées et dans une section dédié dans le menu d'entrée du jeu.

Nous utiliserons pour ce faire le système de multijoueur déjà implémenté dans unity et un autre système qui lui est associé (Hight level API), Ces systèmes sont fiable et permettent de faire tous ce que l'on attend d'un jeu en multijoueur comme celui que nous allons développer.

2.3 Site Internet

Nous mettrons ici le lien du site associé à notre projet lorsque celui ci sera finis et hébergé (Il sera sûrement hébergé par OVH qui permet un bon hébergement à des prix accessibles). Il y aura sur ce site toutes les informations nécessaires à l'installation du jeu, l'exécutable du jeu (pouvant être téléchargé) ainsi que de nombreuses informations sur l'avancement du projet et sur le projet en lui-même.

ce site web sera développé à la main avec du HTML du CSS, quelque script Javascript et du JQuery pour rendre le tout plus joli. Ce site s'inspirera d'un site de promotion d'un jeu tel que celui de Sea of Thieves Sea of Thieves (bien entendu nos compétences en HTML et CSS restant limitée il n'y auras malheureusement pas autant de technique que dans ce modèle) Cependant nous implémenterons Tout de même un site web vendeur qui donneras envie de jouer au jeu.

Nous allons également y insérer des vidéos de gameplay et une présentation de notre groupe.
Site de promotion de Sea of Thieves : <https://www.seaofthieves.com/fr/buy>

2.4 Menu et Interface

2.4.1 Menu

Le seul menu disponible sera celui d'entrée dans le jeu qui présentera plusieurs options : Solo, Multijoueur, Leaderboard. Ce dernier renverra de nouveau sur un menu avec les options Leaderboard Solo ou Leaderboard multijoueur.

2.4.2 Interface

Le HUD/Interface de notre jeu donnera les éléments essentiels sur notre personnage durant la partie. Il montrera la vie du personnage, ses compétences, l'argent qu'il possède et son expérience

2.5 Graphisme et Sound Design

2.5.1 Graphismes :

2.5.2 Présentation générale

Pour ce qui est des graphismes nous allons opter pour un style retro/pixelisé qui convient très bien aux roguelike ainsi qu'aux jeux d'arcade. L'atmosphère du jeu sera sombre. Parmi les pièces, on retrouvera des prisons, des salles de torture, des galeries souterraines, des lieux délabrés et abandonnés (manoir). La luminosité, bien que basse de base sera changée par l'utilisation des torches. La présence de sang nous obligera à mettre un disclaimer au début du jeu. Le site Piskel couplé aux talents d'Adrien seront utilisés pour réaliser les modèles de personnages, block et objets.

2.5.3 Technique

Il y a plusieurs types de Graphismes :

- Le Sol et les Murs : Pour représenter le sol et les murs, des blocks carrés sont dessinés. Ce sont des Tiles qui seront assemblés de manière aléatoire par type lors de la génération du monde. Par exemple, un mur sera composé de Tiles de Type "Mur" qui possèdent de petites variations entre elles et choisissent aléatoirement pour rendre le jeu visuellement plus diversifié.
- Les objets équipables : Les équipements permettant d'augmenter les statistiques du personnage, mis en surbrillance et dans un layer ou "couche" dans unity permettant sa visibilité.
- Les objets du décor : sobre, il ne seront pas mis en avant
- Les entités : ils possèdent des animations, une suite de dessin qui joue un mouvement. Cette animation sera jouée lors d'une action particulière, par exemple lorsque le personnage est touché, l'animation de dégât subit est jouée.

2.5.4 Sound design :

2.5.5 Présentation générale

Quant au sound design, l'utilisation de mélodies mélancoliques, mystérieuses et angoissantes sera nécessaire pour être en accord avec l'univers du jeu. Cependant, des musiques plus "énergiques" ponctueront les moments d'action comme les salles de boss. Il y aura d'ailleurs une musique propre à chaque boss. Ces musiques seront réalisées à partir d'un logiciel de mixage son comme Ableton. Pour ce qui est des sons représentant les ouvertures de coffres, dégâts subis par le joueur, ou encore lors du ramassage d'une arme ou l'usage de différentes compétences, des sons téléchargés sur internet seront utilisés. Il y aura également de la musique générée par certains objets (comme par exemple des petites radios ou des platines vinyle).

