

The balance of the elements

Livable 4 – Premier jouable publiable

Équipe 5

Antoine Beaulieu Lessard - 111 124 645

Marc-André Brunelle-Langevin -111 120 022

Catherine Robitaille - 111 132 001

Réalisé dans le cadre du cours
IFT-2103

Présenté à
Monsieur François Chéné

18 décembre 2017
Université Laval

Catherine Robitaille

Table des matières

Boucle de jeu.....	4
Mécaniques prévues	5
Ressources de jeu.....	9
Plateforme technologique.....	11
Architecture logicielle.....	11
Présentation de l'équipe	13
Antoine Beaulieu Lessard	13
Catherine Robitaille.....	13
Marc-André Brunelle-Langevin.....	14
Post-Mortem.....	15
Annexe 1 : Analyse technologique.....	16
Premier défi : Son avec audacity et musiques.....	16
Deuxième défi : modélisation du héros	17
Troisième défi : GUI personnalisé	18
Annexe 2 : Rapport d'avancement.....	19
Annexe 2 : Rapport d'avancement premier jouable	20
Annexe 2 : Rapport d'avancement jouable publiable	21
Annexe 3 : Liste des fonctionnalités	22

The balance of the elements

RPG axé sur la résolution de puzzles dans un environnement médiéval et fantastique

Mise en contexte

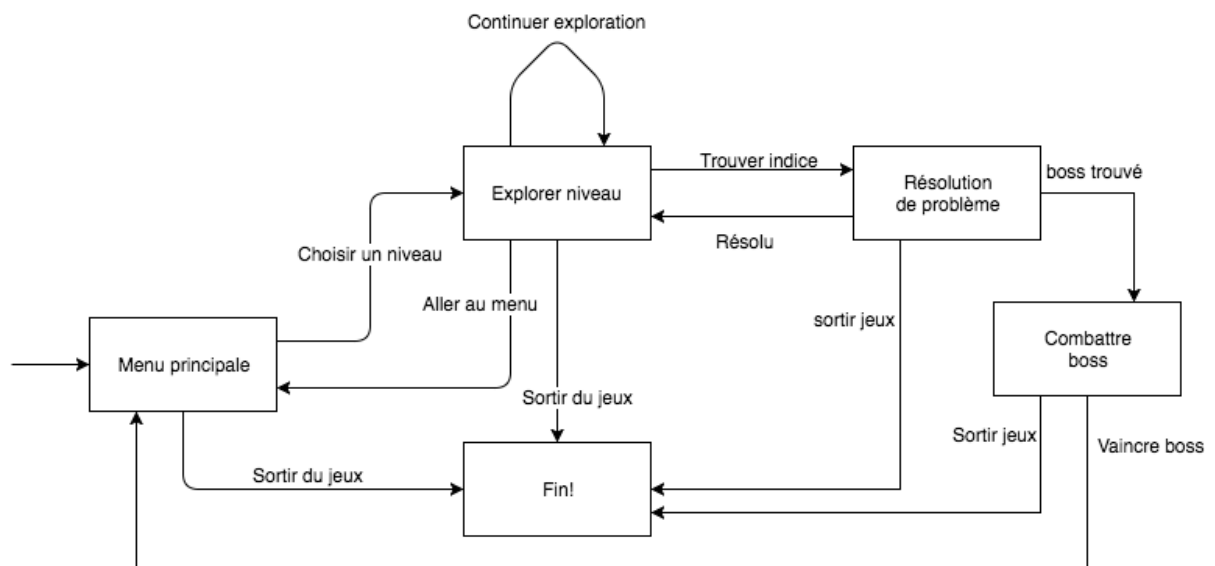
Ulric est un jeune élémentaliste à en devenir dans un univers fantastique. Alors qu'il développe ses talents, il se voit acquies une mission, celle de libérer les quatres temples élémentaires qui ont été attaqués par une force obscure. Ulric doit purger les temples de cet ennemi et rétablir l'équilibre. Ulric cherche à reprendre le contrôle des éléments, il devra parcourir tous les temples pour retrouver la sérénité et, de ce fait, ses pouvoirs d'élémentaliste.

Expérience de jeu

Pour notre projet, nous avons décidé d'élaborer un jeu en mélangeant aventure, stratégie et fantastique. Le personnage principal qui sera incarné, Ulric, est un élémentaliste qui a la mission de protéger les temples des quatre éléments: air, terre, eau et feu. Comme décrit plus tôt, le but de « The balance of the elements » est d'offrir à l'utilisateur une courte expérience de la vie d'élémentaliste. Pour cela, il devra explorer le temple de l'eau, de l'air, de la terre et du feu et combattre le "boss" de chaque temple. Le joueur pourra commencer par l'élément qui lui plait et explorer les temples dans l'ordre qu'il préfère. Cependant, comme les éléments sont bouleversés, il se peut qu'il ait besoin de parcourir plus d'une fois un temple et d'obtenir des indices des autres lieux pour arriver à ses fins.

