

# PROJET JAVA 2020

---

7 MAI

---

TOURE Boubacar Licence 2 MPCIE  
Groupe I3



---

# PROJET JAVA

## La Bataille Navale

### I. Introduction

A l'occasion du projet qui nous a été confié en java dans le but de mettre en œuvre le jeu de la bataille navale dans lequel deux joueurs doivent placer des <<navires>> sur une grille tenue secrète et tenter de <<toucher>> les navires adverses. J'ai l'honneur de vous faire part de mon rapport afin de vous donner une brève description de mon programme en insistant sur les fonctionnalités particulières ainsi que sur les différentes classes que j'ai eu à implémenter.

#### 1. LES CLASSES :

Lors de la création de mon projet, j'ai eu à disposer de cinq (5) classes parmi lesquelles nous avons : **Navire.java**, **Cellule.java**, **GrilleBataille.java**, **MyWindow.java** et **SaverFile.java**.

Ces classes ont tous une particularité très importante et chacun joue un rôle indispensable pour le bon déroulement de mon projet.

- La classe **Navire.java** possède 4 champs permettant d'identifier au mieux un navire (c'est-à-dire si le navire est un navire ennemi ou allié mais aussi le type de navires). Les informations sur ses champs sont bien détaillées en commentaire dans la classe.
- La classe **Cellule.java** est une classe qui hérite de la classe JButton de la bibliothèque javax.swing. Son principal but est de nous permettre de représenter une case de navire en bouton mais aussi de nous permettre de savoir si une partie de navire a été touché ou pas. Les informations sur ses champs et méthodes y sont bien détaillées en commentaire.
- La classe **GrilleBataille.java** est l'une des classes les plus complexes ou du moins le plus complexe de mon projet. Ses fonctionnalités sont indispensables pour la bonne gestion du jeu. Elle possède de nombreuses champs et méthodes dont leur rôle sont bien détaillées en commentaire. En gros, elle possède de nombreuses méthodes statiques qui me permettront l'intégration de certain de mes codes dans l'interface graphique. Comme son nom l'indique, elle correspond au champ de batailles du jeu (la surface permettant de contenir tous les navires ennemis tout comme alliés) et permet de simuler le déroulement du jeu dans la console ainsi que celui de l'interface graphique. Elle hérite de la classe JFrame de la bibliothèque javax.swing et permet donc la représentation des objets de type Cellule (Bouton) à son bord. Au début, j'ai opté plus pour une approche vers la console ce qui m'a pris beaucoup de temps et a alourdi mon code. C'est à deux jours avant le dépôt, que j'ai

---

remarqué que j'implémentais des méthodes qui pouvait être implémenté directement dans l'interface à l'aide de ma classe Cellule et MyWindow. Je voudrais pas même vous assurer avec certitudes que mon code est à 90 % terminer au niveau de la console (en jouant dans la console). Mais par-contre j'avoue qu'il y a encore quelque petite imperfection dans l'interface. Notamment au niveau de l'enregistrement des données ou j'avais cru à une simple saisie du déroulement du jeu. C'est juste aujourd'hui qu'un ami m'a expliqué le but de la sauvegarde en question (paraît-il qu'il faut pouvoir redémarrer la partie).

- La classe **SaverFile.java** est une classe permettant de créer un fichier et de sauvegarder les différentes données du jeu dans son sein. Elle offre aussi la possibilité de lire les différentes données qui compose un fichier.
- La classe **MyWindow.java** est la classe principale de mon projet. Elle hérite de la classe JFrame de la bibliothèque javax.swing et permet d'afficher l'interface graphique du jeu de manière visuelle.

## 2. EXPLICATION DE L'I.A :

La méthode que j'ai eu à développer pour la mise en œuvre de mon Intelligence Artificielle est la suivante. Tout d'abord, mon jeu propose **trois (3) niveaux de difficultés**. Le premier niveau nous permet de jouer à arme égal avec l'ordinateur. On dispose tous de la même capacité et de la même chance de victoire et défaite. Par-contre le deuxième niveau est d'un autre visage. L'ordinateur passe automatiquement en **niveau 2** s'il se sent menacé (i.e s'il a une vie entre 4 et 6) ou si la différence de vie est supérieure ou égale à 5. En niveau 2, les tirs des navires ennemies sont de plus en plus précis et dévastateur car ont effectuée un test afin de repérer la position de tir la plus favorable avec ma méthode **bestChoix (comportement expliquer en commentaire)**. Une fois cette position repérée, on vérifie dans notre arsenal de tir, l'arme la mieux adaptée pour notre situation (**on a 3 chances sur 4 de tomber sur une bombe très puissante**) et on s'en sert pour atteindre le plus de navires possibles. Et enfin le dernier niveau de difficulté, qui est le plus puissant que j'ai mis en œuvre. Il offre la possibilité à l'ordinateur de faire **des tirs avec plus de 80 % de précision** ajouter à son immense puissance de tir qui lui est offerte par la méthode bestChoix. Contrairement au niveau 2, le niveau 3 utilise uniquement ses **3 meilleurs bombes** lorsqu'il tombe sur une possibilité de tir qui lui est très avantageux (il utilise la bombe classique dans cette situation uniquement quand il ne dispose plus de super bombes à son appuie).

Mais comme chaque programme, il se peut qu'il y ait des failles, vous penserez souvent que le niveau 2 est plus mieux que le niveau 3 dû à un jeu de hasard. Par-contre ce n'est pas le cas, car la probabilité de victoires en niveau 3 de l'ordinateur est bien plus important que celui des autres niveaux en fonction du nombre de répétition. J'ai mis en œuvre une approche simpliste qui permettra à

---

l'utilisateur d'espérer à une victoire. Même si cette victoire lui serait assez compliqué à obtenir.

**N.B.** J'ai eu à implémenter de nombreuses méthodes afin que vous n'ayez pas à toujours saisir à la main et de laisser le jeu positionner vos navires à votre place ainsi que de lui laisser choisir aléatoirement votre position de tir notamment en utilisant les méthodes utilisées par l'ordinateur. J'ai pris du temps pour faire en sorte que ses méthodes soient utilisables non seulement pour l'ordinateur mais aussi par le joueur pour vous rendre la tâche plus facile. Vous n'avez qu'à décommenter certaine partie dans le code où à commenter d'autre. Tout est signalé en commentaire pour info.