

## LICENCE 3 INFORMATIQUE

---

# Projet de conception Logicielle Blackjack

---

*Auteur :*

DIALLO ELHADJ ALSEINY  
DJIGUINE MAMADY  
DIALLO ABDOULAYE DJIBRIL  
SOKHONO MALICK

*Chargé du cours :*

MATHET YANN

*Encadrant TP :*

YOHANN JACQUIER

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Le Jeu Blackjack</b>	<b>2</b>
2.1	Généralité . . . . .	2
2.2	But du jeu . . . . .	2
2.3	Notre Blackjack . . . . .	2
2.3.1	Principe : . . . . .	2
2.3.2	Début du jeu : . . . . .	2
2.3.3	Resultat : . . . . .	3
2.3.4	Fin du jeu . . . . .	3
2.4	Organisation du projet . . . . .	3
2.4.1	Répartition des taches : . . . . .	3
2.4.2	Arborescence du projet : . . . . .	4
2.4.3	Architecture du projet : . . . . .	4
2.5	Éléments techniques : . . . . .	5
2.5.1	La librairie Carte . . . . .	5
2.5.2	Hand : . . . . .	5
2.5.3	Players : . . . . .	5
2.5.4	HumanPlayer : . . . . .	5
2.5.5	Robot : . . . . .	5
2.6	Configuartion : . . . . .	5
<b>3</b>	<b>Expérimentations et Usages :</b>	<b>6</b>
3.1	Utilisation : . . . . .	6
3.2	Lancement de l'application : . . . . .	6
3.3	Fonctionnement en mode console . . . . .	6
3.4	Fonctionnement de l'interface graphique : . . . . .	6
3.5	Test de l'application : . . . . .	8
<b>4</b>	<b>Conclusion</b>	<b>8</b>
4.1	Bilan de nos travaux : . . . . .	8
4.2	Améliorations Possibles : . . . . .	8

# 1 Introduction

La programmation orientée objet est un paradigme de plus en plus indispensable pour toute personne aspirant à faire du développement. Ainsi afin de bien l'assimiler et se parfaire en appliquant ses principes notamment en java, un projet nous a été donné dans le cadre de l'unité d'enseignement Conception logicielle. Le but de ce projet est de réaliser une application de jeu blackjack par groupe de quatre (4) étudiants en utilisant quelques designs patterns.

- D'une part, une librairie « cartes » permettant de créer et manipuler des cartes, pouvant être utilisée par d'autres applications le jeu soit exécutable en mode console ;
- D'autre part, un jeu de « blackjack » s'appuyant sur cette librairie
- ET en fin La librairie et le jeu s'appuieront sur MVC, c'est-à-dire proposeront notamment un découpage clair entre modèle et vue(s).

## 2 Le Jeu Blackjack

### 2.1 Généralité

Le blackjack est un jeu de carte populaire dans les casinos physiques et les casinos virtuels. L'origine du jeu est incertaine. Le premier jeu semblable au Blackjack moderne est le jeu du « trente et un », dont le but était d'obtenir 31 points. Ce jeu apparaît au XIV<sup>e</sup> siècle<sup>1</sup>. Le Blackjack moderne fait son apparition à la fin du XVII<sup>e</sup> siècle ou du XVIII<sup>e</sup> siècle sous le nom de « 21 » aux États-Unis. Pour essayer d'attirer les joueurs, les casinoiers

### 2.2 But du jeu

La partie oppose individuellement chaque joueur à la banque. Le but est de battre le croupier sans dépasser le score de 21. Dès qu'un joueur fait plus que 21, on dit qu'il « brûle » et il perd sa mise initiale.

### 2.3 Notre Blackjack

Notre application de jeu blackjack est un cas particulier du blackjack jouable en console mais aussi en mode graphique avec des options de paramétrages côté graphique permettant d'ajouter des joueurs avec leurs mises, mélanger et couper un paquet.