L'accomplissement d'un niveau se fera à l'aide d'un petit jeu de stratégie ou il faut trouver les éléments manquants pour aboutir à la fin. Si l'avancement lorsqu'on sort du temple n'est pas conservé, l'inventaire et les modifications faites à l'intérieur seront conservées. Le but du joueur est de trouver et placer les leviers et les objets aux bons endroits. Il faudra également faire usage d'agilité pour se déplacer à certains endroits et bien sur de vitesse, car il y aura bien quelques adversaires à battre. Une fois tous les objets placés correctement dans un des temples et l'ennemi vaincu, votre personnage acquiert un pouvoir relié à l'élément. Les pouvoirs sont : la lévitation pour l'air, le soin pour l'eau, l'empêchement des ennemis pour la terre et enfin une attaque de feu pour le dernier élément.

Boucle de jeu



Mécaniques prévues

- Gain d'habiletés
 - Lévitiation : le personnage pourra ignorer la gravité et se déplacer en volant
 - Brûlure : le personnage pourra brûler les ennemies
 - Soin : le personnage pourra restaurer sa barre de vie et les brûlures
 - Empêchement : le personnage pourra bloquer les déplacements des monstres
- Système de combat
 - Animation d'attaque et soustraction de points de vie
 - Résistance/faiblesse aux éléments
 - Attaque au corps à corps ou des incantations
- Musique réactive
 - Musique douce durant le jeu et musique rapide durant les combats
 - Bruits ambiants d'objets à proximité
 - Bruit de combat au contact
- Interaction avec le monde
 - Communication avec des PNJ
 - Ramasser des objets dans l'environnement
 - Interagir avec des modèles du jeu.
- Inventaire
- Vie
 - Dégâts par ennemies
 - Restaurer les points de vie avec des objets et des incantations
 - Le joueur doit recommencer le niveau s'il perd toute sa vie
- Changement de scènes
 - Changement de niveau à la guise du joueur
 - Interaction entre les différents mondes pour l'avancement du jeu

Gain d'Habiletés

Le joueur commence le jeu avec seulement ses coups au corps à corps comme source de dégâts. Il développera des sorts supplémentaires au fil du jeu afin d'augmenter ses capacités de combats. De plus, ces sorts permettront aussi d'avancer dans le jeu en mélangeant incantations et éléments du jeu.

Le système de combat

Le joueur rencontrera des ennemis durant son aventure. Les monstres faibles (trash mobs) seront plutôt simples, possédant une attaque au corps à corps ou un sort. Les boss posséderont quant à eux une gamme plus élaborée d'actions mélangeant sorts et attaques. Le combat peut se jouer de deux manières avec le personnage. Il peut soit attaquer librement, c'est-à-dire, sans cible, et lancer ses sorts devant lui. Il peut aussi cibler un ennemi, le joueur regardera toujours en direction de l'ennemi et les attaques/sorts seront toujours dans sa direction.

Interaction avec le monde

Afin de terminer les niveaux, le joueur devra résoudre certaines intrigues. Pour cela, il devra interagir avec les éléments présents dans les niveaux. Il devra trouver des leviers pour

ouvrir des portes et réussir à atteindre certains endroits avec des sauts. De plus, des interactions avec des PNJ seront nécessaires pour terminer les niveaux. Tuer un ennemi fera apparaître des points de vies qui pourront être ramassés par le joueur. Ceux-ci seront automatiquement ajoutés à la barre de vie si elle n'est pas pleine.

Système de points de vies

Une barre de vie sera toujours présente sur l'écran du joueur. Chaque dégât (ennemies, chutes, brûlures) fera descendre le nombre de points de vies. Pour restaurer sa vitalité, le joueur pourra tenter de ramasser des points de vies sur les ennemis ou utiliser son sort de soin s'il le possède. De plus, changer de monde permettra au joueur de remplir sa barre de vie à sa capacité maximale.

Changement de scènes

Afin de terminer le jeu, le joueur devra réussir tous les niveaux. Par contre, ces derniers sont dépendants entre eux. Par exemple, si le joueur commence par le niveau de la terre, il devra quitter en chemin pour aller dans le niveau de l'eau pour aller chercher un sort qui lui permettra de terminer le niveau de la terre.

Le joueur aura toujours la possibilité de changer de niveau par le menu principal ou par le menu pause. Ce choix provoque une réinitialisation du niveau.

L'univers du jeu

La création du monde sera faite à la main. Chacun possèdera son style et son ambiance, selon l'élément représenté. Les niveaux seront surtout linéaires, mais il est toujours possible de revenir en arrière.

