# 

**Sciences Informatiques et Mathématiques**

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

***Super Billard version 2***

## **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 d’expliquer comment nous allons concevoir notre projet intégrateur, fait dans le cadre du cours Projet Intégrateur du programme de Sciences, Informatiques et Mathématiques. PRÉCISER QUE C DANS CARDRE COURS PROJET INTEGRATEUR SIM Le rapport est destiné à la professeure de ce cours, Wafaa Niar et à toute l’équipe qui travaille sur le projet Super Billard. LE RAPPORT EST DESTINÉ AU PROF ET À L’ÉQUIPE

* 1. **Portée du produit logiciel**

Le logiciel se nomme *Super Billard* ENLEVER ESPACE(SuperBillard). 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 que les gens vont pouvoir prendre connaissance 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.

*LES UTILISATEURS DU JEU VONT-ILS VOIR LES ASPECTS PHYSIQUE? A MOINS DE LEUR FOURNIR CES EXPLICATIONS PAR DES MESSAGES, CES CONCEPTS SONT PLUTÔT CODÉS DONC LES JOUEURS NE VONT RIEN VOIR DE CA, ILS VONT JUSTE JOUER (NE PAS CONFONRE UTILISATION DU LOGICIEL ET PROGRAMMATION DU LOGICIEL)*

* 1. **Définitions, acronymes et abréviations**

Rien à signaler

* 1. **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. PRÉCISER QUE C’EST EN ANNEXE.

* NE PAS RAJOUTER LES COLONNES POUR DATES DÉBUT PLANIFIÉ ET FIN PLANIFIÉ, UNE SEULE DATE DÉBUT ET UNE SEULE DATE FIN POUR LES TÂCHES, C’EST A LA FIN DE L’ÉTAPE QUE LES DATES SONT RECTIFIÉES POUR BIEN MONTRER LE DÉROULEMENT DU PROJET ET SES AJUSTEMENTS
* IL FAUT METTRE LES NOMS DES COÉQUIPIERS (DANS UNE COLONNE DU PLAN DE PROJET) , OU ALORS INDIQUER ICI DANS LE RAPPORT, QUI JOUE QUEL RÔLE À CETTE ÉTAPE
* METTRE LA COLONNE DURÉE EN HEURES ET PAS EN JOURS POUR QUE VOUS PUISSIEZ AVOIR UNE IDÉE À LA FIN DU NOMBRE TOTAL D’HEURES QUE LE PROJET VOUS A PRIS

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.

*PRÉCISER QUE L’UTILISATEUR DOIT ÊTRE HABILE À UTILISER LA SOURIS SEULEMENT ET AVOIR LES RUDIMENTS DE BASE DANS L’UTILISATION D’UN ORDINATEUR ET CLAVIER.*

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



OK

* + 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~~

LA LIGNE 5 EST UN CAS ALTERNATIF À LA LIGNE 4 :

Cours alternatifs :

Ligne 4 : un des champs est vide : les système indique une erreur ..

*ATTENTION LE CAS D’UTILISATION DEVRAIT AVOIR UNE SÉQUENCE COMPLÈTE, DONC IL FAUT TERMINER LA DESCRIPTION EN PRÉCISANT QUE LA BALLE VA S’ARRÊTER ET QUE FAIT LE SYSTÈME APRÈS QUE LA BALLE S’ARRÊTE ??*

***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) |

INCOMPLET, PRÉCISER ICI QUE L’UTILISATEUR PEUT FERMER L’AIDE ET COMMENT ?