#### 2.3.1 Principe :

Au début, les mises sont toujours définies en ajoutant les joueurs. La mise minimale est le montant minimum qu'un joueur doit parier pour jouer, 5 dans notre blackjack, et la mise maximale est le montant maximal que le joueur peut parier, 100 dans le notre.

**NB : On a initialisé la banque de chaque joueur à 100 au début du jeu.**

Les valeurs des cartes sont les suivantes :

- **de 2 à 9** : valeur nominale de la carte ;
- **de 10 au roi (surnommées « bûche »)** : 10 points ;
- **As** : 1 ou 11 points (au choix du joueur).

#### 2.3.2 Début du jeu :

Au début de la partie, le croupier distribue une carte face visible à chaque joueur et une carte face visible pour lui. Il tire ensuite pour chaque joueur une seconde carte face visible et une seconde carte face cachée pour lui. Les joueurs débutent donc la partie avec deux cartes visibles.

Une fois les premières cartes distribuées, le croupier demande au premier joueur de la table (joueur situé à sa gauche) l'option qu'il désire choisir. Le joueur a plusieurs options selon la situation : Si le

joueur a un Blackjack, il n'a aucun choix à faire. Le joueur attend donc l'annonce des résultats. Si le joueur n'a pas de Blackjack, plusieurs choix sont possibles :

- Demander une ou plusieurs cartes supplémentaires, afin de se rapprocher de 21, sans le dépasser. Le joueur annonce alors « Carte » par clic sur tirer une carte , on dit que le joueur « Hit », c'est-à-dire qu'il tire) . **Si toute fois il dépasse 21 ,la main est automatiquement passée au prochain joueur**
- S'arrêter ou « rester » (en anglais, on dit que le joueur « Stand »), et donc conserver ses cartes. Le joueur clic sur passer la main, pour indiquer son intention au croupier ;

Une fois tous les joueurs servis, le croupier joue pour son compte selon une règle universelle : « le croupier tire à 16, reste à 17 ». Ainsi, le croupier tire des cartes jusqu'à atteindre un nombre supérieur ou égal à 17.

Dans le cas où le croupier a un « 17 soft », c'est-à-dire un As et un six (valant soit 7 points, soit 17 points), la règle la plus commune est le « stand on soft 17 ». Le croupier s'arrête donc à 17

### 2.3.3 Resultat :

Une fois le croupier servi, il annonce les résultats pour chaque joueur. Les joueurs ayant dépassé 21 points perdent leur mise dans tous les cas.

- Si le croupier a dépassé les 21 points, tous les joueurs encore dans le jeu (et donc n'ayant pas dépassé les 21 points) gagnent une fois leur mise.
- Si le croupier a 21 points (sans Black jack) ou moins :
  - Les joueurs ayant moins de points que le croupier perdent leur mise ;
  - Les joueurs ayant autant de points que le croupier récupèrent simplement leur mise. Dans cette situation, on dit qu'il y a « push »
  - Les joueurs ayant plus de points que le croupier gagnent une fois leur mise.

Après chaque annonce de résultat des joueurs, vous pouvez décider de continuer de jouer ou par clic sur rejouer à gauche de l'interface. A noter aussi que dès qu'un joueur ne dispose plus d'argent dans sa banque afficher en dessous de son panneau (voir l'interface), il est automatique retirer pour le prochain tour si toute fois on clic sur rejouer .

### 2.3.4 Fin du jeu

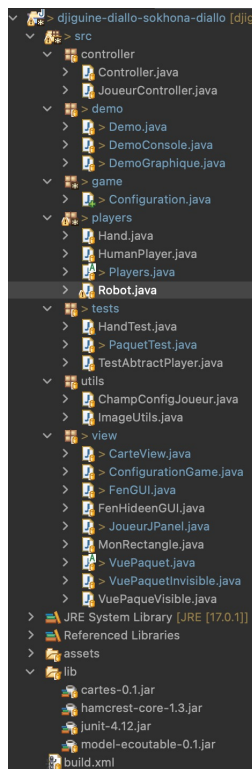
Le jeu prends fin lorsque tous les joueurs n'auront plus aucun montant dans leur banque (c'est à dire lorsqu'ils auront leur banque égale ou en dessous de 0 ) ou quand ils décident d'arrêter tous ensembles.