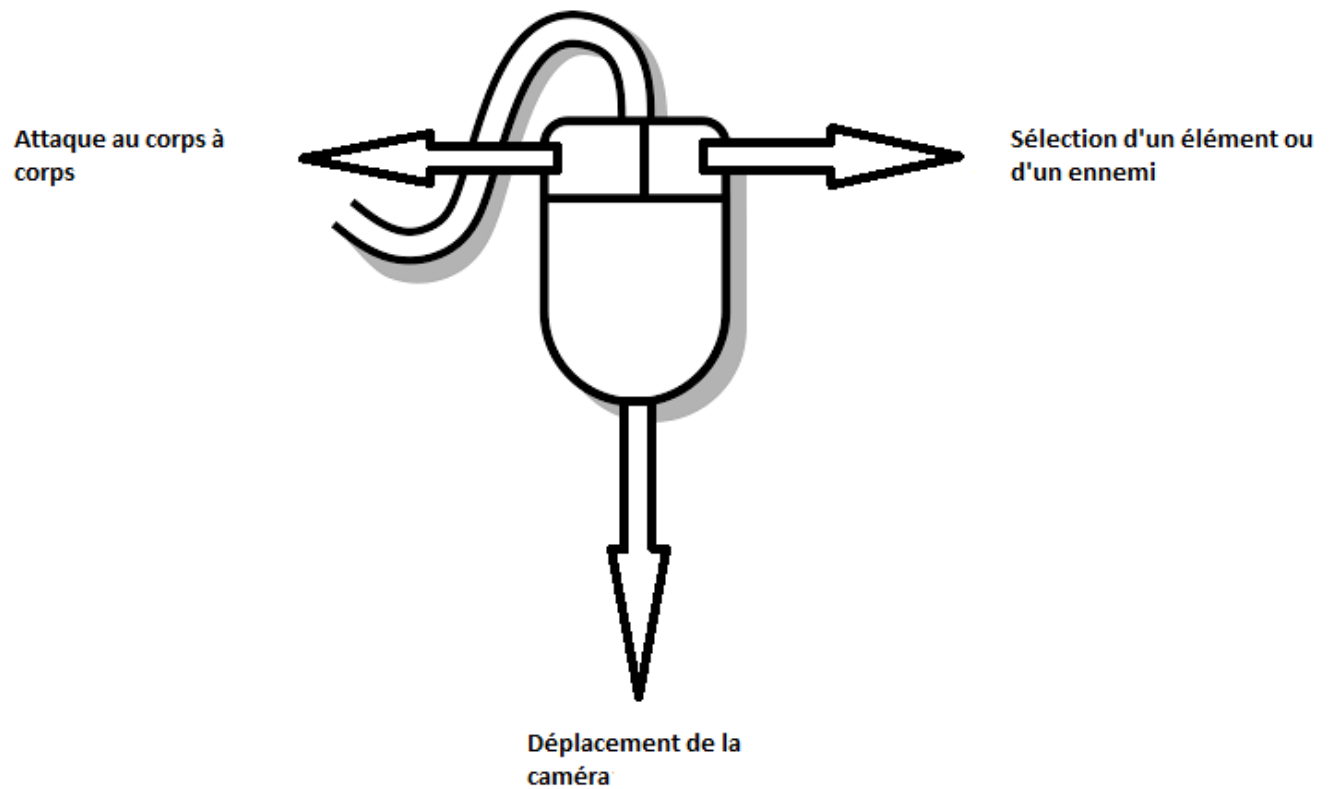
Gestion de la caméra

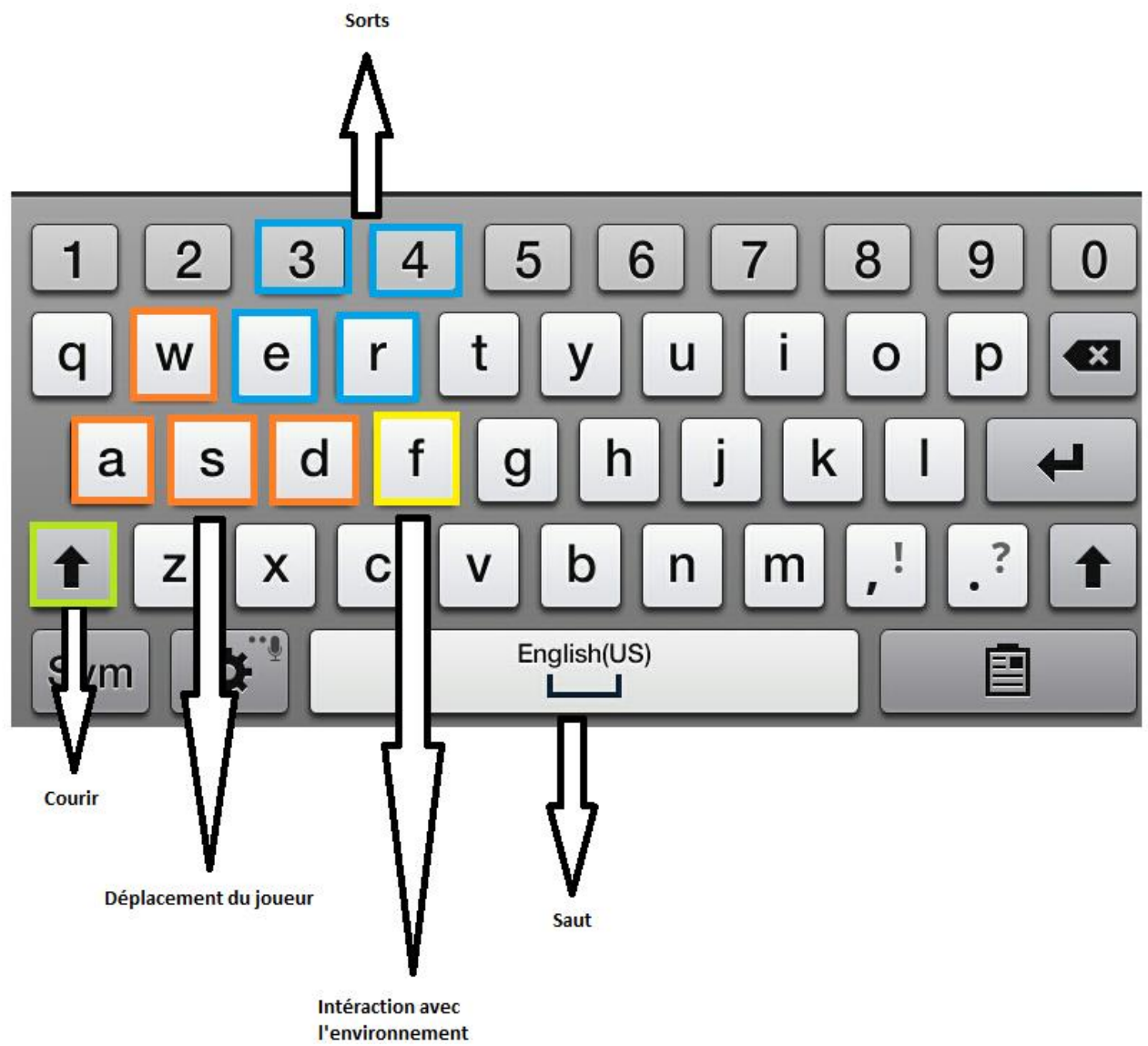
Le système de caméra possède deux états, soit lorsque le joueur reste sur place ou lorsque celui-ci se déplace. Si le joueur reste sur place, la caméra peut se promener autour du joueur librement. Cela donne l'opportunité de bien voir les éléments présents autour du joueur ou de regarder le modèle du héros. Si le joueur se déplace, alors la caméra tentera toujours de revenir derrière le joueur et de pointer dans la direction du déplacement. Le déplacement de la caméra par le joueur est plus puissant que le déplacement automatique, il est donc possible de tourner la caméra, mais si l'on arrête le déplacement, elle reviendra constamment en arrière.

