

Clément DELAMARE  
Paul GOUBARD--LANGERS  
Damien LEVEZIEL  
Pierre-Alexandre ROUSSEAU

Année 2022/2023

# Méthode de Conception

## Rapport De Projet

### Jeux de cartes, Blackjack

# Sommaire

## **1. Présentation du projet**

- a. Introduction
- b. Notre version du BlackJack
- c. Critères d'évaluation

## **2. Analyse du sujet**

- a. Particularités et consignes

## **3. Notre rendu**

- a. Solutions apportées

## **4. conclusion**

## **5. annexes**

# 1. Présentation du projet

## a. Introduction

Le projet vise à réaliser une manière de jouer au Blackjack. Nous devons développer l'application en java avec l'aide de la bibliothèque javafx en se basant sur le patron de conception MVC. Il doit y être intégré un moyen de jouer contre une intelligence artificielle.

## b. Notre version du BlackJack

Nous avons le choix quant à la version des règles du BlackJack pour lesquelles nous souhaitons opter (que ce soit l'original ou l'américain). De ce fait, voici les notre :

- Les cartes valent des points. Pour presque toutes, ce sont leurs valeurs numériques (le 4 de pique vaut 4, le 7 de trèfle vaut 7), à l'exception des têtes (valet, dame, roi) qui valent 10 et de l'as qui vaut 11 avant de totaliser plus de 11 en score total et vaut 1 après.
- Les joueurs choisissent tour par tour s'ils piochent ou s'ils passent, et ont le droit de piocher jusqu'à la fin du paquet.
- Un joueur peut s'arrêter de piocher à tout moment et obtient alors le score du total de ses cartes.
- Lorsque tout le monde arrête de piocher on procède au décompte des points.
- Le joueur se rapprochant le plus ou égalant 21 gagne la partie. Les joueurs excédant ce score perdent. En cas d'égalité, personne ne gagne.

### **c. Critères d'évaluation**

Nous devons travailler à 4 sur le projet et suivre les principes de base du projet de groupe : C'est-à-dire un code clair, lisible et commenté avec des tests mis à disposition. En plus de cela, nous devons implémenter ce que nous avons vu durant le semestre comme l'utilisation des patrons de conception (le MVC étant au cœur de l'application). Pour finir nous devons aussi créer un robot pour se mesurer au joueur

## **2. Analyse technique du sujet**

### **a. Particularités et consignes**

Afin de mener à bien notre projet, nous avons des consignes à respecter. La première étant de réaliser celui-ci en MVC (Modèle-Vue-Contrôleur). Le MVC est un patron de conception destiné aux applications graphiques. En effet, il apporte une organisation claire du code et sépare les composants : Nous avons le modèle qui gère les données, de la vue qui s'occupe de la partie visuelle et du Contrôleur qui gère les actions de l'utilisateur et synchronise le tout.

La deuxième consigne importante était de respecter les méthodes de conception vues en cours, notamment les patrons. Nous avons donc utilisé nos connaissances acquises en cours pour mener à bien le projet, et c'est ainsi qu'on a pu mettre en place une liste de patron pour le projet de blackjack.

## **3. Notre rendu**

### **a. Solutions apportées**

Pour les différents problèmes, différentes solutions :

- Template Method

Un humain, un joueur et un croupier sont des joueurs différents, mais restent des joueurs avec un paquet de cartes. Afin de ne pas faire de code redondant, on utilise une classe Joueur qui va se faire hériter des différents types de joueurs, qu'on utilisera plus tard

- Factory Method

Un jeu de blackjack peut être joué avec un paquet de 52 ou 32 cartes. Pour créer ce paquet de cartes, on utilise donc le Factory Method, qui renvoie un paquet de cartes selon la classe utilisée.

- State Method

Une mise est un état, et renvoie une stratégie différente. Ici, elle renvoie un gain et un nom différent, selon l'état de la mise du joueur une fois la partie terminée.