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

UTILISER LE MÊME TITRE DU CAS D’UTILISATION QUE DANS DIAGRAMME

***Acteurs*** : Joueur

***But***: Changer les options

***Description***:

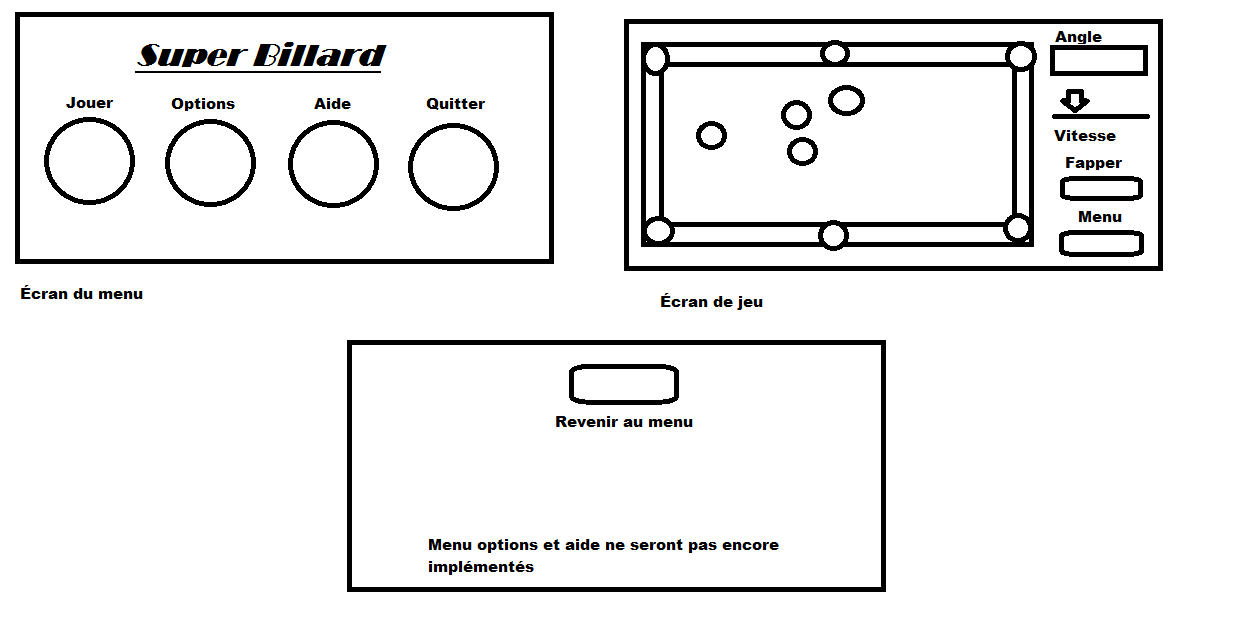
***Cours typique d’évènements :***

|  |  |
| --- | --- |
| **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) |

INCOMPLET : PRÉCISER QUELLES OPTIONS ON PEUT MODIFIER, COMMENT LES MODIFIER (SAISIE, CHOIX ?,…) ET COMMENT QUITTER CETTE OPTION

SI LES 2 DERNIERS CAS D’UTILISATION NE SONT PAS IMPLÉMENTÉS DANS CETTE VERSION, NE PAS METTRE DE DESCRIPTION DÉTAILLÉE, OU PRÉCISER QUE CA SERA COMPLÉTÉ DANS LES PROCHAINES VERSIONS.

* + 1. **Prototypes d’interfaces :**

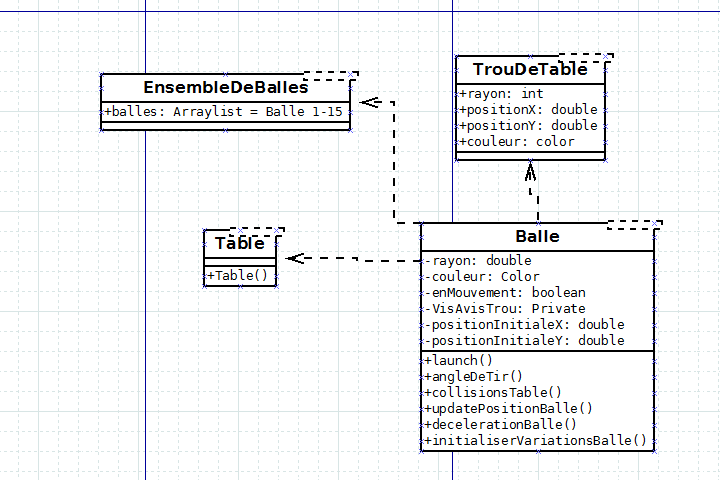


* 1. **Le modèle structural :**

**

*OK*

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

**

Classe menus :

-menuJouer

-menuOptions

-menuAide

-menuPrincipal

* IL MANQUE DES CLASSES (PRÉSENTES DANS VOTRE CODE SOURCE), A COMPLÉTER DANS DIAGRAMME
* IL MANQUE LES ATTRIBUTS DES CLASSES, A COMPLÉTER
* PRÉCISER QUE LES LIENS DU DIAGRAMME CI-DESSUS SONT UNE AGRÉGATION/COMPOSITION , CAR EN PRINCIPE, LE BOUT DOIT ÊTRE SOUS FORME DE LOSANGE ET NON PAS DE FLÈCHE
* DANS CE DIAGRAMME TOUTES LES CLASSES QUI ONT ÉTÉ IMPLÉMENTÉES DANS CETTE 1ERE VERSION
* VOUS POUVEZ PROFITER DE L’HÉTAGE ET POLYMORPHISME ICI (CONSIDÉRER UNE SUPER-CLASSE POUR LES BALLES QUI AURAIT LES ATTRIBUTS COMME RAYON ET COULEUR QU’ON RETROUVE DANS PLUSIEURS CLASSES
  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**

Les fichiers se trouvent dans l’archive SuperBillard\_V1.

PRÉCISER PLUS (MOODLE ? GITHUB ,….)

**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

*OK*

*REMARQUES POUS CODES SOURCES :*

* *ATTENTION , ÉVITER LES CARACTERES ACCENTUÉS (COMME LE NOM DU PACKAGE NETBEANS)*
* *NE PAS UTILISER DE CLASSE VUE DANS UNE CLASSE MODÈLE, PAR EXEMPLE VOUS UTILISEZ LA CLASSE MENUJOUER DANS LA MÉTHODE LANCH() DE LA CLASSE BALLE, , CE N’EST PAS PERMIS DANS ARCHITECTURE MVC, IL FAUT TROUVER UN MOYEN DE DÉFINIR UNE CLASSE CONTRÔLEUR QUI FERAIT LE LIEN ENTRE LES CLASSES MODÈLE ET LES CLASSES VUE.*
* *ENCAPSULATION DOIT ÊTRE COMPLÈTE DANS LES CLASSES MODÈLE*
* *INTÉGRER HÉRITAGE*
* *IL EST PRÉFÉRABLE D’UTILISER UN TABLEAU OU UNE COLLECTION DANS LA CLASSE BALLE AU LIEU DES ATTRIBUTS BALLE1 , BALLE2, ,…. BALLE15*
* *IL MANQUE DES COMMENTAIRES, EXPLIQUER CHAQUE CLASSE UTILISÉE ET CHAQUE MÉTHODE*

**Grille d’évaluation**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Équipe no : 2  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 | 87 % | | | | |

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