

# Rapport Intermédiaire d'avancement de Projet Androïd

Avancement du 26/03/2020

*Projet réalisé par :*

MONTANI Maïlis

*Lien du Repos Github :*

<https://github.com/mayisu7798/ChuckNorrisJokesApp>

*Projet encadré par :*

DUPONCHEL Nicolas

## Partie I - 2 : RecyclerView instantiation

Ne sachant toujours pas ce qu'était une RecyclerView, je me suis renseignée. Il s'agit donc d'un outil permettant d'afficher des listes de manière plus optimisée que l'outil ListView que nous avons utilisé précédemment.

J'ai donc ajouté une ligne dans mon fichier build.gradle (:app). Cette ligne permet entre autre l'import des fonctions du RecyclerView.

```
implementation "androidx.recyclerview:recyclerview:1.1.0"
```

J'ai également ajouté quelques lignes dans mon fichier activity\_main.xml en créant un Layout dans lequel j'ai ajouté mon RecyclerView et en lui ajoutant des contraintes et un identifiant.

Enfin, j'ai ajouté quelques lignes dans mon fichier MainActivity.kt pour ajouter le RecyclerView que j'ai créé dans le fichier activity\_main.xml en utilisant la fonction :

```
findViewById(R.id.RecyclerView)
```

Pour fonctionner, le RecyclerView a besoin d'un Layout dans lequel s'afficher que l'on lui ajoute en utilisant la ligne de code :

```
recycler.layoutManager = LinearLayoutManager(this)
```

mais également d'un Adapter contenant les données à afficher, que nous allons tenter de créer à la question suivante.

Note : Cette partie a été très rapide à réaliser puisque je l'avais déjà commencée la dernière fois.

Temps : ~ 20 minutes

## Partie I - 3 : Custom Adapter

Pour cette partie, il faut créer un nouveau fichier classe Kotlin pour créer notre classe JokeAdapter. Pour lui fixer l'attribut correspondant à la liste à afficher (càd notre StaticList), on utilise directement les arguments de la classe (constructeur implicite).

On ajoute ensuite une sous classe JokeViewHolder qui extends RecyclerView.ViewHolder avec la ligne suivante :

```
class JokeViewHolder (var textView : TextView) :  
    RecyclerView.ViewHolder(textView)
```

Ainsi notre classe JokeAdapter extends RecyclerView.Adapter<JokeAdapter.JokeViewHolder>(). On peut maintenant override trois fonctions :

- `onCreateViewHolder` : Fonction qui crée le `JokeViewHolder` associé au `JokeAdapter` et le retourne
- `getItemCount` : Fonction qui retourne le nombre d'items de la liste passée en paramètre de `JokeAdapter`
- `onBindViewHolder` : Méthode qui prend l'élément de la liste à la position indiquée en paramètre et remplace l'élément affiché par celui-ci

Nous devons ensuite créer une méthode `setter` pour indiquer au `JokeAdapter` le changement éventuel de la liste passée en paramètre.

Dans le `MainActivity`, nous avons ensuite créé notre `JokeAdapter`, en lui passant en paramètre notre `StaticList` de `Jokes`. Nous avons ensuite pu l'assigner au paramètre `adapter` de la `RecyclerView`.

Note : J'ai passé énormément de temps sur cette partie, car mon application refusait de se lancer. Je suis donc passée à la suite en attendant de trouver une solution pour voir s'il s'agissait simplement d'une erreur qui allait se résoudre avec la suite.

Temps : ~ 2h00

## Partie I - 4 : Custom View for Items

Nous devons créer un nouveau fichier `joke_layout.xml` dans lequel afficher le contenu de notre `RecyclerView`. Ce fichier ne contient que un `<TextView/>` avec quelques options supplémentaires pour l'affichage du texte.

Dans le fichier `JokeAdapter`, nous devons ensuite changer le contenu de la fonction `onCreateViewHolder` afin de lui assigner le `TextView` que nous venons de créer dans le fichier précédent grâce à la ligne :

```
val textView = LayoutInflater.from(parent.context)
    .inflate(R.layout.joke_layout, parent, false) as TextView
```

Nous changeons également le contenu de `activity_main.xml` et ne laissons dans ce fichier que le `RecyclerView`, encapsulé dans son `Layout`.

Note : Je travaillais toujours à la suite de cela à trouver mon erreur sur la partie précédente puisque mon application refusait toujours de se lancer. Il s'est avéré qu'il s'agissait d'une erreur dans l'écriture de `<androidx.recyclerview.widget.RecyclerView/>` dans mon `activity_main.xml`. Maintenant l'application fonctionne. Je me suis aussi rendue compte que j'avais oublié depuis un certain temps de commit sur mon git. Je l'ai donc fait.

Temps : ~ 2h00

- Commit : [Création d'un Joke Adapter](#)
- Commit : [Changement pour un affichage du texte plus personnalisé](#)

## Partie II - 1 : Create the data model matching the API

J'ai lu le début mais je vais m'arrêter là pour aujourd'hui, j'ai beaucoup de cours à rattraper encore.