2.5.6 Technique

Tous les sons et musiques seront regroupés dans un dossier sous notre projet Unity. Chaque niveau possèdera un attribut tableau/array "musique" et ces musiques seront jouées les unes après les autres de manière aléatoire dès que le joueur entre dans le niveau en utilisant la fonction Random qui prendra un indice aléatoire dans le tableau. Quant aux sons, ils seront liés aux actions des entités. Par exemple, lorsque le personnage se fera frapper, un son dédié aux dégâts subis sera joué en plus de la mélodie de fond.

2.6 Level Design

2.6.1 Présentation générale

Comme dans de nombreux jeux de donjon, l'aléatoire est important : Disposition des ennemis, générations des salles, des pièges. Cet aléatoire permet de casser la routine que l'on pourrait avoir en recommençant de nombreuses fois le jeu comme on le fait souvent dans les jeux de type roguelike. Tous les niveaux de ce jeu seront donc générés aléatoirement, à l'exception des salles de boss qui leur seront spécifiques.

2.6.2 Technique

D'un point de vue plus technique, le niveau sera constitué d'un grand espace carré divisé en plusieurs sous-espaces carrés, des sous-salles. Les sous-salles seront placées aléatoirement dans le grand espace. De même, dans les sous-salles seront disposés aléatoirement des objets et les ennemis. Pour réaliser cet aléatoire, les objets seront placés dans des tableaux(array) et y seront sélectionnés aléatoirement grâce à la fonction Random de C#.

Chapitre 3

Répartition des tâches et avancement

3.1 Répartition des tâches

Chaque membre de l'équipe est responsable d'une tâche et est supporté par un suppléant.

	Alexandre	Mathis	Adrien	Léo
Site Internet		RESPONSABLE		Suppléant
Multijoueur		RESPONSABLE	Suppléant	
Gameplay			Suppléant	RESPONSABLE
Level Design	RESPONSABLE			Suppléant
Menus et HUD		Suppléant	RESPONSABLE	
Graphisme	Suppléant		RESPONSABLE	
Sound design	RESPONSABLE	Suppléant		

Par exemple, pour la création d'un bloc "eau", le bloc sera dessiné par Adrien qui l'enverra à Alexandre. Alexandre placera ce bloc dans le niveau et gèrera le son produit lorsque le personnage passera sur ce bloc, Léo se chargera d'implémenter les effets de l'eau sur le personnage mouillé, comme un ralentissement. Adrien pourra rajouter un effet "mouillé" sur le personnage qui apparaîtra dans le HUD et Mathis vérifiera le bon fonctionnement du bloc dans le mode multijoueur.

3.2 Avancement

	Première soutenance	Deuxième soutenance	Troisième soutenance
Site Internet	40%	90%	100%
Multijoueur	30%	80%	100%
Gameplay	50%	90%	100%
Level Design	50%	90%	100%
Menus et HUD	40%	90%	100%
Graphisme	15%	60%	100%
Sound design	15%	70%	100%

Chapitre 4

Budget

Le budget dépendra énormément des attendus sur ce projet, il va sûrement falloir compter l'hébergement du site web, quelques assets et animations, cependant cela restera en dernier recours car nous essaierons de faire nos sprites nous même.

	Prix
Raspberry Pi	37
Ecran	60
Bois	30
Bouton/Joystick	69
Total	196

Prix de la borne d'arcade :

Chapitre 5

Conclusion

Nous compléterons la conclusion une fois le projet terminé, ainsi nous pourrons exprimer dans cette conclusion ce que nous aurons réussi ou non, Ce qui nous a posé problème, comment nous avons procédé pour travailler ainsi que d'autres informations (il sera également possible de trouver ces informations sur le site web du projet).