

# Rapport du projet en JAVA

## Index

- 1) Les différentes classes et héritages
- 2) Les choix d'implémentation
- 3) Les difficultés rencontrées

## **1 - Les différentes classes et héritages**

Premièrement nous avons conçu une classe "Fenêtre" dans laquelle nous avons défini notre interface graphique, ainsi qu'un "main" pour lancer le programme.

La classe Game permet de gérer l'enchaînement des cartes.

La classe Carte est une classe abstract contenant les accesseurs pour les différents attributs des cartes. Elle implémente une interface ICarte.

D'autres classes comme Bar, Bowling, Café... sont des classes filles de la classe Carte implémentant la méthode actionlancee() de l'interface ICarte.

Ces cartes sont générées à partir de fichier texte, pour cela nous avons créé une classe CarteFactory qui prend en paramètre le chemin du fichier texte et qui, grâce à de la réflexion génère des cartes à la volée. Ces cartes sont ensuite mises dans une pile de carte.

La classe pile carte permet de générer des cartes du matin de l'après-midi et du soir, et ainsi, contient l'ensemble des cartes utilisées au cours du jeu.

PileException est une classe fille d'Exception et permet de gérer les exceptions de la pile.

Il y a un dernier héritage ou "Pretendant" est la classe mère et les classes filles sont :

- dragueur
- geek
- Lourd
- gay

Ces 4 classes implémentent une interface IPretendant contenant les méthodes evolutionRelation(), rencontrePossible(), et evolutionRelation().

## **2 - Les choix d'implémentation**

Nous avons décidé de réaliser 3 interfaces "ICarte", "pile" et "IPretendant". Nous avons fait ce choix pour définir 3 comportements différents ceux de nos cartes, celui des prétendants et celui du moteur de jeu, c'est à dire la pile.

L'interface "ICarte" implémentent donc les classes suivantes :

- Bar
- Cinema
- Maison
- Piscine\_longueur
- Piscine
- Restaurant
- ...

L'interface "IPretendant" :

- drageur
- geek
- Lourd
- gay

L'interface "pile" :

- PileCarte

La classe Game s'occupe de la partie métier de l'application et déclenche l'affichage des cartes dans la classe Fenetre.

Nous avons fait le choix d'utiliser la réflexion permettant d'instancier une classe à partir de son nom au format string plutôt que de passer par un switch pour chaque carte et chaque prétendant.

## **3 - Les difficultés rencontrées**

Nous avons rencontré plusieurs difficultés :

-Générer une pile de carte avec alternance matin->après-midi->soir en gérant le fait de ne pas être à court de carte d'une de ses périodes.

- Générer les cartes d'un type particulier à la volée.