Maxime Brassard, Gabriel genest et Nathan gagnon

Programmation

420-204-RE, gr.00001

Dossier de conception

**Les Indécis**

Travail présenté à

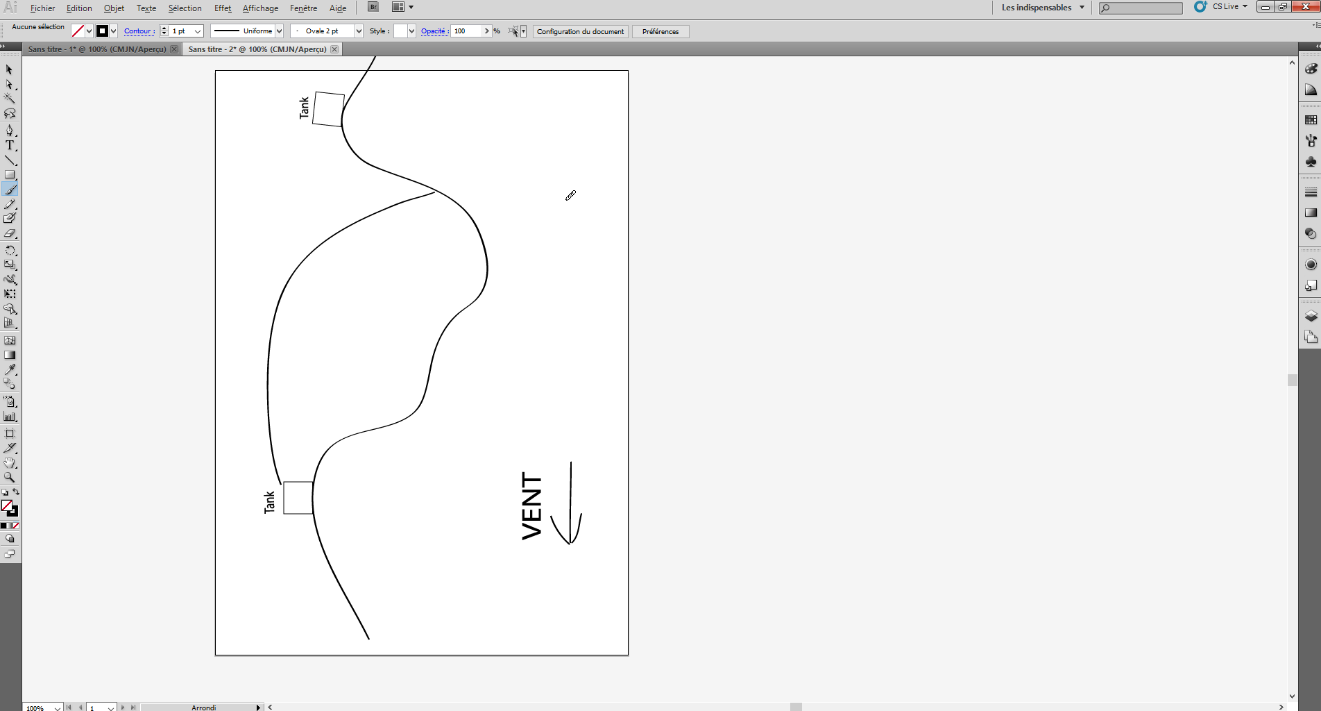
M. Jocelyn Goulet

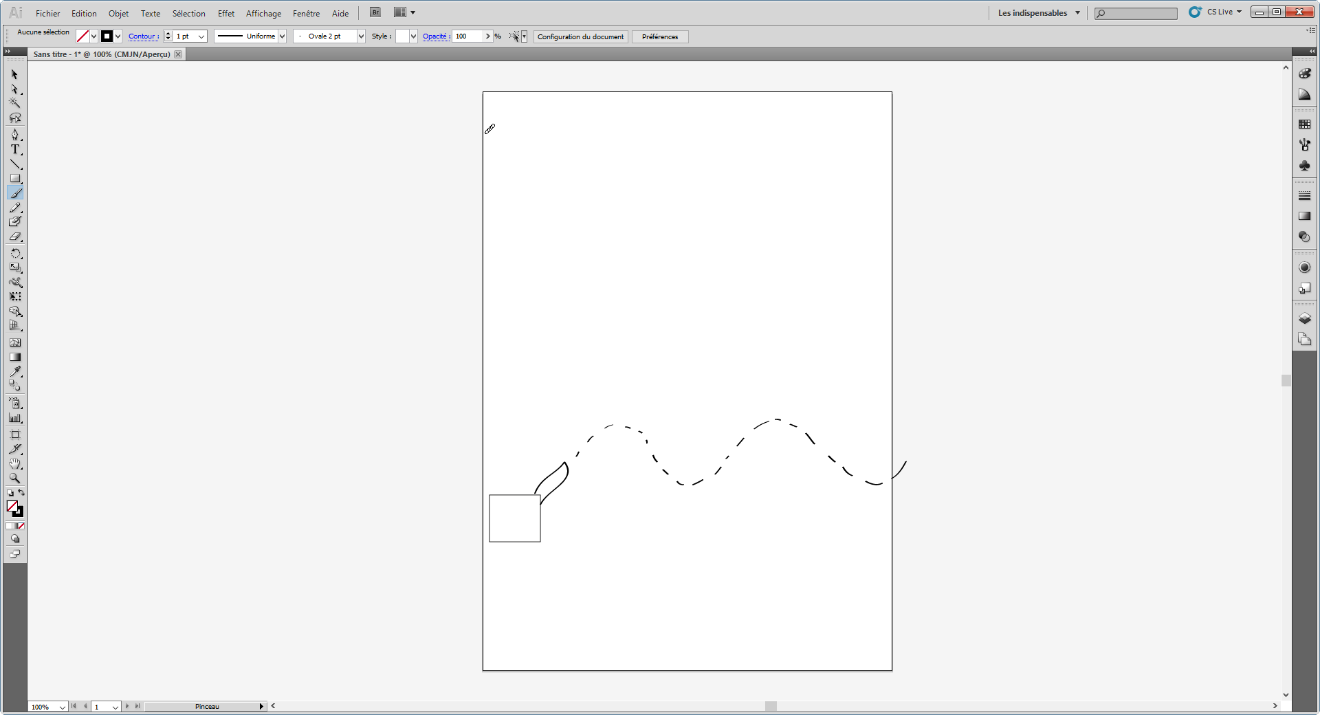
Département de l’informatique

Cégep Limoilou

Le 9 février 2017

## Description du projet :

L’objectif de notre projet est de s’amuser tout en comprenant ce que sont les impacts des variations d’environnements sur des projectiles. Il permet de mieux comprendre aussi le modèle de différentes fonctions mathématique.



Dans notre application, nous utiliserons la notion de projectile de mécanique classique. Il sera aussi question de l’effet de la résistance du vent sur lesdits projectiles.

Il y aura aussi des notions de mathématique. On utilisera des fonctions sinus, valeur absolue, inversement proportionnelle, ainsi que d’autre fonctions plus exotiques pour former le terrain (exemple : sin t + cos (sqrt(3)t) )

Une grande partie de la difficulté qui sera rencontrée vient du fait de trouver un IA performant et évolutif. Ensuite, faire un jeu en temps réel avec les gestions diverses qui sont aussi en temps réel (gestion des collisions, rebond, explosions…). Finalement, la génération d’un terrain aléatoire et sa gestion de destruction.

Les concepts nouveau x avec lesquels nous expérimenterons seront de faire des algorithmes (ou plutôt implémenter) d’intelligence artificielle évolutif avec le temps et le niveau. En restant dans les algos, un autre concept nouveau serait d’en faire qui tiennent en compte les différents concepts physiques qui varient en paramètres, tels que les rebonds et la résistance de l’air. Finalement, utiliser les interfaces graphiques de façon plus poussée (génération aléatoire, destruction…).

Notre projet va contenir plusieurs interfaces incluant l’interface d’accueil (celle présente à l’ouverture), l’interface d’options et l’interface principale.

L’utilisateur aura la possibilité de choisir parmi une dizaine de tirs différents pour adopter des stratégies adaptés à la situation. Comme mentionné plus haut, les différents tirs seront divisés en deux catégories (les tirs conventionnels et les tirs à énergie). Par exemple, parmi ces tirs, il y aura le tir rebondissent, le tir normal, des tirs suivant des fonctions, etc.

