INCOMING!!!

Document d’analyse

Par

Samuel DALLAIRE

Vincent BOILY

Travail présenté à

M. Michel Payette

Dans le cadre du cours Projet intégrateur

420-KMA-JQ

Session hiver 2017



**Phases de développement**

Prototype (Semaine 4-5-6)

Conception d’un programme très basique ayant pour but d’assimiler les divers concepts. Le résultat espéré serait 2 objet se tirant des projectiles sur 2 écrans. La durée de deux semaines est justifiée par la bonne compréhension des sockets de java : le transfert d’un projectile entre écrans et l’entrée multiple de commandes par les joueurs.

Structure de base (semaine 7)

Mise en place des images de la scène de jeu, des menus, des catapultes et des projectiles (images simples). Ce sera aussi l’occasion pour nous de travailler sur les proportions des divers éléments.

Moteur de physique (semaine 8)

Ajout des calculs nécessaires pour le déplacement des divers projectiles à travers les écrans et l’ajout de la gestion de collision. Un travail sur les proportions sera peut-être encore demandé.

Ajouts importants (Semaine 9-10)

Ajout de des projectiles et de catapultes supplémentaires, de nouveaux écrans dont le ciel ainsi que divers évènements comme le facteur vent. Il est à noter que s’il est jugé nécessaire dû à du retard ou tout simplement par choix, cette partie ainsi que les suivantes pourront être sacrifiées pour la cause en échange d’un programme de meilleure qualité.

Ajouts facultatifs (Semaine 11)

Ajout d’interaction environnement-projectiles et d’un historique de trajectoires de tirs. Nous comptons aussi passer le code en révision pour le simplifier et le rendre plus clair.

Ajouts très facultatifs (Semaine 12)

Ajout de power-up et/ou de mode de jeu supplémentaire tel qu’un 2vs1. Ce sera aussi le moment où nous effectuerons les finitions pour la présentation. Des petits détails de « show-off » pourront être ajouté pour obtenir plus de points.

**Classes Serveur**

Model

Catapulte : Élément principal du jeu. Contiendra un type, une quantité de vie, une direction de tir et une force de tir. Les contrôles du joueur seront reliés à cette classe. Cette classe contiendra sa propre position.

Projectile : Les projectiles auront une masse, un type, ainsi qu’une position contenue par elle-même.

Trajectoire : Cette classe contiendra les trajectoires de chaque tir effectué au cours de la partie. Son fonctionnement n’est pas essentiel au fonctionnement du jeu.

Controller

Moteur : S’occupe du calcul des trajectoires et du fonctionnement du jeu dans sa forme de base. Gere les collisions des projectiles ainsi que les déplacements de différents éléments du jeu.

Event : Gère la transition entre les menus et le jeu, et contient l’information des options, du mode de jeu et des joueurs pour le moteur.

**Classes Client**

View

ViewJeu : Affichage du jeu et contrôle visuel des éléments du moteur.

Menus : Affichage des menus de sélection.

Controller

Relais : Classe en charge d’accéder au serveur et de transmettre les données au reste du programme (la position des objets aux différentes scènes)