Contrôles du héros

Les contrôles seront exclusivement au clavier. La main gauche permettra de se déplacer, d'utiliser ses sorts ainsi que pour interagir avec l'environnement.

La main droite quant à elle permet de déplacer la caméra, utiliser les attaques au corps à corps ainsi que pour sélectionner des ennemis





Ressources de jeu

- Héro
 - 1 modèle texturé/riggué
 - Animations
 - Marche
 - Course
 - Saut
 - Retombe (après saut)
 - Attaque
 - Dommage
 - Idle (x 2-3)
 - Ramasser un item
 - Mort
 - Sons
 - Marche
 - Course
 - Attaque (x 2-3)
 - Dommage (x 2-3)
 - Saut
 - Retombe (après saut)
 - Ramasse item
 - Mort
- Par monstre/boss (4 monstres et 4 boss)
 - 1 modèle texturé rigger
 - animations
 - Déplacement
 - Attaque
 - Dommage
 - Mort
 - Sons
 - Déplacement
 - Attaque (x 2-3)
 - Dommage (x2-3)
 - Mort
 - IA
- Niveau de l'eau
 - Rocher
 - Texture + Modèle
 - Eau
 - Effet
 - Son proximité
 - Texture sol
 - Texture murs
 - Musique exploration
 - Musique combat
 - Son ambiant (grotte)
- Niveau de terre

- Arbres
- Lianes
- Texture sol
- Texture mur
- Musique exploration
- Musique combat)
- Son ambiant (vent dans la forêt)
- Niveau de l'air
 - Texture sol
 - Texture mur
 - Musique combat
 - Musique exploration
 - Son ambiant (vent)
- Niveau du feu
 - Lave
 - Effet
 - Son proximité
 - Rocher
 - Feu
 - Effet
 - Son proximité
 - Volcan
 - Ciel apocalyptique (skybox)
 - Texture sol
 - Texture mur
 - Musique exploration
 - Musique combat
 - Son ambiant (volcan au loin)
- Menus
 - Image de couverture
 - Icônes de compétences
 - GUI comprenant retour au menu, sortie du jeu, pause, barre de compétence et inventaire
 - HUD

