# 

**Sciences Informatiques et Mathématiques**

# **PROJET D’INTÉGRATION- ITÉRATION (1)**

***Super Billard version 1***

## **Professeure** : Niar Wafaa

## **Cours** : 420-204-RE

## **Fait par** : LACHANCE, Marc-Antoine, PASSUELLO-DUSSAULT, aaaaaaaaaaaaaaVittorio, RAFFIS, Édouard, RONDEAU, Victor

## **Session** **: Hiver 2017**

1. **INTRODUCTION :**
   1. **Objectif du document**

Le but de ce rapport est de pouvoir expliquer comment nous allons concevoir notre projet intégrateur pour sa première version. Le rapport est destiné à un public qui s’intéresse à l’informatique et plus précisément à la programmation de logiciels qui utilisent des concepts scientifiques.

* 1. **Portée du produit logiciel**

Le logiciel se nomme *Super Billard*. Il permet de jouer au billard en ajoutant la possibilité de modifier certains paramètres scientifiques. Ces paramètres pourront être sauvegardés même lorsque l’on quitte le jeu. Il ne permet pas de jouer à plusieurs et de modifier la taille de la table de billard. Ce produit s’intègre dans un contexte où les jeux utilisent des concepts scientifiques, surtout physiques. Le logiciel a pour objectif de permettre à ses utilisateurs de pouvoir voir concrètement les effets d’un certain concept scientifique. Le principal bénéfice de cette application est qu’il s’agit d’une représentation concrète et graphique de plusieurs concepts scientifiques. Les retombées du logiciel sont les gens vont pouvoir voir que les concepts scientifiques s’appliquent dans des activités de tous les jours. Le programme va permettre de modifier le frottement de la table, la masse des balles, la vitesse des balles et l’angle de tir. L’application utilise les concepts de la cinétique et du mouvement.

* 1. **Définitions, acronymes et abréviations**
  2. **Documents de référence**

ORACLE ET SUN MICROSYSTEMS. API Java SE 8. [En ligne]. https://docs.oracle.com/javase/8/docs/api/ [1er mars 2017].

ORACLE ET SUN MICROSYSTEMS. API JavaFX 8. [En ligne]. https://docs.oracle.com/javase/8/javafx/api/toc.htm [1er mars 2017].

SPOLSKY, Joel et ATWOOD, Jeff. StackOverflow. [En ligne]. http://stackoverflow.com/ [1er mars 2017].

* 1. **Planification des tâches**

Document MS Project.

1. **ANALYSE :**
   1. **Perspective du produit**

Dans cette première version, l’utilisateur sera capable de choisir l’angle et la force de aaa frappe d’un tir et d’effectuer le tir. Il sera aussi possible de frapper les balles dans les aaa trous pour les faire disparaître. Dans les prochaines versions les options de masses et aaa frottement seront disponible. Il y aura aussi l’ajout des règlements complets du billard aaa ainsi que l’addition d’une intelligence artificielle pour pouvoir jouer des parties de 8 aaa ball pool complètes.

* 1. **Caractéristiques des utilisateurs**

Ce programme est destiné aux utilisateurs de tous les niveaux. Tant bien aux jeunes gens qu’aux adeptes de billard. Les utilisateurs n’auront pas besoin d’avoir de formation et n’importe qui devrait être en mesure de jouer à Super Billard

* 1. **Modèle environnemental : les cas d’utilisation**
     1. **Diagramme de cas d’utilisation :**



* + 1. **Description narrative des cas d’utilisation**

***Cas d’utilisation***: Joueurau jeu

***Acteurs*** : Joueur

***But***: Démarrer une partie de billard

***Description***:

***Cours typique d’évènements :***

|  |  |
| --- | --- |
| **Actions Acteur** | **Réponses système** |
| 1. L’utilisateur initie en cliquant sur le bouton Jouer | 1. Le système répond en affichant la table et les champs de paramètres |
| 1. L’utilisateur entre les données de force et vitesses et clique sur le bouton lancer | 1. Le système met la balle en mouvement |

