

Université de Caen Basse-Normandie
U.F.R. Sciences
Département Informatique
Année 2018-2019



Rapport Du Projet Bataille Navale en CPOO

Réalisé par :

David Ragot 21708377
Chloé Le Gentil 21700429
Mohamed Lamine Seck 21711412
Robin Moriniere 21606393

Table des matières

1	Introduction	3
2	Model MVC	3
2.1	Explication	3
2.2	Implémentation	3
3	Architecture Projet	4
3.1	Diagramme Package	4
3.2	Diagramme de Classe	4
4	Fonctionnalités implémentées	6
5	Problèmes rencontrés	8
6	Conclusion	8

1 Introduction

Le but de ce devoir était de concevoir, par groupe de 3 ou 4, un logiciel de jeu de bataille navale doté d'une interface graphique. Le logiciel devait être conçu entièrement en MVC, avec un modèle complètement indépendant de l'interface graphique. Le programme devait pouvoir être joué aussi bien à l'aide de l'interface graphique qu'en ligne de commandes. Le joueur humain joue contre l'ordinateur, qui joue de façon aléatoire.

2 Model MVC

2.1 Explication

MVC, ou modèle-vue-contrôleur, est un motif d'architecture logicielle destiné aux interfaces graphiques. Il est composé de 3 modules différents :

- Le modèle qui contient les données à afficher.
- La vue qui contient la présentation de l'interface graphique, ainsi qu'en ligne de commande.
- Le contrôleur qui contient la logique concernant les actions effectuées par l'utilisateur.

La vue et le contrôleur étant intimement liés, ils sont souvent réalisés dans un seul et même module.

Cependant chaque module est indépendant des autres, chacun peut fonctionner seul, permettant ainsi de changer un module ou de réutiliser un des modules dans un autre projet MVC.

2.2 Implémentation

Pour l'implémentation, nous avons choisi de séparer notre projet en 3 packages principaux

- **main** qui contient la méthode main de lancement du jeu
- **model** qui contient le package **player** contenant les différents types de joueurs et **utils** contenant les "Enums" pour les différentes option du jeu et un objet stats pour le score.
- **view** contient 2 sous-packages **graphics** et **terminal** contenant chacun leur sous package spécifique **controller**
- **graphics** contient 2 autres sous-packages **Pannel** comprenant les couches d'affichages et **newComponent** contenant des nouveaux objets pour la réalisation de l'interface graphique.

