

# Pyramide des Ages

---

Zhixing CAO, Yuxiang LI

1<sup>er</sup> avril 2015

## 1 INTRODUCTION

Le projet consiste à développer une application sur Android en appliquant la programmation concurrente et réalisant éventuellement un environnement serveur-client. Le jeu Set est un jeu de cartes constitué de 81 cartes toutes différentes qui se distinguent selon 4 caractéristiques : les couleurs, les formes, les remplissage et les nombres d'objet. Le but du jeu est de trouver 3 cartes qui ont des caractéristiques soit toutes les trois identiques, soit toutes les trois différents. Le jeu peut se jouer à un ou plusieurs joueurs, à chaque tour, on dispose 12 cartes sur la table, et le premier joueur qui trouve un bon set de 3 cartes gagne et ces 3 cartes vont remplacer par 3 autre cartes. Si il n'y a pas de bon set parmi ces 12 cartes, on ajoute 3 cartes supplémentaires, comme la probabilité qu'un ensemble de 15 cartes différentes ne contienne aucun set est assez petit, on admet ici qu'il y a toujours un set parmi 15 cartes.

## 2 DESCRIPTION DU PROJET

### 2.1 ANDROID/JAVA

La partie Android consiste principalement 2 parties : les classes 'Card' et 'CardView' et le 'mainActivity'.

- Class 'Card' et 'CardView' Dans l'écran, on a un tableau de 16 cases de class 'CardView' qui gérer les dispositions des cartes. Dans chaque case, on peut déposer soit une carte définie par la class 'Card', soit une carte spéciale qui illustre le temps passé, le score et le bon set que l'on vient de trouver. Chaque carte est caractérisé par 4 chiffres qui représentent les 4 caractéristiques et également un entier entre 0 et 80 qui encode ces 4 chiffre.

- MainActivity Le 'MainActivity' se charge d'interagir avec l'utilisateur. Il permet d'augmenter ou diminuer le score, changer les couleurs des 'CardView' et remplacer les cartes selon les cartes choisis par l'utilisateur.

## 2.2 SERVEUR

Le jeu par multijoueur se réalise plutôt avec le serveur. Dès que le mode 'multijoueur' est choisit, l'appareil local ne distribue plus les cartes ni changer des score. En revanche, on écoute éternellement le serveur. Et le serveur va envoyer les messages qui indiquent les changements des cartes et les renouvellement des scores à chaque clients.

# 3 MÉCANISME

## 3.1 LE JEU

### 3.1.1 LE TAS DE CARTE

Dès que le jeu commence, on crée un tas de 81 cartes différents et ces 81 cartes sont triés aléatoirement. Au début, on mets les premières douze cartes sur le table, et on prend toujours la première carte dans le tas pour renouveler une cartes sur le table.

### 3.1.2 LE DESIGN DE CARTE

Le tas de cartes est une liste de class 'Carte' qui représente chaque caractéristique des cartes par 4 nombres différents. Et c'est dans le class 'CardView' qui permet de décoder ces chiffres et dessiner les cartes sur le table. Chaque 'CardView' correspond à un case de table et contient une carte, dès que la carte est mis. La class 'CardView' commence à désigner la carte selon ses caractéristique.

### 3.1.3 LE JUDGEMENT DE SET

A chaque fois quand il y a trois cartes sélectionnées, la fonction 'haveSet' dans 'mainActivity' juge automatiquement si ces trois cartes formant un bon set et changer les couleurs de ces cartes, le score et changer ou bloquer les trois cartes selon le résultat.

## 3.2 COMMUNICATION

En mode multijoueur, ce n'est plus l'appareil qui crée le tas de carte et renouveler le score. Par contre, le design de carte et le judgement de set se font toujours localement.

### 3.2.1 SERVEUR

La taches de serveur se répart principalement en parties :

- Initialiser le jeu. Cette fois-ci, c'est le serveur qui crée le tas des cartes. Ou plus précisément, 81 entiers différents entre 0 et 80.

- Envoyer et recevoir les messages de chaque clients Pour chaque client, on crée ici un 'thread' qui permet de communiquer avec lui.

### 3.2.2 CLIENT

Le côté client est plus simple. Selon les messages envoyés par le serveur, chaque client change les cartes, renouveler le score ou débloquent les cartes. Par contre, quand l'utilisateur a choisit 3 cartes, le judgement de set se fait par le client. Et on bloque les cartes immédiatement si ils ne sont pas les bonnes cartes. Sinon, on envoie les 'code' des ces trois cartes à serveurs. C'est-à-dire quand le serveur a reçu un set, il est admis que c'est un bon set et le serveur ne juge pas.

## 4 RÉALISATION

### 4.0.3 LE TAS DE CARTE

Pour créer le tas de carte, on veut tirer aléatoirement les 81 entiers entre 0 et 80 qui encode les 81 cartes. Pour faire cela, on a d'abord crée 81 entiers au hasard par la méthode 'random'. Et puis, en gardant les indices de ces 81 entiers, on les trie en ordre croissant. Et comme cela, les indices de ces entiers répartiront aléatoirement. Après en décodant ces entiers, on a bien un tas de carte 'aléatoire'.

### 4.0.4 HANDLER