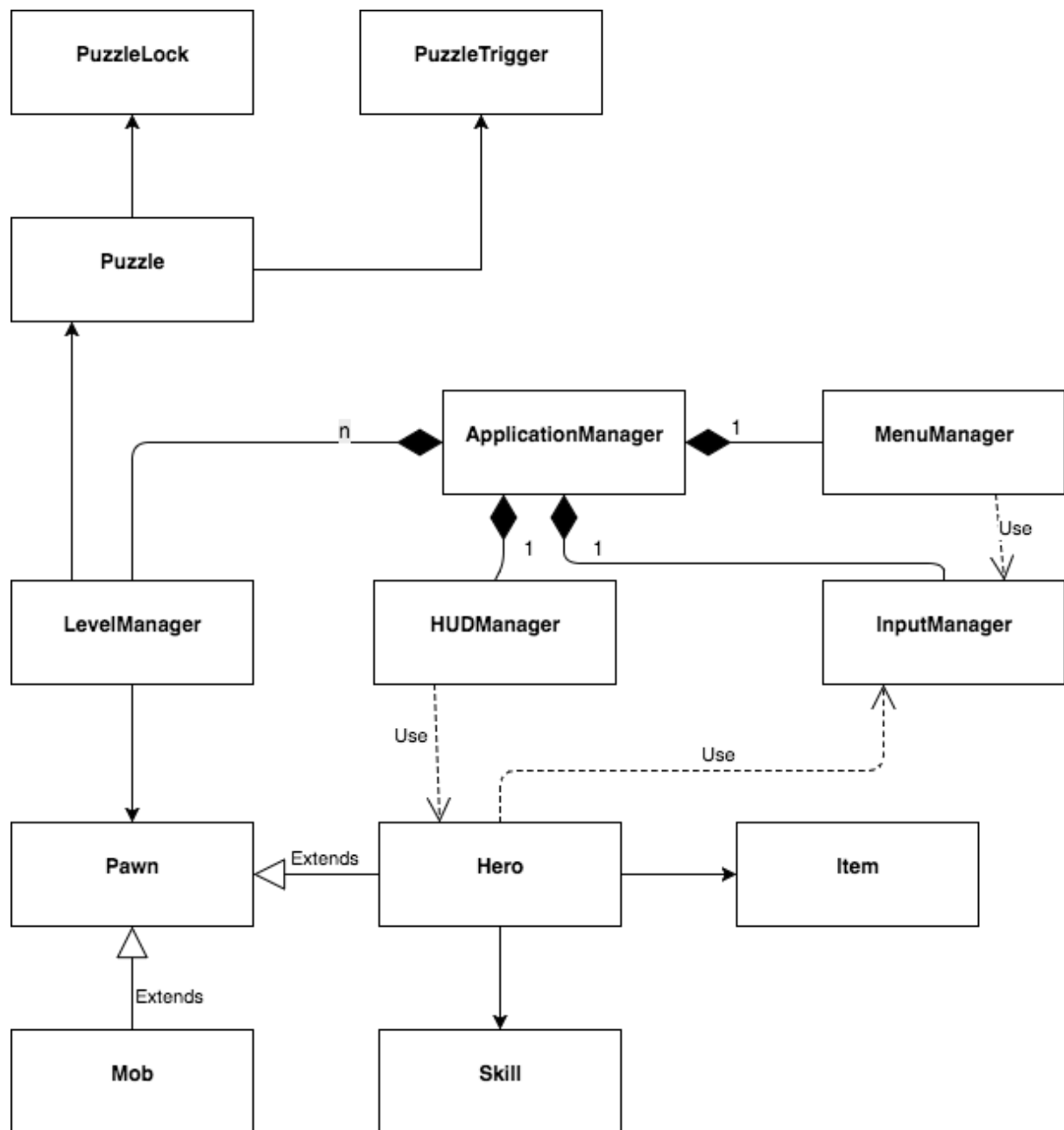
Plateforme technologique

Notre jeu sera réalisé en 3D avec le moteur Unity. Les scripts de jeux seront donc faits en C# avec Visual Studio et MonoDevelop. Les musiques de jeux seront réalisées à l'aide de Garage Band et les sons avec Audacity. Les éléments 2D seront réalisés avec des médiums d'arts traditionnels et un numériseur. Le personnage principal sera créé dans MakeHuman et importé dans Blender pour ajouter les touches finales.

Au niveau de l'organisation du travail, nous utiliserons un *repository* Git hébergé en privé sur GitHub pour gérer le code source du projet. Les divers outils de Google Drive seront utilisés pour la documentation du projet et un Trello servira à établir une planification plus sympathique pour nos tâches.

Architecture logicielle

L'application se base sur un gestionnaire principale "ApplicationManager" qui gère les changements de niveaux et la terminaison du programme. Ce gestionnaire contient plusieurs niveaux "Level". Chacun de ces niveaux contient à son tour un ou plusieurs défis "Puzzle". Chacun de ces défis contient un ou plusieurs verrous "PuzzleLock". Chacun des verrous doit être déverrouillé afin que le défi soit complété. Une fois complété, le défi enclenchera un ou plusieurs événements "PuzzleTrigger" qui à leur tour qui seront responsables de modifier l'environnement de jeux. Le niveau contient aussi des monstres "Mob" et un héros "Hero". Le héros contient un ensemble d'habileté "Skill" et d'item "Item". Le héros utilise un gestionnaire d'entrée pour ses déplacements. Il existe aussi un gestionnaire d'interface "HUDManager" ce gestionnaire utilise le héros pour, par exemple, choisir une habileté ou utiliser les items. Finalement, il existe un gestionnaire de menu qui s'occupe d'afficher et gérer le menu de pause du jeu.



Présentation de l'équipe

Antoine Beaulieu Lessard

Programmeur autodidacte qui à appris le C dans son sous-sol au secondaire et qui, comme tout jeune programmeur de sous-sol, a tenté et échoué à faire un jeu *from-scratch* dans le-dit sous-sol. Maintenant détenteur d'un DEC et (surprise surprise) étudiant en informatique à l'université laval. Il travaille avec des comptables dans un ministère depuis près de trois ans et n'a (autre que la tentative de créer un jeu *from-scratch*) jamais vraiment touché au développement de jeux vidéo, mais y joue depuis sa tendre enfance.