1. Le système indique une erreur car un des champs est vide.

***Cas d’utilisation***: Obtenir de l’aide

***Acteurs*** : Joueur

***But***: Apprendre comment jouer

***Description***:

***Cours typique d’évènements :***

|  |  |
| --- | --- |
| **Actions Acteur** | **Réponses système** |
| 1. L’utilisateur initie en cliquant sur le bouton Aide | 1. Le système répond en affichant la fenêtre d’aide (Règles et explication des paramètres) |

***Cas d’utilisation***: Changer les options

***Acteurs*** : Joueur

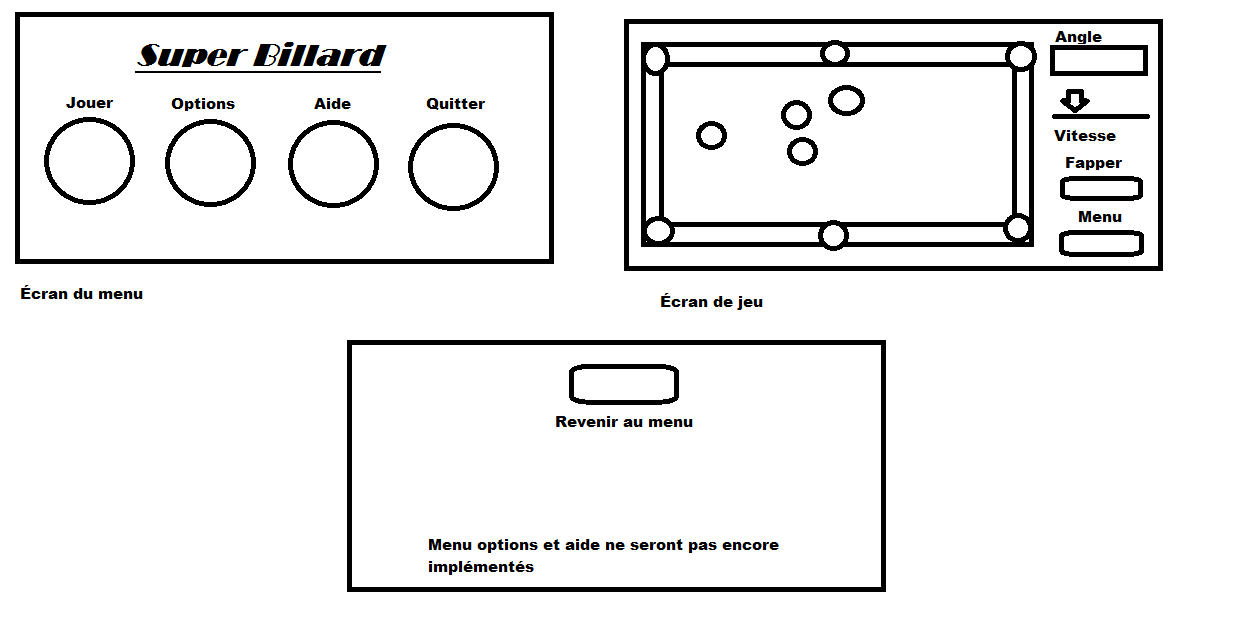
***But***: Changer les options

***Description***:

***Cours typique d’évènements :***

|  |  |
| --- | --- |
| **Actions Acteur** | **Réponses système** |
| 1. L’utilisateur initie en cliquant sur le bouton Options | 1. Le système répond en affichant la fenêtre d’options (changer la difficulté de l’ordinateur) |

