



Independant

Année de Céation du Projet  
2021

PROJET  
BLACKJACK

---

Auteur 1 : POREE Baptiste



# TABLE DES MATIÈRES

<b>I</b>	<b>Introduction</b>	<b>1</b>
<b>II</b>	<b>Base</b>	<b>3</b>
<b>1</b>	<b>Carte</b>	<b>4</b>
1.1	idée Carte . . . . .	4
1.1.1	Exemple de type . . . . .	4
1.2	idée Packet . . . . .	5

# INTRODUCTION

# INTRODUCTION

Dans le cadre de ce projets le **code** ainsi que le **pseudo-code** sont juste là pour aider les personnes a comprendre l'idée général du projet ainsi de ca Structure.

La structure dans ce dossier sera sous cet forme :

- Répertoire avec le *type de langage*, contenant les information suivant :
  - Répertoire *src* contenant le code commenté.
  - Répertoire *doc* contenant le doc générer par le code commenté quand le langage peut le faire.
  - Répertoire *dist* contenant l'exécutable ou la librairie.
  - Répertoire *test* contenant les test et le buildTest.
- Ficher *build.xml* contenant l'automatisation des information du code.
- Ficher *README.md* contenant le mode emplois du code.

BASE

# 1) CARTE

## 1.1: idée Carte

On souhaite réaliser un black Jack donc on aura besoin de fait un jeux de carte qui peu être utiliser dans d'autre application donc il faudra faire un librairie externe si le langage, le permet.

On souhaite que cet librairie peut crée et manipuler des carte de jeux classique comme plus spécifique.



### 1.1.1: Exemple de type

Pour se faire on fera sous la formes d'une *Map* avec des modelé peu être pour les key par exemple :

- hauteur
- couleur
- faceCacher
- description
- type
- img
- ...

---

**Algorithme 1** : base carte

---

**Résultat** : modifi l'insance pour les carte

1 Carte  $\leftarrow$  Map<key 'Str' : valeur 'Str'>

---

Pour se faire il faut que la librairie peu se tester lui même, avec des méthode comme :

- `isExisteKey(key)` : vérifi que la key existe bien dans la Map
- `getCarte()` : retourne la Map
- `getItmeCarte(key)` : retourne la valeur qui est en lien avec la key dans la Map Carte
- `upgradeCarte(key, valeur)` : fait fait un teste pour modifier la valeur de la key exsitanse

- `supprime()` : fait un clean la map
- `supprimeItem(key)` : fait un clean de la key dans la Map Carte

## 1.2: idée Packet

On peut voir le *Packet* comme une liste de *Carte* et aussi comme une liste de liste qui peut représenter des *pioche*, *défauce*, *joueur*, ....