

Clément BELLANGER

Charlotte LANUEL

Lucas SCHWAB

Dossier d'implémentation

Bataille navale

DépotGit : https://github.com/lukyky414/DP2018_Team_Rocket

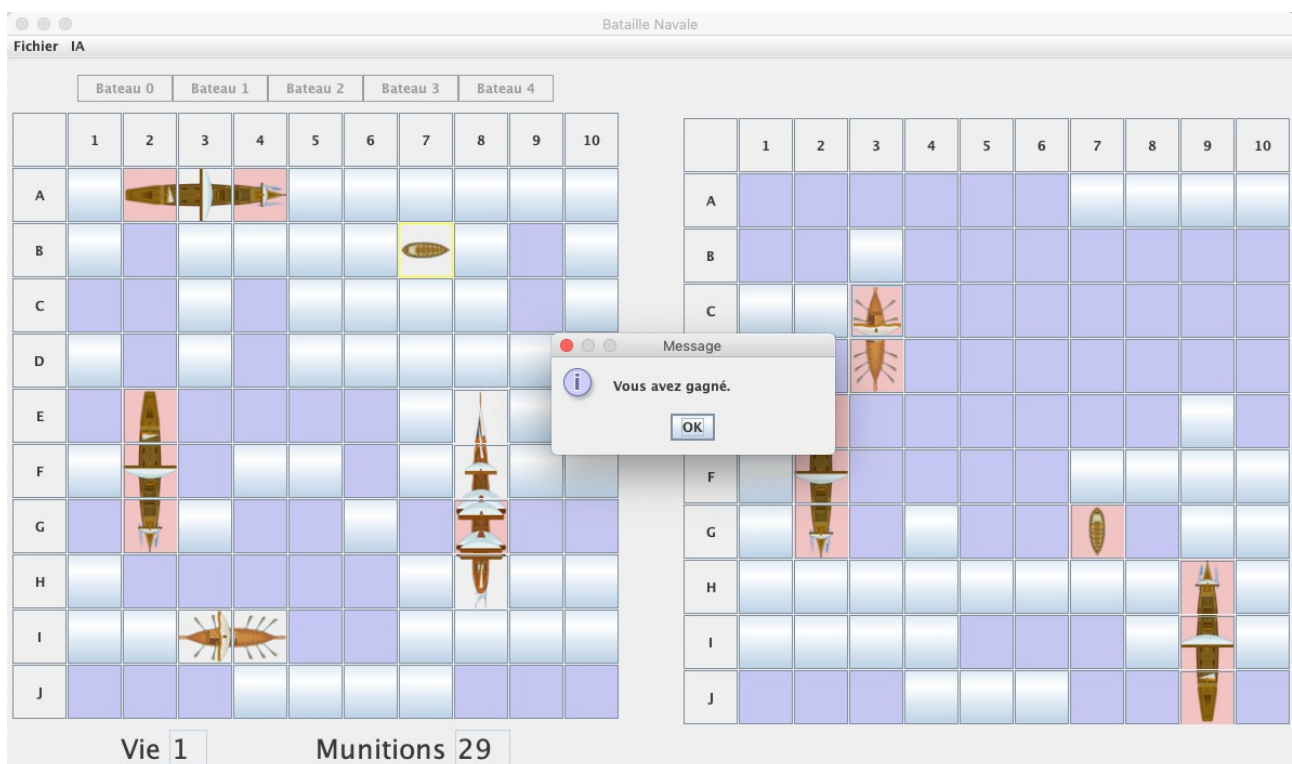
(Explication des règles dans le README sur github)

Introduction

La bataille navale de notre équipe contient quatre époques, chacune avec ses particularités.

1. L'époque médiévale est constituée de : une barque, deux petits bateaux de guerre (deux cases), deux bateaux de guerre moyen (trois cases) et un galion (quatre cases).
2. L'époque contemporaine est constituée des bateaux de nos jours : un de deux cases, deux de trois, un de quatre et un de cinq.
3. L'époque du futur est composé : de deux vaisseaux de deux cases, d'un de trois cases, d'un de quatre cases et un de cinq cases.
4. Enfin, la meilleure époque de tous les temps : Star Wars. La flotte est composée d'un Faucon Millenium sur une case, d'un TieFighter sur une case, de l'Etoile de la mort sur une case et de deux x-wings sur deux cases.

Pour chacune de ces époques et chacun de ces bateaux, une puissance de tir les distingue et rend en réalité des petits bateaux insignifiants face à des puissants bâtiments de guerre.



L'exemple ci-dessus montre une partie où le joueur a gagné contre une IA. Les cases neutres n'ont pas été testées, les cases en violet sont celles où on a tiré dans l'eau et les cases en rouge sont celles où nous avons touché le navire. L'image entière ne s'affiche que si le bateau est mort.

