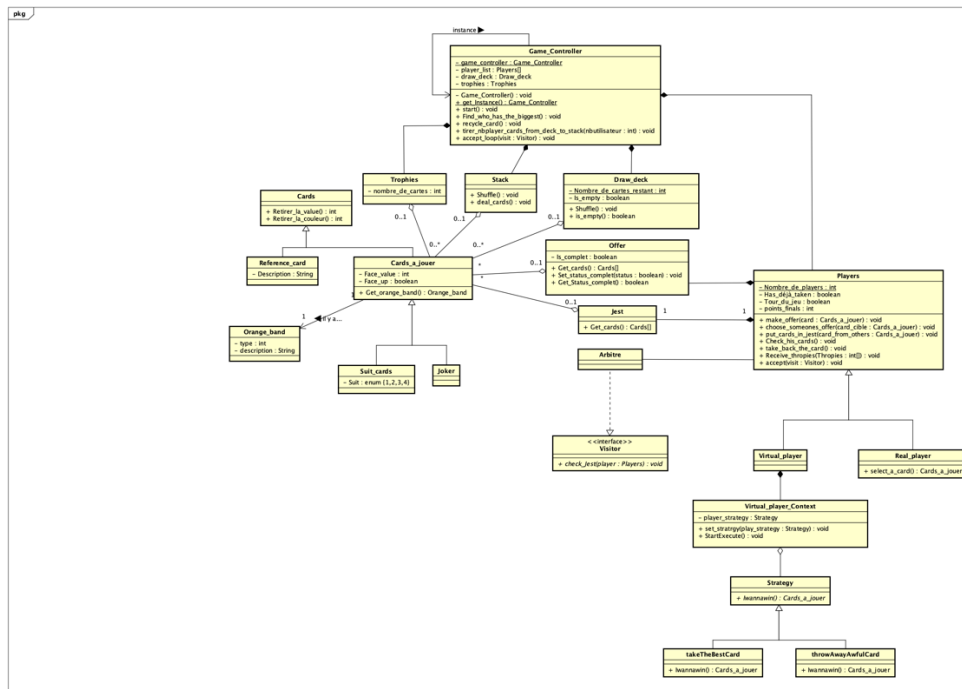


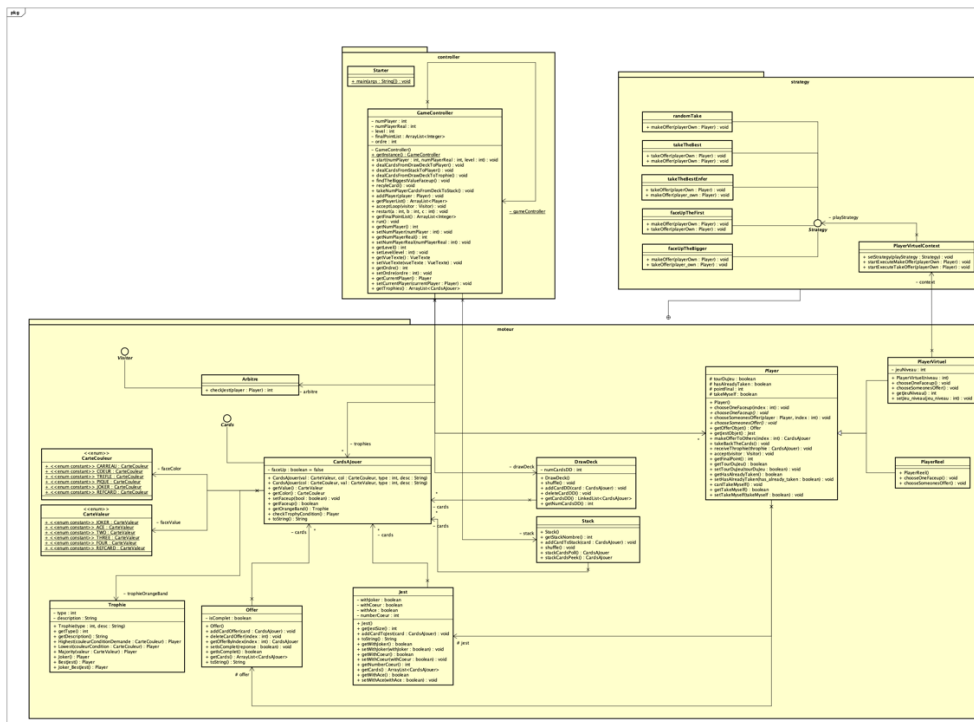
Le diagramme de classes

Initial :



La photo de la taille d'originale se situe dans /Diagrammes.

