



# Programmation orientée objet

Patchwork

BASTOS Antoine

ROUX—DAUGROIS Yann

23/12/22



## Sommaire

Introduction.....	3
Contrôleur.....	3
Modèle.....	3
Automa.....	3
Vue.....	4
Schéma synthétique de l'architecture.....	5



## Introduction

Nous avons tenu compte des remarques faites durant la soutenance du rendu intermédiaire pour retravailler notre architecture.

## Contrôleur

Précédemment nous avons amalgamé le rôle du contrôleur avec une classe `GameBoard`, désormais supprimée. Toutes deux s'occupait de fait du gameplay et des scénarios du jeu, à ce défaut que `GameBoard` était également un élément du modèle. Désormais le modèle est indépendant des règles et de la manière par lesquelles celui-ci est modifié. C'est uniquement dans le contrôleur que ces opérations sont effectuées, et que l'état du jeu à un instant donné est représenté.

## Modèle

Après avoir compris que notre architecture précédente complexifiait d'avantage le projet plus qu'autre chose comparativement à la portée du sujet, nous avons supprimé les classes / abstractions superflues (`XML`, `ButtonOwner`, `Event`, etc...) et à la gestion des éléments du jeu. Afin de les rendre d'avantage immuable, dans la mesure du possible.

Une classe s'occupe de la représentation du plateau et de ses joueurs (`TrackBoard`) et une autre des patches du plateau (*`PatchManager`*).

## Automa

Pour ajouter l'extension automa on a dû interfacer les joueurs pour qu'ils soient manipulable dans le `TrackBoard`. Le `Player` d'avant devient `HumanPlayer`. `HumanPlayer` et `Automa` interfacent donc `Player`.



## Vue

La vue en affichage graphique a été ajoutée. Il suffit de spécifier en argument du jeu -c ou -g pour choisir de passer au contrôleur l'interface du terminal ou graphique. Clarifier et séparer la construction des éléments de la vue, certaines classes intermédiaire permettant de construire des objets graphique ont été implémentés, pour le TrackBoard dont la forme en spirale est assez complexe, ou bien les patches du plateau.

## Schéma synthétique de l'architecture

Voici un diagramme UML assez synthétique, représentant dans les grandes lignes notre implantation.