- DJ attiré à la création des sons du jeu.
- Maniaque de la gestion du Trello.

Catherine Robitaille

Avec une enfance sans console ni autres jeux d'ordinateur que ceux trouvés dans des boîtes de céréales, on ne peut pas dire que je suis une gameuse ni même très cultivée dans le domaine. Malgré cette enfance où l'électronique n'a de toute évidence pas été partie intégrante de ma vie, j'ai cependant développé mes talents en art et en scénarisation. À ma dernière année de secondaire, j'ai eu la chance d'expérimenter l'infographie et la programmation. C'est arrivé au Cégep que j'ai développé mes premiers jeux vidéos: Pac Man, Snake et les Échecs. Dans ce nouveau défi qu'est ce projet de jeu, je me donne pour objectif de découvrir pleinement la conception d'un jeu dans toutes ses étapes!

Quelques points sur moi :

- Se concentre sur la programmation back-end, la scénarisation et la conception de la structure du jeu
- Bonne connaissance dans les médiums d'art et création de petits films image par image
- Compte faire une sur-utilisation des fiches de bogue pour ce projet

Marc-André Brunelle-Langevin

Depuis ma tendre enfance, je m'intéresse aux jeux vidéo. J'ai débuté ma carrière avec Contra, je passais mes journées à jouer en espérant réussir à le terminer un jour (ce qui n'est pas arrivé). Depuis, je n'ai pas arrêté de jouer. J'étais toujours intrigué par les techniques nécessaires pour bâtir un jeu et ma curiosité n'a fait qu'augmenter avec le temps.

J'ai commencé à programmer au Cégep, dans des cours hors programmes, n'était pas certain si je souhaitais faire un DEC en informatique. Finalement, après le Cégep, j'ai décidé d'aller faire le bac en informatique et de me m'investir plus sérieusement dans la programmation. Depuis, je me suis trouvé un emploi comme développeur et j'essaie de toucher au plus de domaines possibles afin d'éclairer mon choix de carrières ainsi que pour choisir des cours optionnels qui pourront soutenir ce choix. Après avoir fait le cours d'infographie, j'ai développé une attirance pour la modélisation et l'intégration 3D. J'ai pratiqué un peu chez moi depuis et je souhaitais concrétiser le tout avec ce cours. Mon objectif est de faire carrière en tant que développeur de jeux, principalement du Front-end.

- Programmeur front-end par excellence
- Souhaite importer le moins possible d'éléments de l'Asset store et créer ses propres modèles
- Souhaite la création d'un jeu attirant et agréable à découvrir

Post-Mortem

À l'aube de la remise de notre dernier livrable, nous proposons une petite revue du projet dans son ensemble. Nous ferons un retour sur notre calendrier, et enfin un topo des erreurs à éviter lors d'un futur projet.

Au courant de la session, nous avons plusieurs remises à effectuer et une période assignée au travail de chacune de ces remises. Les rencontres que nous avons faites tout au long de la session ont permis une répartition des tâches rapide et efficace. Cependant, malgré cela, nous avons déjà une bonne charge de travail en dehors du cours et le temps assigné au projet était moindre par rapport à ce que nous aurions voulu. Le calendrier prévu et les dates des remises étaient corrects et bien réparties au cours de la session.

Étant donné que nous avons entamé nos débuts dans le monde du développement de jeu vidéo, nous avons rencontré bon nombre d'erreurs reliés à notre manque d'expérience. L'utilisation de Unity était pour nous un défi et cela a ralenti le développement du jeu. Ces erreurs se sont rapidement placées cependant et au courant des deux derniers livrables, nous devenions plus à l'aise avec les outils que nous avions.

Une autre erreur a été l'architecture de notre projet. Construire une architecture efficace et intuitive a été compliqué et nos sources sont pêle-mêle. Jouer dans le code de nos coéquipiers devenait alors plus compliqué, quoique tout de même réalisable. S'entendre sur une architecture de projet et prendre le temps de la réaliser ensemble est quelque chose qui devrait être fait dans nos projets futurs.

Enfin, nous constatons que le développement d'un jeu prend beaucoup plus de temps que prévu et que les objectifs que nous nous étions fixés étaient trop grands pour ce qui était réalisable. Nous aurions dû prévoir des éléments plus simples à implémenter et se concentrer à effectuer un jeu plus complet, et pourtant plus court. En bref, le but du projet n'était pas non plus adéquat à l'idée que nous avions du jeu à réaliser.

Pour terminer sur une bonne note, nous présenterons quelques points forts : les outils de développement choisis ont été adéquat à notre demande et l'utilisation de Git et Google Drive a permis une bonne collaboration des coéquipiers. La communication par Messenger a également bien fonctionné, même s'il y a eu des périodes plus occupées cela a permis de bien se mettre à jour et planifier nos rencontres.

