

PRÉSENTATION DU PROJET de POO

Consigne principale

Programmer en POO un jeu de cartes, plus précisément le jeu de cartes "Bang".
À rendre pour le 9 mai 2021 (à priori).

Règles du jeu de "Bang"

disponible à l'adresse bang.zanapher.fr

Il s'agit d'un jeu à 4 joueurs minimum, jusqu'à 7+.

On a des cartes rôles, sherif, adjoints, hors la loi, renégat répartis entre les joueurs qui cachent leur rôle (sauf le sherif).

Le sherif et les adjoints doivent tuer tous les hors la loi.

Les hors la loi doivent tuer le sherif.

Le renégat doit tuer le sherif en face à face, lorsqu'il sont les derniers en vie.

En plus des cartes rôles, chacun a un personnage au début de la partie. Un nom, un nombre de points de vie ainsi qu'une règle spéciale qui ne s'applique qu'à eux. Les cartes personnage sont face visible.

La partie va se jouer avec le restant des cartes.

Les personnages sont des Personnages.

Les rôles sont des rôles.

Le restant des cartes sont des Cartes.

On a une pile de carte de pioche et une pile de défausse.

Les joueurs peuvent mettre des cartes devant eux (comme de l'équipement ou des objets).

Initialement, on distribue autant de cartes aux joueurs que leur personnage a de points de vie.

Chacun son tour (en commençant par le sherif), les joueurs vont jouer autant de cartes qu'ils le souhaitent tant qu'ils le peuvent, ou alors elles vont en poser devant elles.

Deux types de cartes :

Les cartes oranges :

Carte action : Elle font une action, on la joue, on la défausse.

Les cartes bleues :

Carte effet : On la laisse poser devant le joueur, il s'agit d'équipement ou alors de cartes qui ne s'activent que sous certaines conditions.

Moins on a de point de vie, moins on peut garder de cartes dans sa main. On est obligé de défausser les cartes inutilisées.

Quelques concepts :

Les joueurs à ma gauche ou à ma droite sont à distance 1 de moi. Leurs voisins sont à distance 2, etc. Cette notion de distance permet l'action de certaines cartes.

Les cartes possèdent une valeur style carte à jouer (du 2 à l'as, carreau, trèfle, pique, cœur) en bas à gauche. Cela ne sert presque jamais, sauf lorsqu'il y a une notion de tirage aléatoire (Si vous tirez une carte cœur, un effet se produit...).

Si je possède la carte BANG, je peux choisir un joueur à portée (qui est de distance 1 par défaut), et lui tirer dessus. Soit il perd un point de vie, soit il défausse une carte RATEÉ.

Il existe 19 types de cartes dans le jeu, et elles sont disponibles en grand nombre.

Toutes les règles seront disponibles par mail plus tard dans la journée.

Attentes du projet en termes de Java

Il faudra fork la base du projet sur github, la plupart des classes sont présentes mais elles ne sont pas écrites.

La classe Game gère la partie complète (distribution des rôles, des personnages, des tours, de la pioche...)

La classe Player représente un seul joueur qui représente la main du joueur, le tapis du joueur, le rôle et le personnage du joueur, ses PV et ses fonctions pour jouer son tour.

La classe Card représente toutes les cartes à jouer, (les oranges ou les bleues). Elles sont représentées par des sous classes de BlueCard et OrangeCard, sous-classes de Card.

La classe BangCharacter représente les cartes personnages, et il va falloir check dans certains cas comment la partie se déroule selon la sous-classe qui a été attribuée au joueur.

Les classes StageCoach, GeneralStore et BlackJack seront implémentées dans notre dépôt pour donner un exemple.

On va créer une instance de Game avec le nom des joueurs. La partie va se dérouler par une succession des toString de Game. On voit toutes les informations sur l'écran, on ne cherche qu'à simuler les règles du jeu donc on affiche tout (dans la console ou dans l'interface gratuite).

On écrit le nom des cartes que l'on veut jouer dans l'entrée standard. Si l'on entre la chaîne vide, on passe son tour. Si le mode console fonctionne, le GUI devrait fonctionner automatiquement.

Les méthodes à compléter sont indiquées dans le code, elles ne sont pas implémentées et on nous explique ce qu'il faut faire.

Pour que tout fonctionne (l'interface console), nous allons devoir utiliser certaines méthodes prédéfinies pour communiquer avec le jeu. Il ne faudra utiliser que ces fonctions pour les interactions entre le joueur et le système.

Il faudrait qu'à chaque étape de développement, le jeu compile. On ne fait **JAMAIS** de commit qui ne compile pas.