Final :



La photo de la taille d'originale se situe dans /Diagrammes.

Ce que l'on a modifié :

1. La relation entre la classe `PlayerReel` et `PlayerVirtuel` n'était pas juste, cette erreur ne peut être vue dans le diagramme initial, mais c'est dans le code d'origine : en effet, avant le 2^{ème} jalon, le joueur virtuel héritait du joueur réel, et il redéfinissait les méthodes, ça ne respecte pas du concept de l'orienté objet. Maintenant on l'a corrigé et c'est juste.
2. Dans `Trophie.java` on a corrigé un problème. On avait utilisé thread pour réaliser la détection de condition de trophées, c'était absolument inutile et n'avait pas de sens.
3. Dans `Strategy.java`, il y avait justement une méthode qui permet de réaliser deux fonctions : choisir une carte face-up comme l'offre, et également prendre le choix quand le joueur doit choisir une offre de quelqu'un d'autre, c'est pas correct, donc après on a séparé les deux fonctions : `makeOffer()` et `takeOffer()`.
4. On a également ajouté des attributs et des méthodes nécessaires qui ne sont pas été prise en considération ni pendant la discussion ni dans le développement du code de la première phase. Par exemple des getters et des setters.

L'état actuel

Le fonctionnement complet du jeu est réalisé, un jeu peut être configuré avec 3 ou 4 joueurs, ils peuvent être virtuels ou réels, et il y a 3 niveaux de difficultés : facile, difficile et enfer. Dans le mode « enfer », le(s) joueur(s) virtuel(s) a(ont) le droit de tricher. Malgré que ça ne respecte pas bien la règle du jeu, il augment bien la difficulté.

Ce projet est intégré dans une interface graphique. Le jeu reste jouable à la fois en ligne de commandes et par le biais de l'interface graphique, bien sûr au sein d'une même partie. Mais quant au patron de conception MVC, on n'a pas bien séparé la vue texte et le modèle, il reste quelque ligne de « `System.out.println...` » dans le modèle. C'est le point que l'on n'a pas parfaitement fait.

Selon le cahier des charges, c'est l'état actuel précise :

- ✓ • L'application devra permettre de jouer à plusieurs joueurs qui seront des joueurs physiques auxquels s'ajoutent des joueurs virtuels. Dans le cas d'un joueur virtuel, le jeu devra lui attribuer une stratégie de jeu simple mais cohérente qui lui permettra de déterminer action mener jouer à chaque instant.
- ✓ • L'ensemble de l'application sera intégré dans une interface graphique. On pourra utiliser des images telles que celles données en annexe.
- ✓ • Avant la conception et le développement de l'interface graphique, une interface rudimentaire en ligne de commande permettra de tester le moteur du jeu. Cette interface devra être conservée lors de l'évaluation fonctionnelle et la remise des fichiers.
- ✗ • On intégrera une notion d'extension qui correspondent à des nouvelles cartes, qu'il vous faudra concevoir et implémenter. En début de partie, il sera demandé au joueur s'il souhaite intégrer ces cartes d'extension ou pas à la partie qui va débiter.
- ✗ • On intégrera enfin une notion de variantes dans les règles du jeu. En début de partie, le choix d'une variante sera proposé au joueur et c'est la variante choisie qui imposera les règles d'une partie. Il est demandé de concevoir et implémenter deux variantes en plus des règles de base.
- ✓ • L'architecture retenue devra veiller au respect des règles de la conception orientée objet. L'architecture devra ainsi être (1) modulaire en identifiant des composants indépendants liés entre eux par des relations (2) extensible en permettant le changement de règles exposé ci-avant ou l'ajout de cartes avec de nouvelles fonctions.