Annexe 1 : Analyse technologique

Premier défi : Son avec audacity et musiques

(Antoine Beaulieu Lessard)

Ma première étape a été de tester la réalisation de bruitage à l'aide d'Audacity. Pour commencer, j'ai testé la création de son avec ce qui s'apparente à une sphère qui roule sur une surface dure : afin de parvenir à ce résultat, j'ai enregistré une boîte de conserve qui roule sur une table de nuit et j'ai appliquée quelques effets simples avec Audacity avec d'avoir une sonorité intéressante : changer la vitesse du son et modifier la tonalité du son par exemple. J'ai aussi enregistré des sons à la réception d'un coup pour un ennemi dans mon prototype, sons dont il s'agit de moi qui crie au travers de plusieurs effets ajoutés dans Audacity. Le tout fut relativement facile, ce qui confirme que l'utilisation d'Audacity pour les bruitages est un bon choix.

Par la suite, j'ai validé l'utilisation de Garage Band pour la composition de musiques de jeux. Pour ce faire j'ai composé deux pistes principales à l'aide d'échantillons fournis dans Garage Band, que j'ai agencés afin de satisfaire notre ambiance de jeux. J'ai donc créé une mélodie douce et une plus rapide, ce qui m'a aussi permis de tester la musique réactive avec ce prototype. Cet étape a été beaucoup plus longue que prévue, car j'ai dû apprendre à utiliser Garage Band.

Finalement, j'ai intégré le tout dans un petit prototype dans Unity qui inclus un ennemi qui produit des sons au contact et un cube de feu qui permet de tester les effets sonores 3D. J'ai aussi réalisé un petit contrôleur de musique qui diffuse une mélodie douce en tout temps et qui change pour l'autre musique plus rythmée à l'approche de l'ennemi.

Source pour le bruitage du feu : <https://www.freesoundeffects.com/free-track/fire2-89307/>

Deuxième défi : modélisation du héros

(Marc-André Brunelle-Langevin)

Le deuxième défi consiste en la modélisation du héros. Cela va nous aider à créer l'environnement qui conviendra au personnage selon sa grandeur et la hauteur de ses sauts.

Dans le prototype, nous retrouvons :

- Modelisation
 - Éléments physiques du personnage
 - Corps
 - Cheveux
 - Barbe
 - Vêtements
 - Épée
 - Rig complet
- Animation
 - Déplacement
 - Marcher
 - Courir
 - En attente
 - Attaque 1
 - Attaque 2
- Contrôles
 - Avancer (W)
 - Tourner (A/D)
 - Reculer(S)
 - Attaquer(Clic-gauche)
 - Sauter (Espace)
 - Bouger la caméra (Déplacement souris)
- Gestion de la physique du personnage

Pour construire le personnage, j'ai d'abord créé son modèle sur MakeHuman. Je lui ai ajouté son apparence et ses vêtements. J'ai ensuite exporté le modèle vers Blender afin de faire les retouches, car il y avait des incohérences entre la barbe et la tête, ainsi que les cheveux et les vêtements qui étaient en collision.

Après avoir terminé les retouches, je l'ai exporté vers Unity où j'ai conclu le personnage. J'ai utilisé certains assets disponibles pour les personnages de MakeHuman afin de créer les animations plus complexes. J'ai ensuite créé l'arbre d'Animator pour permettre de pouvoir se déplacer. J'ai créé un autre layer pour attaquer afin qu'il puisse le faire en se déplaçant. J'ai terminé les animations en ajoutant un masque sur l'animation d'attaque, pour la rendre plus réaliste et dynamique.

J'ai ensuite attaché l'épée du personnage dans sa main droite et placé sa position à (0,0,0) pour suivre la main.

Troisième défi : GUI personnalisé

(Catherine Robitaille)

Le troisième défi est de faire l'interface utilisateur. On parle ici du menu principale et des éléments d'interfaces non amovibles. En plus de cela, la couverture du jeu sera réalisé en dessin et utiliser comme présentation.

Le prototype d'interface comportera les éléments suivants :

- Un menu principal contenant la possibilité de jouer ou de quitter, avec une sélection de niveau.
- Une interface utilisateur (GUI) comportant le menu standard, les compétences, l'inventaire et la communication avec un PNJ. Les icônes des compétences sont dessinés à la main, puis numérisés.

Pour réaliser ce défi, j'ai commencé par élaborer les images des compétences élémentaires. Une fois les croquis faits, j'ai choisi d'utiliser des marqueurs à l'alcool pour le coloriage, car il s'agit d'un médium qui ressort bien lors de la numérisation. Pour l'image de couverture, j'ai travaillé avec ce même médium, mais j'ai détaillé beaucoup plus cette image comme elle servira de présentation et sera dans un format plus grand. Une fois les dessins coloriés et numérisés, j'ai recadré les images, joué un peu avec la luminosité, et finalement le résultat me semble satisfaisant.

Pour la deuxième partie de mon prototype, j'ai jouer avec Unity afin de créer un GUI basique. J'ai créé deux scènes, le menu principale et la scène de jeu, et ajouter à ces scènes les boutons requis. Il est donc possible dans le prototype de faire le passage entre les scènes, d'afficher ou de cacher les options et de quitter le jeu. J'ai également ajouter les icônes à une barre de compétence afin de donner une idée général de son résultat.