## 2.4 Organisation du projet

### 2.4.1 Répartition des taches :

Le projet étant essentiellement composé de deux grandes parties, nous nous sommes dans un premier temps réunis afin de mieux comprendre le principe du jeu et élaborer un plan de travail, ce qui nous a permis de comprendre différentes façons d'implémenter ce jeu. ELhadji Alseiny et Mamady djiguiné se sont chargé de développer la librairie carte , la partie modèle du jeu et son exécution en console, Abdoulaye Djibril Diallo et Malick Sokhona se sont occupés de la partie interface graphique.

## 2.4.2 Arborescence du projet :



## 2.4.3 Architecture du projet :

Concernant l'architecture de notre projet, nous avons voulu la relier au sujet pour une compréhension facile en utilisant quelques notions au cours de ce semestre.

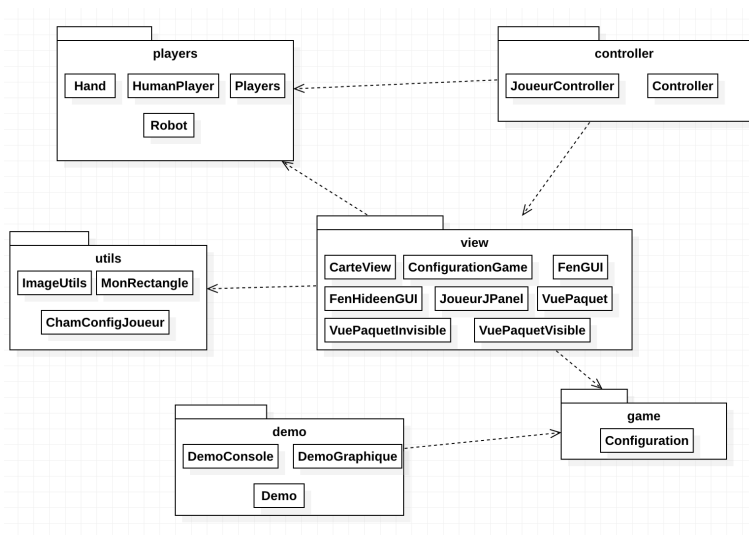


FIGURE 1 – Diagramme de packages

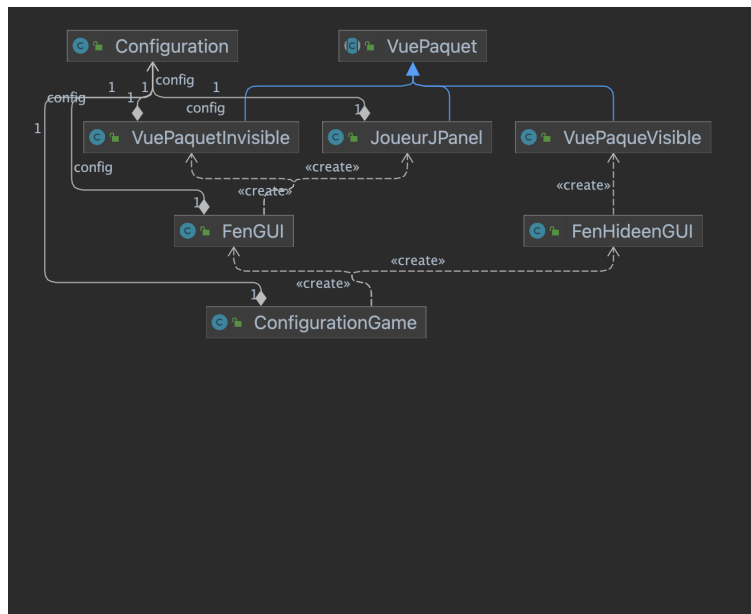


FIGURE 2 – Diagramme de Classes

## 2.5 Éléments techniques :

### 2.5.1 La librairie Carte

Permet de créer et manipuler des cartes à travers méthode qui y sont définies. Elle pourrait aussi être utilisée par d'autres applications. Cette librairie en plus de la notion de carte intègre celle de paquet pour représenter la main d'un joueur et de manipuler ces derniers (Ajout ,retrait ...). La librairie contient permettant de créer des paquets de 32 ou 52 cartes (une classe avec des méthodes statiques).