Distribution des rôles : - Chef : Nathan

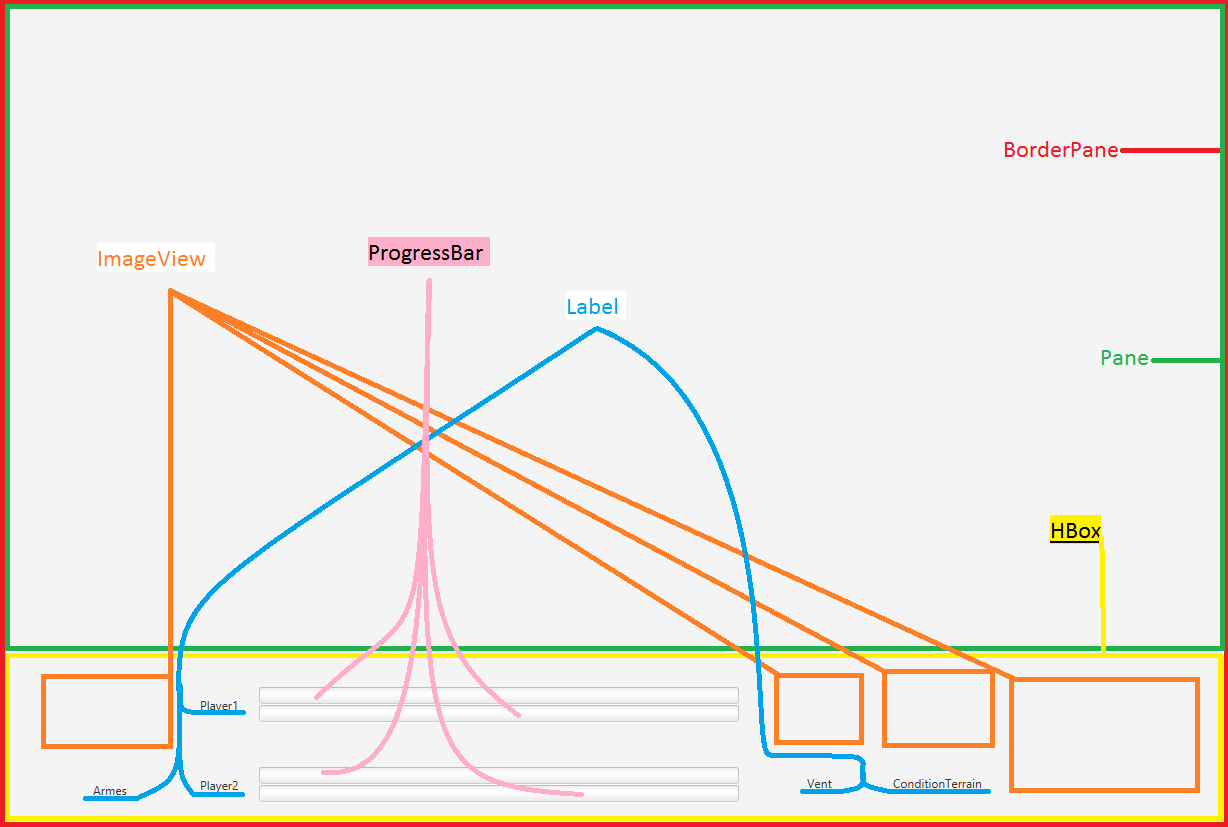
-Contrôle qualité : Gabriel

-Secrétaire : Maxime

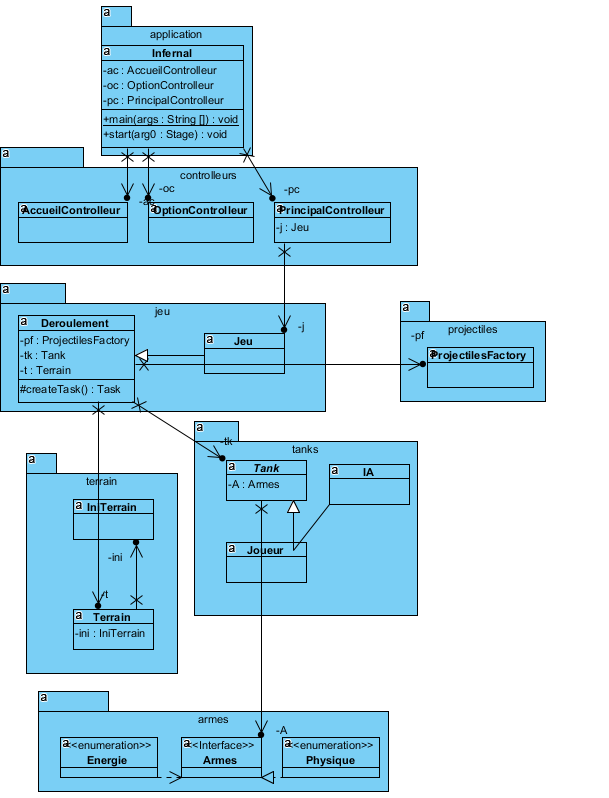
Type d’application : Windows

Langage : Java, CSS

Technologies impliquées : Il y avait possibilité d’utiliser les manettes d’xbox 360 mais le temps impliqué pour cette réalisation qui rembourre le projet ne serait que réellement conçus si le projet avance bien et que la librairie n’est pas trop difficile à utiliser.



