

M4104 – CDAM

TD 6 : Mini-projet

1 Thème

Le projet de semestre en Info1 va consister à développer une application de bureau permettant de gérer des cartes HEARTHSTONE, en particulier enregistrer sa collection de cartes et constituer un "deck" ([https://fr.wikipedia.org/wiki/Deck_\(jeu\)](https://fr.wikipedia.org/wiki/Deck_(jeu))). Pour être solidaires de vos camarades, vous allez développer une application android permettant de rechercher et visualiser des cartes HEARTHSTONE. Les cartes HEARTHSTONE sont soit des cartes *Serviteur*, soit des cartes *Sort* soit des cartes *Arme* (<https://fr.wikipedia.org/wiki/Hearthstone#Cartes>). Elles ont des caractéristiques communes et des caractéristiques spécifiques. Avant de jouer une partie d'HEARTHSTONE, un joueur va constituer un deck, c'est-à-dire qu'il va sélectionner parmi toutes les cartes celles qu'il va utiliser pour la partie. Il y a plusieurs sortes de decks avec des contraintes sur le nombre de cartes, les cartes possibles, etc.

Votre application utilisera l'api <http://hearthstoneapi.com/> (documentée ici : <https://market.mashape.com/omgvamp/hearthstone>).

2 Fonctionnalités attendues

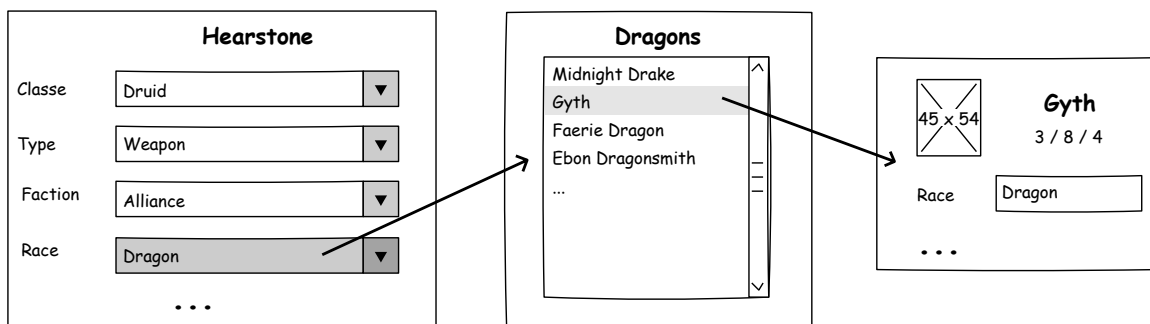


FIGURE 1 – Proposition de maquette

Une proposition de maquette (minimale) est fournie Figure 1. Vous pouvez en concevoir une autre.

- des spinners (indépendants) permettent de choisir le critère de recherche des cartes,
- une présentation des résultats de la recherche est faite dans une liste,
- la sélection d'un item de la liste permet d'en présenter le détail (dont une image) dans une nouvelle activité.
- un soin tout particulier sera apporté à la présentation et à l'ergonomie de l'application.

- les fonctionnalités possibles de votre application (liste NON exhaustive) : pagination des résultats, filtrage, tri des résultats, utilisation de gson/moshi, ...
- un plus : mémorisation de son deck en choisissant des cartes dans les recherches effectuées.

Les bibliothèques autorisées (mais non obligatoires) sont :

- réseau : Volley (<http://developer.android.com/training/volley/index.html>) ou Ion (<https://github.com/koush/ion>).
- pour la gestion simplifiée de json : gson <https://github.com/google/gson> ou moshi <https://github.com/square/moshi/>
- pour les images : <https://square.github.io/picasso>

3 Modalités de rendu

Déposer, par binôme, sur madoc, au plus tard le 03 avril (23h59), une archive compressée (zip) contenant :

- une archive du répertoire de votre projet android studio (après "nettoyage" : `$PROJECT/gradlew -Dhttp.proxyHost=proxyetu.iut-nantes.univ-nantes.prive -Dhttp.proxyPort=3128 -Dhttps.proxyHost=proxyetu.iut-nantes.univ-nantes.prive -Dhttps.proxyPort=3128 clean`)
- une recette de votre projet sera organisée à l'iut le vendredi matin de la semaine 14