Époque et choix de conception lié

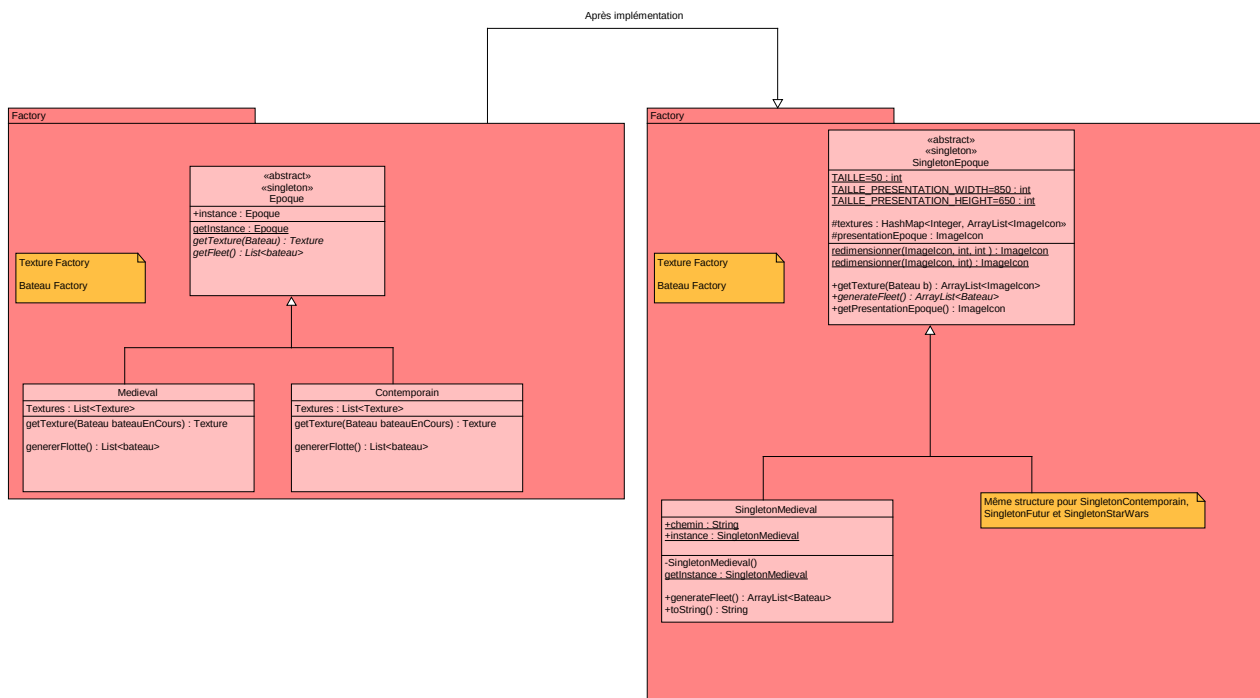


Diagramme montrant la différence entre la prévision et la réalisation

La conception des « SingletonEpoque » a été faite de telle façon que l'on puisse ajouter une nouvelle époque facilement.

C'est donc elle qui gère les textures de chaque bateau pour son époque.

Chaque bateau peut également avoir une vie différente (toujours inférieure ou égale à la taille du bateau), une taille différente, une zone de tir différente et des munitions différentes en fonction de l'époque.

Les bateaux sont générés avec tous les paramètres de l'époque par la méthode `generateFleet()`.

Pour les textures, on utilise des `ImageIcon` qui sont stockés dans l'attribut `textures`. Celui-ci contient comme clé l'id du bateau et une liste d'`ImageIcon` pour chaque case du bateau.

Une image de présentation de l'époque existe également mais elle n'est pas utilisée.

Vue, Controlleurs et choix de conception lié

La vue était la grande absente de notre dossier de conception.
Nous voulions peaufinez celle-ci afin d'avoir un bon rendu.

En effet nous ne connaissions pas l'étendue des possibilités offertes par les JButtons (images, couleurs, bordures, ...), et certaines autres fonctionnalités de Swing.
Nous savions néanmoins qu'un pattern Observer Observable serait adapté dans ce cas.

On a donc finalement utilisé une grille fonctionnant sur un GridLayout, les autres éléments utilisant principalement des JPanel avec des BoxLayout verticales et horizontales imbriqués les uns dans les autres.

Un JMenu permet de choisir l'époque à lancer.

Lors du placement des bateaux, des JButtons permettent de choisir les bateaux qui seront placés en 1^{er}.

Nous avons aussi cherché à minimiser le nombre de controleurs, il n'y a normalement qu'une seule instance par controleur.
Ceux-ci sont adapté pour reconnaître directement le JButton et éventuellement la grille qui les a appelés.

Modèle, DAO et choix de conception lié

Afin de pouvoir charger et sauvegarder les données, nous avons utilisé 2 DAO qui font une sauvegarde des parties en XML et en binaire (.sav).

Pour le modèle le design pattern Observer, Observable a été utilisé, les classes sont également séparés entre vue, modèle et controleurs.

Récapitulatif des fonctionnalités implémentées

- Fonctionnalité de sauvegarde de 2 façons différentes.
- 4 Époques et les époques peuvent s'ajouter facilement.
- Taille, munitions, zone de tir et vie pour les bateaux par époque (un bateau de 2 cases peut parfois couler avec juste un tir). Les règles sont identiques pour le joueur et l'ordinateur.
- IA avec deux stratégies (aléatoire pour facile et tir en croix pour normal (et hardcore)).
- Visualisation de deux grilles avec différenciation avec les cases touchés, les bateaux adverses apparaissent seulement si ils sont morts ou à la fin de la partie.

Le RMI n'a malheureusement pas pu être implémenté.

Ressources utilisés

Nous avons utilisé certaines images des projets :

<https://github.com/ghaniturismo/BatailleNavale-Equipe3>

<https://github.com/ChristophePhan/bataille-navale>

Pour l'époque StarWars :

https://vignette.wikia.nocookie.net/disney/images/0/0d/X-Wing_Top_View.png/revision/latest?cb=20131229051005

<https://cdn.wallpaperjam.com/52e32189ba19082eb42aa05c130e4d9be46c56e8/star-wars-star-destroyer.jpg>

<http://thehypedgeek.com/wp-content/uploads/2017/09/Lego-UCS-millennium-falcon-top-down.jpg>

<https://www.thegreenhead.com/imgs/star-wars-death-star-worktop-saver-2.jpg>

Pour l'époque contemporaine alternative (inutilisée) :

<http://www.jeux.fr/jeu/bataille-navale>