



Documentation Technique

Spendor



Anaïs Aoucher

Ethan Dorothée

SOMMAIRE

STRUCTURE	2
DESCRIPTION DES CLASSES PRINCIPALES	2
DEROULEMENT DU JEU	3

Structure

Le projet est organisé en un seul package : splendor.

Structure des classes principales :

- **Main** : point d'entrée du programme
- **Game** : gère la boucle principale du jeu
- **Player** : représente un joueur
- **Board** : gère les cartes et les ressources
- **Action** : contient la logique des choix du joueur
- **PrintGame** : affiche les options textuelles
- **Stones** : énumération des types de pierres précieuses

Description des classes principales

- **Game**
 - **Rôle** : Lance et contrôle le déroulement de la partie.
 - **Attributs** :
 - Board board : Plateau de jeu
 - List<Player> players : Liste des joueurs
 - PrintGame printGame : Utilitaire d'affichage
 - Action action : Gestion des actions
 - **Méthodes** :
 - run() : boucle principale
 - playerTurn(Player) : gère le tour d'un joueur
- **Action**
 - **Rôle** : Gère les différentes actions possibles (3 choix)
 - **Méthodes** :
 - play(int choice, Player, Board) : exécute l'action choisie
 - one() : prendre 3 pierres différentes
 - two() : prendre 2 pierres identiques
 - three() : acheter une carte

- **Board**
 - **Rôle** : Contient les ressources et les cartes visibles
 - **Méthodes** :
 - selectTokens(Player, Stones, int) : sélectionne des pierres
 - selectCard(Player, Card) : effectue un achat
 - revealCards() : affiche les cartes disponibles
- **Player**
 - **Rôle** : Gère un joueur
 - **Attributs** :
 - String name
 - int points
 - Inventaire de pierres et cartes
 - **Méthodes** :
 - buyCard(Card) : achète une carte
 - getPoints() : retourne les points du joueur
- **PrintGame**
 - **Rôle** : affiche uniquement du **texte d'aide ou des options**
 - **Fonction** : affiche les choix d'actions disponibles au joueur

Déroulement du jeu

1. Main instancie les joueurs et crée un objet Game
 - Entrer le nombre de joueurs
 - Entrer un prénom et un âge pour chaque joueur
2. Game.run() lance une boucle de jeu :
 - Affiche le numéro de tour
 - Révèle les cartes
 - Chaque joueur joue à son tour
3. Chaque tour appelle playerTurn() :
 - Affiche les options avec PrintGame
 - Récupère un choix utilisateur
 - Action.play(...) applique le choix
4. Répétition jusqu'à arrêt