### 2.5.2 Hand :

- Est une classe la main d'un joueur et contient en son sein des méthodes telles que :
- addCardToHand : s'appuie sur le paquet pour ajouter une carte à la main d'un joueur.
  - gethandvalue : Permet d'évaluer la main d'un joueur courant au cours du jeu .

### 2.5.3 Players :

- Est une classe abstraite permettant de manipuler les joueurs contient en son sein une méthode :
- drawCard : qui s'appuie sur la classe Hand pour ajouter une carte et à la main d'un joueur.

### 2.5.4 HumanPlayer :

Est une représentation d'un joueur se mesurant au dealer .il va disposer d'une bank et pourra parier au tant de fois qu'il dispose de l'argent dans sa bank. Nous avons décidé d'initialiser la bank de tout joueur à 100. Il ne peut parier en dessous 5 et au delà du montant totale de sa bank.

### 2.5.5 Robot :

- Représente le dealer et contient une méthode particulière en son sein :
- PrintFirsthand : Qui, permet d'afficher la carte les premières carte du dealer tout en cachant la première.

## 2.6 Configuration :

Représente notre blackjack proprement dite. Elle permet d'ajouter toutes les configurations nécessaires dispose en son sein des méthodes permettant faire fonctionner le jeu.

## 3 Expérimentations et Usages :

### 3.1 Utilisation :

Nous avons développé notre application avec du java standard Édition 11 pour des raisons de compatibilité avec les ordinateurs de l'université. Elle peut s'exécuter sur les distributions Linux avec interface graphique telle que : Ubuntu version 20.04.4 LTS, sur mac Os 12.3.1 et windows 10.

### 3.2 Lancement de l'application :

Vous devez ouvrir un terminal à la racine du projet . Pour :

- Lancer l'application exécuter : `ant run`

Une fois la commande **ant run** exécutée, il vous sera demandé de choisir un mode :

- **1** : pour jouer en mode graphique
- **2** : pour jouer en console
- **3** : pour quitter .

```
+----- MENU -----+
+                     +
+ 1 - pour jouer en mode graphique +
+-----+-----+
+ 2 - pour jouer en mode console +
+-----+-----+
+ 3 - pour quitter +
+-----+-----+
Veuillez entrer votre choix(1..3) 2
Vous êtes en mode console
```

FIGURE 3 – Menu

- Initialiser le projet : `ant init` Cette initialisation créera les dossiers de bases bin, doc et dist
- Compiler le projet : `ant compile`
- Générer le Javadoc : `ant javadoc`
- Générer le fichier jar : `ant packaging`
- Nettoyer le projet : `ant clean`
- Lancer le test : `ant test`

### 3.3 Fonctionnement en mode console

En mode console vous devez entrer le nombre de joueur au préalable .Puis pour chaque joueur renseigner son et sa mise .Ces indications sont données par des séries de questions sur la console. Une fois les informations des joueurs saisies,deux sont distribuées à chacun des joueurs et un affichage de la de leurs mains y suit. En ce moment pour chaque joueur il lui sera demandé s'il veut tirer une carte ou passer la main : Il doit répondre Par O pour oui ou N pour non. **Il peut arriver que le joueur dépasse le chiffre 21 après avoir décidé de tirer une carte ; en ce moment la main est automatiquement passé au joueur suivant**