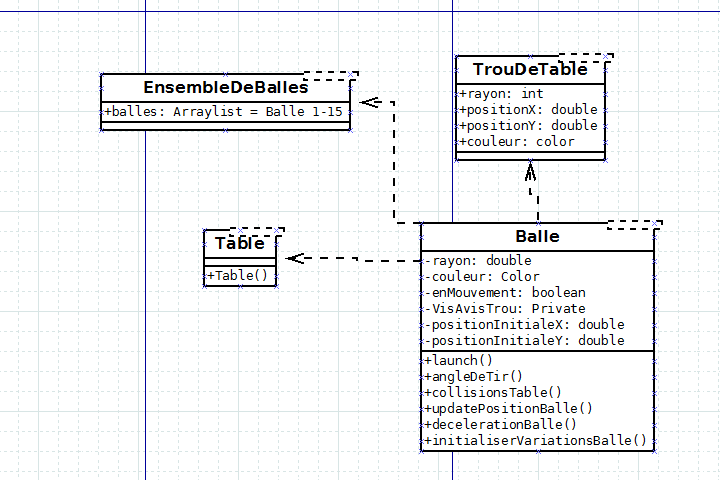
* + 1. **Prototypes d’interfaces :**



* 1. **Le modèle structural :**

**

1. **CONCEPTION**
   1. **Diagramme des classes logicielles**

**

Classe menus :

-menuJouer

-menuOptions

-menuAide

-menuPrincipal

* 1. **Classes modèle**

-Table

-EnsembleDeBalles

-TrouDeTable

-Balle

1. **IMPLÉMENTATION ET TESTS**

**4.1. Choix technologiques**

Plateforme utilisée : Microsoft Windows 7 Professionnel, Microsoft Windows 10 Famille et Ubuntu 16.04.2 LTS

Langage de programmation utilisé: JavaFX 2.2.75, Java SE 1.7.0, HTML 5, CSS 3 et JavaScript (Pour l’aide en ligne)

IDE utilisé : Netbeans 8.1

Autres outils : MS Project, Dia, Github, StackOverflow, API Java SE 8, API JavaFX 8

**4.2. Codage**

*Préciser ici où se trouvent les fichiers de l’application (nommer le chemin et dossier de remise dans lequel le projet sera déposé.)*

**4.3 Tests**

Tests effectués :

-Test de la force de la balle (Données utilisées : 1, 2 et 3)

-Test de la vitesse de la balle (Données utilisées : 1 à 10)

-Test de l’angle de la balle (Données utilisées : 0 à 360)

-Test de la position de la table (Données utilisées : 0 à 20)

-Test de la grandeur de la table (Données utilisées : 500, 1000, 325, 650)

-Test de la position des trous sur la table (Données utilisées : 0 à 20(puis en fonction de la grandeur de la table))

-Test de la grandeur des trous (Données utilisées : 10 à 30)

Critères de tests :

-Le programme doit rester fluide

-Le programme doit être simple d’utilisation

-Le programme doit faire ce que l’on attend de lui

-Tous les éléments graphiques doivent être présents et cohérents (Pas plein de couleurs différentes)

Bogues rencontrés :

-Lorsque la balle touche un coin, elle se déplace uniquement de gauche à droite ou de haut en bas

-La balle disparait et se divise en deux à chaque secondes

-La balle accélère lorsqu’elle devrait décélérer

-Les trous de la table n’apparaissent pas

-Lorsqu’on entre une vitesse trop grande la balle ne bouge pas

**Grille d’évaluation**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Équipe no :**  **Noms** |  | | | | |
|  | **Très satisfaisant** | **Satisfaisant** | **Assez satisfaisant** | **Peu satisfaisant** | **Insatisfaisant** |
| **Rapports :**  Planification réaliste  Diagrammes précis  Respect normes UML | **25** | **20** | **15** | **5** | **0** |
| **Codage :**  Concepts orienté objet (encapsulation,héritage,polymorphisme)  Organisation classes (MVC)  Réutilisation | **25** | **20** | **15** | **5** | **0** |
| **Convivialité :**  Normes ergonomiques  Aide en ligne\*  Fichier jar\* | **15** | **12** | **8** | **4** | **0** |
| **Validations :**  Absence des buggs  Gestion exceptions  Traitement erreurs | **15** | **12** | **8** | **4** | **0** |
| **Qualité du code :**  Normes programmation Documentation javadoc | **15** | **12** | **8** | **4** | **0** |
| **Qualité français :**  Rapport  Interfaces graphiques | **5** | **4** | **3** | **1** | **0** |
| **TOTAL** | **%** | | | | |

**\* :** *pourraient ne pas être exigées dans les 1ères versions.*