Annexe 2 : Rapport d'avancement

Première rencontre

14 octobre 2017

- Sons avec Audacity enregistrés sous format wav
- Modélisation du personnage, déplacement et caméra
- Croquis des icônes de compétences
- Liste des ressources de jeux complète
- Mécanique de jeux en cours

Deuxième rencontre

20 octobre 2017

- Prototype de sons complet
- Icônes de compétences complet
- Prototype d'interface en cours
- Image de couverture en cours
- Boucle de jeu complète
- Mécanique de jeux complètes

Obstacles rencontrés

Marc-André Brunelle-Langevin

J'ai remarqué que créer des animations demandaient plus de temps que je pensais. Il est assez difficile de s'assurer que celle-ci soit fluide avec les autres animations. J'ai dû aller chercher assets pour terminer le tout car je n'aurais jamais eu assez de temps.

Heureusement, la plupart ont été faites dans le prototype et je pourrai donc me concentrer sur autres choses pour le prochain livrable.

J'ai aussi rencontré des difficultés avec MakeHuman. Le logiciel semble davantage conçu pour modéliser des personnages récents et dans un style moderne. Notre héros ayant un style gréco-romain, il était très difficile de lui trouver des assets malgré MakeClothes à partir de Blender. Le logiciel semblait se concentrer davantage sur des chandails, pantalons, manteaux etc. C'est définitivement pas le style auquel on souhaitait donner à notre héros,

Annexe 2 : Rapport d'avancement premier jouable

Première rencontre

10 novembre 2017

- Initialisation du projet sur le git
- Liste des tâches et répartition
- Intégration du prototype de GUI comme base du projet
- Intégration de sons en cours
- Ajout du personnage modélisé en cours
- Map de jeu en cours
- Mise à jour de l'image de couverture

Suivi par messagerie instantanée

17 novembre 2017

- Intégration de de sons complète
- Modélisation d'une map de jeu améliorée depuis sa première version
- Modélisation et intégration du monstre complète
- Système de combat fonctionnel
- Modification GUI en cours
- Système d'inventaire en cours

Obstacles rencontrés

Au cours de cette étape, nous avons rencontré quelques accros qui nous ont pris du temps sur notre labeur. Tout d'abord, nos versions d'Unity n'étaient pas toutes les mêmes, et cela a causé des problèmes de compatibilité. Ces problèmes empêchaient notamment la compilation correcte du projet et ont été nuisibles. C'est également au courant de cette étape que notre GUI qui était relié à notre caméra a perdu son lien et ses fonctions. Suite à ces problèmes et quelques recherches, nous avons mis à jour Unity, procéder à une réimportation des éléments du jeu et redémarrer Unity après ces étapes. Le problème du GUI a été un peu plus persistant et a été réglé suite à un renommage de notre classe : la classe GUI entrainait en conflit avec celle de certaines dépendances ajoutées.

Dans l'ensemble, notre premier jouable nous a permis de découvrir les bons et les mauvais côtés du jeu vidéo. Il présente plusieurs bogues graphiques et fonctions incomplètes. Notre idée de ce prototype était quelque chose de beaucoup plus avancé, et pourtant nos objectifs n'étaient pas assez réalistes. Nous sommes tout de même satisfait de notre travail, car l'aspect général du jeu et de ses mécaniques a du sens et s'approche quand même de notre objectif.

Annexe 2 : Rapport d'avancement jouable publiable

Première rencontre

26 novembre 2017

- Planification des tâches
- Discussion sur la construction de la carte
- Mise à jour du Trello

Deuxième rencontre

3 décembre 2017

- Intégration du pouvoir spécial de la terre
- Système d'objectif en cours
- Amélioration de la carte du niveau de terre
- Intégration du boss de terre en cours
- Mise à jour de la planification des tâches

Troisième rencontre - Skype

10 décembre 2017

- Intégration du pouvoir spécial de feu
- Ajout d'une barre de vie sur le joueur et les monstres
- Modélisation du PNJ à faire
- Communication avec le PNJ à intégrer
- Animation mort [Optionnel?]
- Code de test à nettoyer
- Mise à jour du menu principale à faire [Optionnel]
- Post-Mortem à écrire

Problèmes rencontrés

Lors de ce dernier livrable, nous avons rencontré une grosse contrainte : le temps. La réalisation d'un jeu dans un temps très limité était un défi et bien que nous devenions plus à l'aise avec Unity, nous ne pouvions pas intégrer tout ce que nous voulions. Nous avons donc fait le choix des fonctions qui nous semblaient plus importantes pour un temps moindre lors de notre planification.

Annexe 3 : Liste des fonctionnalités

Ci-dessous la liste des fonctionnalités finales que nous avons constatées dans notre dernier livrable.

FPS (chargement exclus) presque toujours > 30	Complet
Toujours > 30	Complet
Presque toujours > 60	Ok
Toujours > 60	Ok
Interface	
Menus interactifs	Ok
GUI	Ok
Caméra	Complet
Contrôlable	Ok
Automatique	Ok
Saisie des entrées par bouton	
Ergonomique	Ok
Environnement	
Interactif	Ok
Agent autonome (golem)	Basique
IA (basique)	Basique
Avatar	
Animés	Ok
Audio	
Musique	Complet
Réactive	Ok
SFX	Ok
Ambiance	Complet
Réactive	Ok
Physique – Inclus dans Unity	
Collision – Inclus dans Unity	
Outil de développement – Make Humain, Blender, Audacity	Ok