## UML du projet :



## User Stories :

**1**

En tant que : Développeur

Je veux que : nous ayons un environnement de développement stable et efficace.

Afin que : -Nous ayons un environnement qui est propice au développement de l’application, que l’on soit au cégep ou non.

-Nous ayons un environnement sur lequel on peut travailler en même temps sans que les fichiers ne soient écrasés à chaque modification (GitHub)

-Nous ayons un moyen de communication rapide et efficace lorsqu’on n’est pas au même endroit.

-Nous ayons un environnement de développement Java avec *Eclipse*.

-Nous ayons

Test acceptation : Essayer, chacun de notre côté, de modifié un fichier à l’extérieur du cégep et vérifier si le fichier reste propre et ne se fasse pas écraser à chaque changement.

Complexité : 1

Effort : 1 (Fibonacci)

**2**

En tant que : développeur

Je veux que : nous ayons un packaging qui suit les règles du MVC

Afin que : -Nous ayons de l’ordre dans les classes et packages

-Nous ayons une organisation MVC pour qu’on n’ait pas à y retoucher en fin de projet

Test acceptation : Avoir une structure de package qui est divisé en packages généraux

Complexité : 3

Effort : 2 (Fibonacci)

**3**

En tant que : développeur

Je veux que : nous ayons une idée générale et documentée de ce que la hiérarchie des classes et packages doit être

Afin que : -Nous ayons une vision à moyen-long terme de comment structurer nos classes et les différents packages

Test acceptation : Avoir une hiérarchie qui convient à tous dans sa simplicité et dans son organisation cohérente.

Complexité : 3

Effort : 2 (Fibonacci)

**4**

En tant que : développeur

Je veux que : nous ayons une interface de jeu qui s’affiche sans problème

Afin que : -Nous ayons une interface de jeu qui s’affiche sur laquelle on peut faire des tests si besoin

-Nous ayons une interface du jeu qui permet d’avoir une vue d’ensemble et pouvoir déterminer des proportions convenable

Test acceptation : Faire afficher l’interface avec tous les éléments de base

Complexité : 3

Effort : 3 (Fibonacci)

**5**

En tant que : développeur

Je veux que : nous ayons une interface qui s’affiche pour le menu principal

Afin que : -Nous ayons une interface pour permettre d’entrer dans le jeu quand on est prêt

-Nous ayons une interface pour entrer dans un menu d’option

Test d’Acceptation : Faire afficher l’interface avec les boutons qui s’affichent

Complexité : 3

Effort : 3 (Fibonacci)

**6**

En tant que : développeur

Je veux que : nous ayons les prototypes des images et sons utilisés dans le jeu

Afin que : -Nous ayons des images de tanks

-Nous ayons de la musique

-Nous ayons des fonds d’écrans

-Nous ayons ces outils a porté de main sans avoir à les chercher à la dernière minute

Test acceptation : Avoir au moins une image de tanks et deux musiques (une pour le menu, l’autre pour le jeu)

Complexité : 2

Effort : 1 (Fibonacci)

**7**

En tant que : développeur

Je veux que : nous ayons une classe qui s’occupe de la gestion de la trajectoire des particules

Afin que : -Nous ayons des algorithmes de calcul des trajectoires selon des paramètres externes (gravité, vent)

-Nous ayons une classe qui peut être utilisée pour déterminer le point d’impact d’un projectile

Test acceptation : Être capable de calculer une trajectoire précise sans erreur

Complexité : 6

Effort : 5 (Fibonacci)

**8**

En tant que : Utilisateur

Je veux que : nous puissions sélectionner une arme

Afin que : -Nous puissions sélectionner l’arme avec la molette de la souris

* Le joueur puisse choisir son arme

Test d’acceptation : Être capable de choisir son arme en utilisant la molette. Chaque tic de la molette change d’une arme

Complexité : 3.5

Effort : 3 (Fibonacci)

**9**

En tant que : Développeur

Je veux que : Les opérations possibles sur les barres de vie et de gaz soient préparés

Afin que : -Cela soit facile, lorsque nous aurons à programmer les mécaniques du jeu de modifier ces barres

Test d’acceptation : Les tests faits sur les barres pour vérifier si les données restent cohérentes passent

Complexité : 2

Effort : 2 (Fibonacci)

**10**

En tant que : Développeur

Je veux que : Mon tank s’affiche sur l’écran

Afin que : Nous puissions faire des manipulations qui le nécessite (tirs, etc.)

Test d’acceptation : Test visuel (On voit le tank s’afficher)

Complexité : 2

Effort : 2 (Fibonacci)

## Références et documentation :

https://fr.wikipedia.org/wiki/Chute\_avec\_r%C3%A9sistance\_de\_l%27air