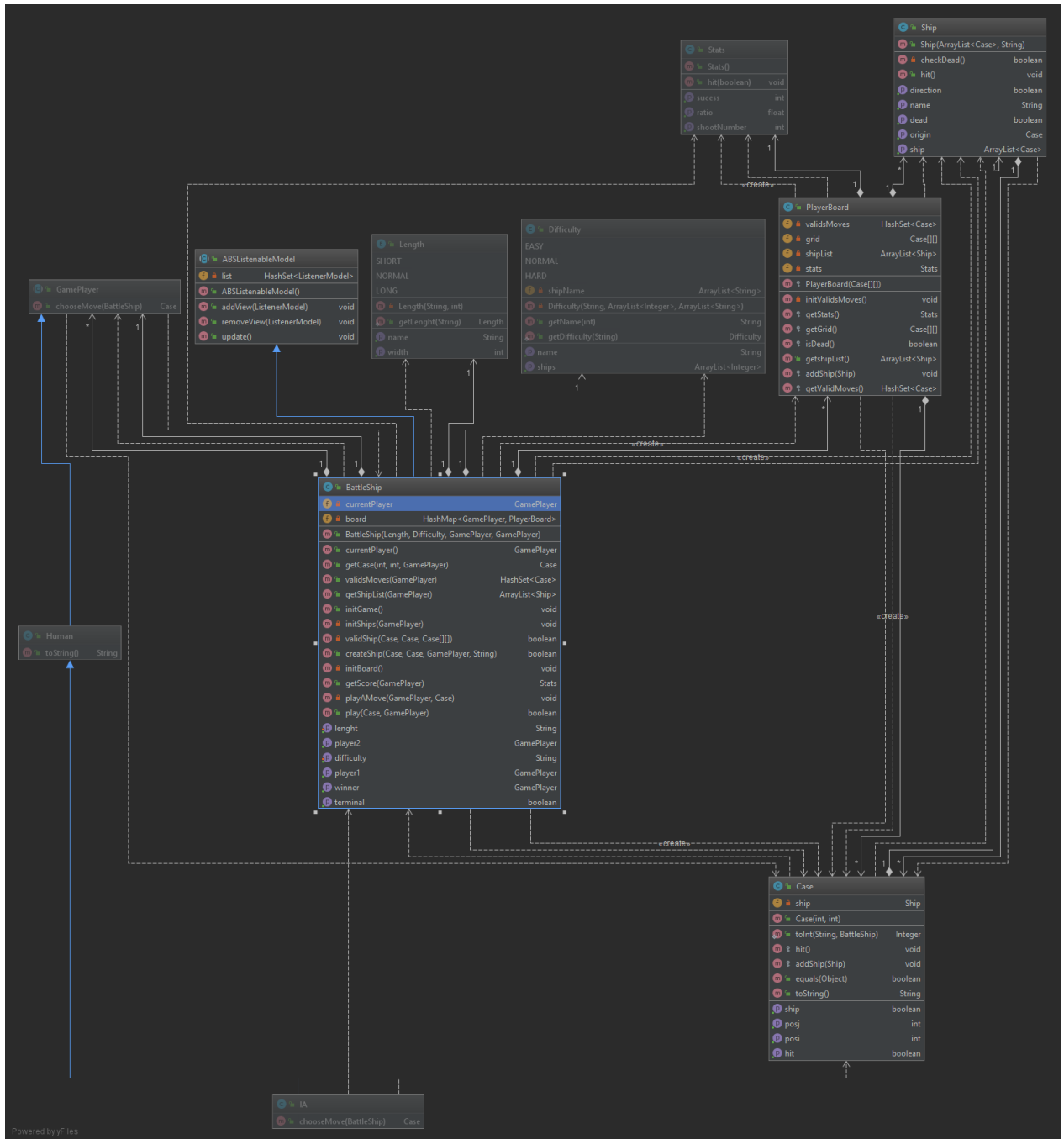


FIGURE 1 – Diagramme de classe du package model

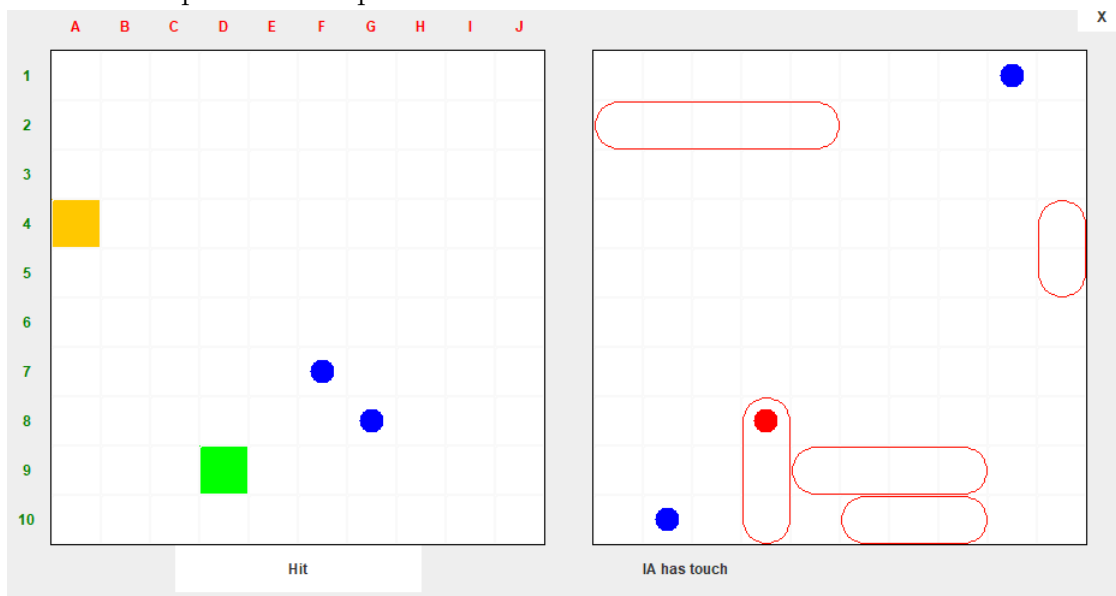
4 Fonctionnalités implémentées

Nous avons implémenté plusieurs fonctionnalités :
(Les commandes effectuées dans cette partie sont effectuées directement dans le répertoire des build ".class")

- Par défaut, nous lancerons tout le temps la vue terminale, pour lancer le jeu en mode graphique :

```
$java main.Main gui
```

- En mode graphique, le fait de voir la case que l'on "vise/target" en orange et en vert la case que l'on a cliquée.



- Quelques animations au survol comme sur les boutons "Hit" et "X" (pour fermer) tel que :



- La possibilité de choisir la taille de la grille parmi des tailles prédéfinies 8x8, 10x10, 15x15
- La possibilité de choisir la difficulté du jeu en changeant la taille des bateaux, c'est à dire plus on choisit un mode difficile, plus les bateaux seront petits et inversement. Graphiquement :



En ligne de commande :

```
$java main.Main terminal short hard
```

```
$ java main/Main terminal normal normal
normal
1.0
Normal
TA' board
 A B C D E F G H I J
0 - - - - -
1 - - - - -
2 - - - - -
3 - - - - -
4 - - - - -
5 - - - - -
6 - - - - -
7 - - - - -
8 - - - - -
9 - - - - -
Human' board
 A B C D E F G H I J
0 - - - - -
1 - - - - -
2 - - - - -
3 - - - - -
4 - - - - -
5 - - - - -
6 - - - - -
7 - - - - -
8 - - - - -
9 - - - - -
Enter your following move
Number : |
```

- Dans le jeu plateau de la bataille navale, le joueur qui touche un bateau peut rejouer de suite, nous avons choisi d'implémenter cette fonctionnalité pour rendre le jeu plus dynamique et moins long. Mais nous aurions pu le laisser en option sur le menu.

5 Problèmes rencontrés

Dans l'ensemble nous n'avons pas eu d'énormes problèmes mis à part que nous voulions créer un adversaire un minimum intelligent pour augmenter la difficulté et rajouter un type de joueur d'où le package player destiné à cela.

6 Conclusion

Pour conclure nous sommes plutôt satisfaits du résultat obtenu. Nous aurions quand même aimer pouvoir rajouter quelques fonctionnalités et en améliorer d'autres comme par exemple l'ia qui choisit tous ses coups de façon random, alors que nous arions aimez-la rendre plus intelligente pour qu'elle puisse être un minimum plus intéressante.