- Prototype

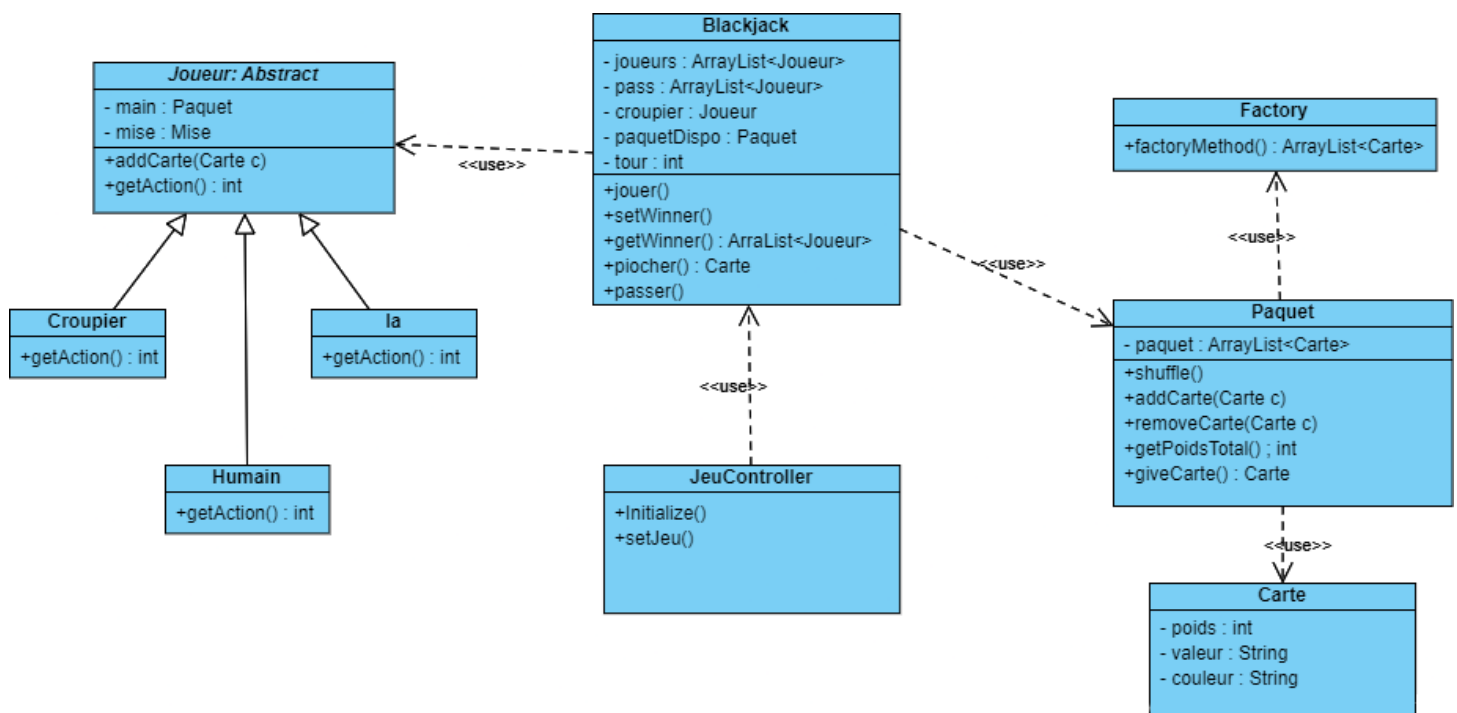
L'IA fait des prototypes de paquets pour regarder pour chaque carte si elle ne dépasse pas 21. Elle fait donc un clone de son paquet pour regarder le résultat du poids total du paquet avec la carte que l'on regarde.

## 4. Conclusion

Pour conclure, le projet nous a permis de comprendre plus en détail et de mettre en place plusieurs patrons de conception. Avec la réalisation d'une interface graphique nous avons également pu en apprendre plus sur la librairie graphique JavaFX et le pattern MVC. Ce projet nous a également permis d'en apprendre plus sur le côté "gestion de projet" et la mise en place d'un fichier ANT.

## 5. Annexes

### Diagramme de classes



# Une partie qui commence

Nombre de carte: 42

Pique

Trefle

Dame

Trois

Trefle

Carreau

Deux

Dix

Coeur

Trefle

Six

Neuf

Pique

Coeur

Neuf

Deux

Coeur

Carreau

Neuf

Roi

Joueur 1

Valeur: 12

Joueur 2

Valeur: 15

Joueur 3

Valeur: 11

Joueur 4

Valeur: 19

Tour du joueur numero 1

Piocher

Passer

Nouvelle partie

Retour

# Une partie terminée

Nombre de carte: 38

Partie terminee !

Coeur

Trefle

Coeur

Pique

Deux

Quatre

Roi

Deux

Croupier

Valeur: 18

Perdu

Coeur

Carreau

As

Neuf

Carreau

Carreau

Roi

As

Carreau

Pique

Cinq

Valet

Carreau

Trefle

Six

Dix

Coeur

Pique

Valet

Sept

Joueur 1

Valeur: 20

Gagne

Joueur 2

Valeur: 21

BlackJack

Joueur 3

Valeur: 25

Perdu

Joueur 4

Valeur: 23

Perdu

Tour du joueur numero 1

Piocher

Passer

Nouvelle partie

Retour