

# Rapport final du projet de D  v informatique avanc   : application

## Cahier des Charges

### ##### Probl  matique

Le client aimerait avoir une application pour jouer    Danish (jeu de carte) en multijoueur.

### ##### Objectif

Pouvoir lancer une partie compl  te de Danish et que les r  gles soient automatiquement respect  es.

### ##### Contraintes techniques

Gestion des cartes, connexion entre les joueurs, respecter les r  gles

### ##### Description

Minimum 2 joueurs, maximum 5 joueurs.

Lancement de partie, chaque joueur re  oit 3 cartes cach  es pos  es sur la table, 3 cartes r  v  l  es sur les cartes cach  es et 3 cartes en mains.

Chacun joue    son tour, il peut soit jouer et prendre une carte dans la pioche soit passer et r  cup  rer le tas jou  .

Une carte ne peut   tre jou  e que si sup  rieur    la derni  re carte jou  e.

Certaines cartes ont des r  gles sp  ciales. (2, 3, 7, 8, 10, As).

(Possibilit   d'avoir un mode de jeu o   la triche est autoris  e.)

### ##### Users Stories

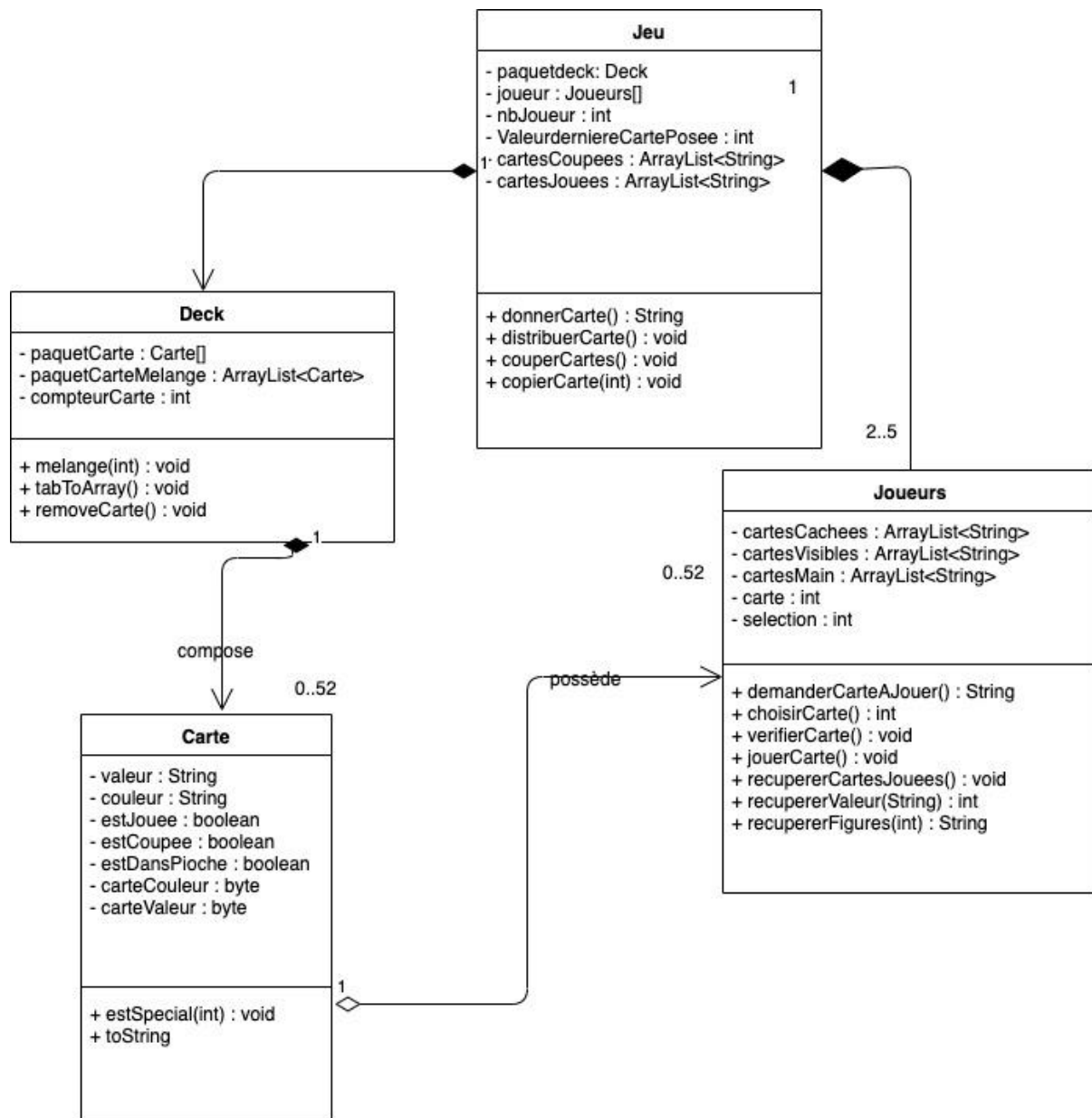
En tant qu'utilisateur, j'aimerais jouer au Danish avec au moins 1 autre personne et maximum 4 autres personnes afin d'avoir des interactions.

En tant que joueur, je veux pouvoir d  marrer ma partie et mettre pause afin de pouvoir jouer quand il m'est possible de jouer.

En tant que joueur, j'aimerais que les r  gles soient respect  es correctement afin d'avoir la meilleure exp  rience de jeu.

En tant que joueur, je veux que le paquet de carte soit m  lang      chaque d  but de partie afin de faire des parties diff  rentes    chaque fois.

## Diagramme UML



## Choix d'implémentations

Nous avons choisi l'implémentation par MVC avec des sous package pour le modèle, le contrôleur et la vue.

Lors de la réalisation de notre projet, nous avons pris de mauvais choix au début ce qui nous a rendu fortement en retard. Notre projet n'a donc pas pu être entièrement fonctionnel comme nous le voulions.

Malgré cela nous avons pu mettre en place des threads, des ArrayList et une partie du MVC, nous voulions pouvoir implémenter un mode multijoueur avec un SocketServer qui a fortement changer notre code et qui ne faisait pas fonctionner notre jeu comme on le voulait.

## Difficultés rencontrées.

Lors de notre projet, nous avons rencontrés plusieurs difficultés telles que la disparition d'un des membres du groupe et des erreurs de choix qui ont payés cher sur le timing de production.

## Piste d'amélioration

Pour améliorer notre projet, nous voulions permettre aux joueurs de tricher en mettant plusieurs cartes à la fois.

Cependant, nous avons eu beaucoup de retard sur les fonctions principales de notre application qui ne sont, elles, pas toutes finies.

## Conclusions individuelles

### Simon Mages

Ce projet fut pour moi très instructif, le fait de ne pas suivre à la lettre un énoncé mais au contraire de devoir moi-même réfléchir à comment réaliser ce projet, comment implémenté toutes les fonctionnalités voulues m'a permis de bien mieux apprendre ce langage.

### Humbert Meyers

Ce projet fut intéressant pour moi car je ne connaissais pas la manière de travail de mon groupe au départ. Et j'ai pu approfondir mes connaissances dans le langage Java. Cependant je trouve dommage qu'on ne soit pas arrivé au bout du projet.