

MICHELETTI Julien

THIS Elliot

WILLAIME Noah

# Design patterns

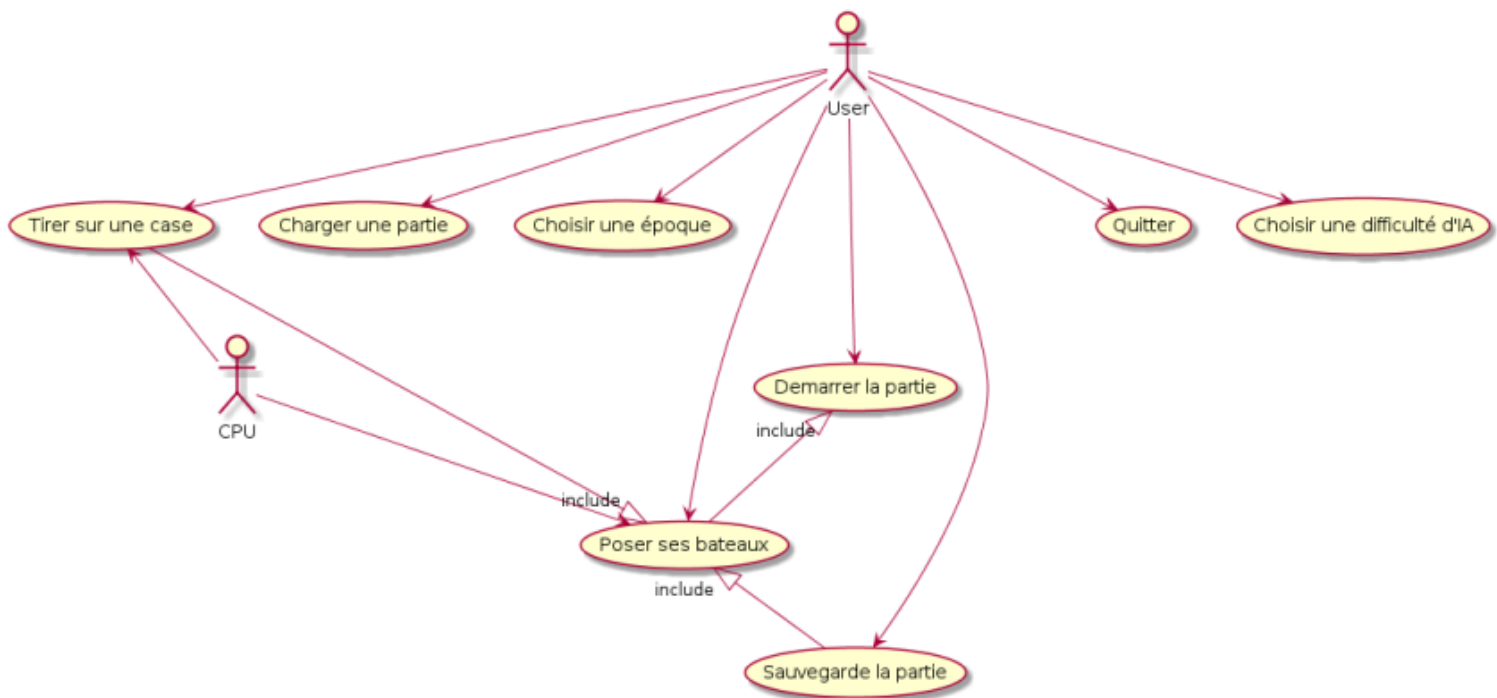
## BATAILLE NAVALE

*Groupe : SegFault*

## Lien git vers le projet

[https://github.com/NoahWillaime/Bataille\\_Navale.git](https://github.com/NoahWillaime/Bataille_Navale.git)