**Attention a bien suivre les indications sur la console pour mieux voir ce qui se passe.**

### 3.4 Fonctionnement de l'interface graphique :

Au lancement du programme, il se présente avec une fenêtre vous demandant le nombre de joueurs maximum quatre dans notre blackjack. Après avoir saisi le nombre de joueur et cliquer sur valider : une nouvelle fenêtre se présente demandant de saisir le nom de chaque joueur avec sa mise (entre 5 et 100) voir ci-dessous :

**Configuration du jeu**

Nombre de joueurs: 1-4

Nom du joueur 1:	Abdoulaye	la mise: 5-100	<input type="text" value="100"/>
Nom du joueur 2:	Mamady	la mise: 5-100	<input type="text" value="100"/>
Nom du joueur 3:	Elhadj	la mise: 5-100	<input type="text" value="100"/>
Nom du joueur 4:	Malick	la mise: 5-100	<input type="text" value="100"/>

FIGURE 4 – Configuration

**NB :** Vous devez juste renseigner le nom de chaque joueur ainsi que sa mise, puis après confirmer par clic sur le bouton **Confirmer**

Après avoir confirmé vous êtes dirigé sur l'interface principale du jeu du jeu (**voir ci-dessous**)

**BlackJack**

**Croupier: 9**

Mélanger  
Couper  
Distribuer  
Rejouer

Tirer une carte Passer la main

Abdoulaye: 14 Bank: 100.0 Mise: 100.0  
Mamady: 18 Bank: 100.0 Mise: 100.0  
Elhadj: 16 Bank: 100.0 Mise: 100.0  
Malick: 20 Bank: 100.0 Mise: 100.0

FIGURE 5 – Interface du jeu

Les labels :

- **Mélanger** : Permet de mélanger le paquet
- **Couper** : Pour couper 5 ,10 ou 25 cartes.
- **Distribuer** : Pour distribuer les deux premières cartes à chacun des joueurs. Il ne faut le faire qu'une seule fois et en début de chaque partie. Retenez que le paquet sera mélanger si toute fois vous cliquez directement sur distribué sans avoir mélange.



- **Rejouer** : Permet de rejouer après l'annonce du résultat par le croupier .A ce niveau aussi si vous décidez de refaire une nouvelle partie vous devez juste saisir que la mise des joueurs et confirmer.

Un carré blanc indique le joueur à qui le tour de jouer : Chaque panel en dessous contient le nom d'un joueur ,sa mise ,sa banque,la valeur de main et d'autres éventuelles informations se produisant avant et après le jeu. Un joueur courant peut tirer une carte par clic sur le label **tirer une carte** ou de passer la main au joueur suivant par clic sur **passer la main** .Si toute fois un joueur courant dépasse 21 un label s'affiche sur sa zone lui indiquant automatiquement qu'il a perdu sa mise et la main est passée au prochain joueur voir ci-dessus.

### 3.5 Test de l'application :

Nous avons réalisé le test de notre application en utilisant le framework open source Junit-4.12 pour le développement et l'exécution de tests unitaires. Elle teste l'ensemble des méthodes pertinentes du modèle.

## 4 Conclusion

### 4.1 Bilan de nos travaux :

Travailler un sujet aussi passionnant que le blackjack nous a appris beaucoup de choses dans ce vaste domaine qui est la conception de jeu allant d'une expérience de modélisation des différentes étapes de développement en passant par des exercices de programmations en Java. Il était très agréable de pouvoir mettre en pratique les enseignements vues durant tout le semestre en programmation orientée objet. Nous retenons également de ce projet, la nécessité d'une bonne préparation, l'importance de la recherche et la gestion du temps pour la réalisation d'un programme.

### 4.2 Améliorations Possibles :

- Nous pouvons continuer ce projet de nombreuses façons :
- On aurait pu intégrer une version plus complexe intégrant les splits et les assurances.
  - On aurait pu faire en sorte de pouvoir ajouter un nouveau joueur après chaque partie .

## Références

- [1] Wikipedia <https://fr.wikipedia.org/wiki/Blackjack>
- [2] Java Platform Standard Edition 8 Documentation <https://docs.oracle.com/javase/8/docs/>