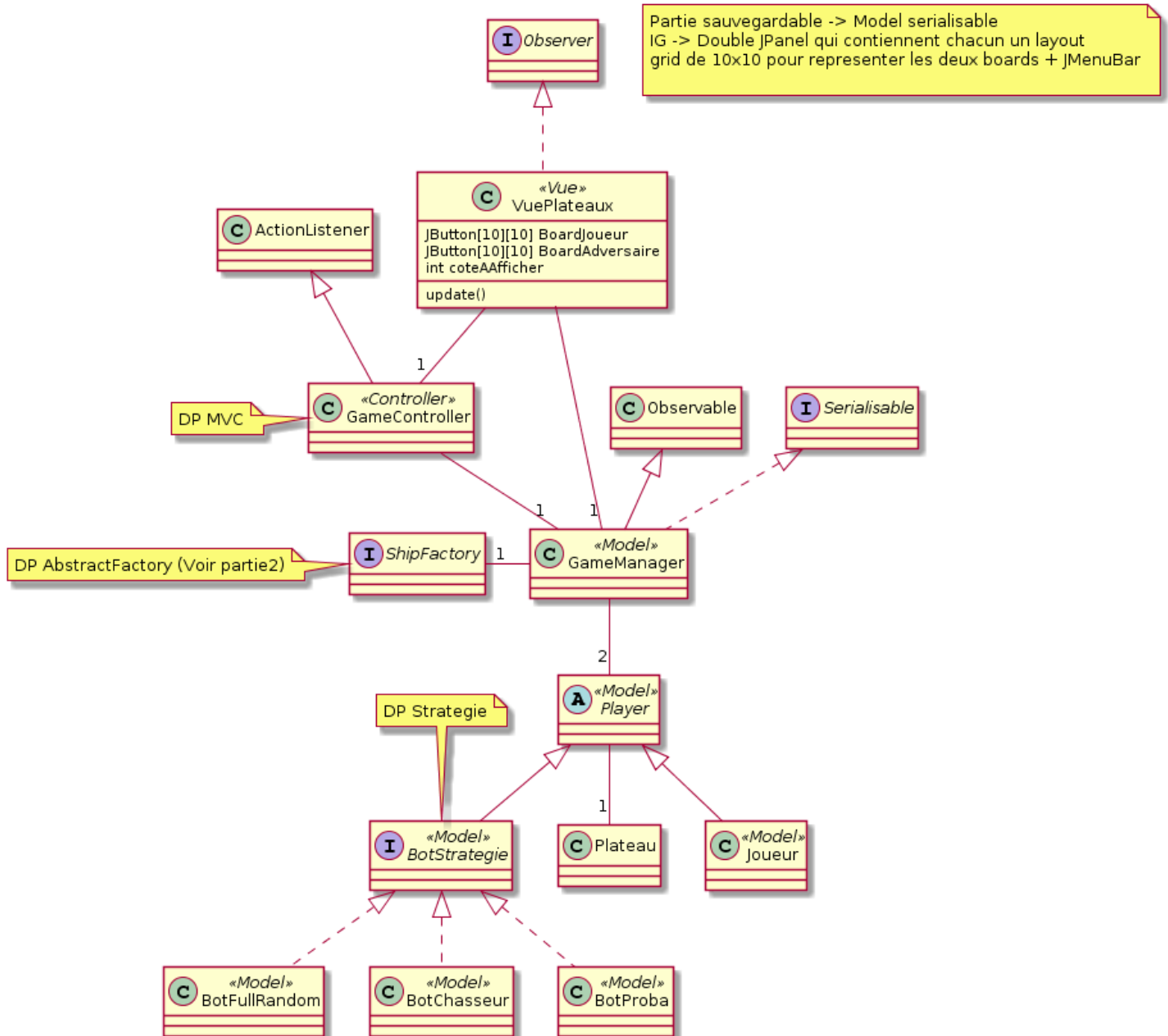
## Diagramme des cas d'utilisation



# Diagramme de classes du projet

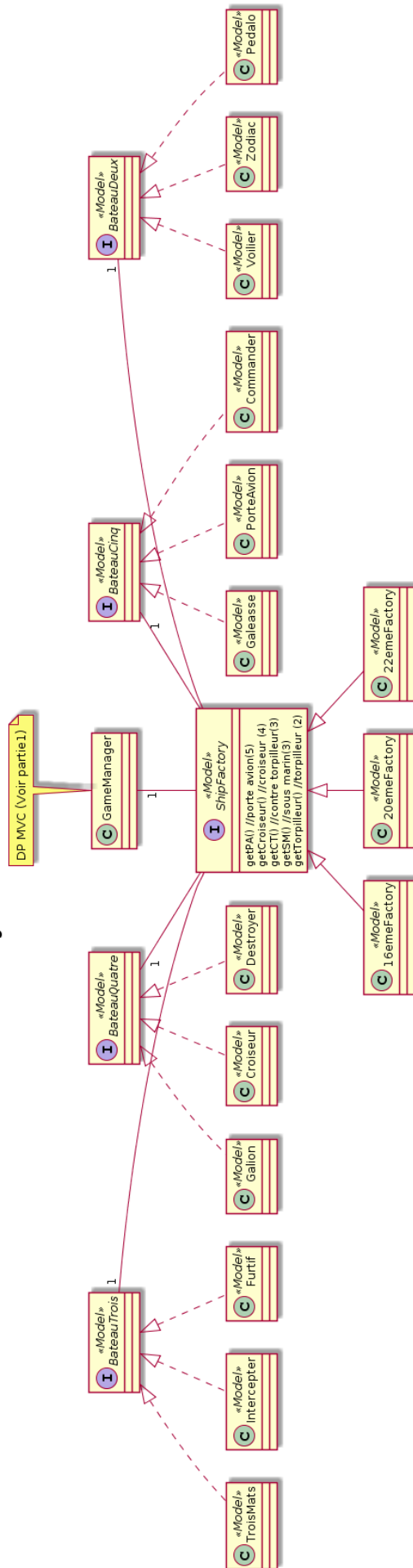
## - Partie 1/2 : MVC

Diagramme UML - Bataille Navale (Partie 1)



- Partie 2/2 : Abstract Factory

Diagramme UML - Bataille Navale (Partie 2)



## Explication du choix des Design Patterns

- MVC :

Permet la gestion du logiciel séparément grâce au modèle et aux vues / contrôleurs.

- Strategy :

Permet l'application des différents algorithmes de difficulté au joueur artificiel.

- Abstract Factory :

Permet la gestion des différents bateaux présents dans le jeu car on doit instancier dynamiquement les sous-classes représentants chaque bateaux.