

CR TP Séquence 1

1. Introduction

J'avais déjà réalisé une application Android en équipe pour le projet G1G2 à Centrale. Ainsi l'utilisation de l'IDE Android Studio, la logique de programmation Android, la mise en place de layouts évolués utilisant des RecyclerView et des Adapters et la persistance de données en préférences ne m'ont pas posé de réels problèmes insurmontables après avoir effectué des recherches dans la littérature (voir partie 5 sur la bibliographie).

L'utilisation du langage Kotlin au lieu de Java était une première pour moi et m'a demandé de reprendre les bases de la programmation Android en Kotlin. Cependant, je me suis rendu compte de la praticité de Kotlin par rapport à Java pour la programmation Android.

Concernant la sérialisation/désérialisation en JSON avec la librairie GSON, c'est la première fois pour moi que je l'utilise. Je pense avoir réussi à l'implémenter, toutefois je ne suis pas certain d'avoir respecté les us et coutumes par rapport à leur utilisation.

2. Analyse

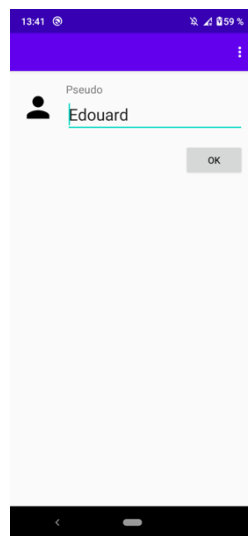
D'une manière générale l'application fonctionne correctement et répond aux fonctionnalités exigées dans l'énoncé du TP voir aussi quelques fonctionnalités facultatives (auto-complétion lors de la saisie du pseudo, vidage de l'historique des pseudos saisis auparavant, modification ou suppression d'une liste ou d'un item).

En résumé, j'ai réalisé la programmation de 4 activités : *MainActivity*, *ChoixListActivity* et *ShowListActivity* qui hérite de la classe *AppCompactActivity*, qui ont chacune leur layout respectif, et qui utilise la même toolbar avec le même menu ; et *SettingsActivity* qui hérite de *PreferenceActivity* qui a son propre fichier xml.

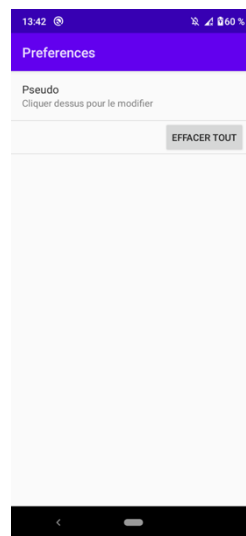
J'ai également créé 3 fichiers kotlin supplémentaires : les classes *ListeToDoAdapter* et *ItemToDoAdapter* qui hérite de *RecyclerView.Adapter* et qui implémentent les méthodes nécessaires à la gestion des adapters des recyclerviews des activités *ChoixListActivity* et *ShowListActivity* pour l'affichage des listes et des items dans les listes, et *ProfilListeToDo* qui contient les 3 classes nécessaires à la désérialisation du fichier Json *ProfilListeToDo*, *ListeToDo* et *ItemToDo*.

Par ailleurs j'ai utilisé 3 autres layouts : *liste_to_do* et *item_to_do* qui sont utilisés au sein des ViewHolders des recyclerviews des activités *ChoixListActivity* et *ShowListActivity*, *dialog_new* qui est utilisé lors de l'affichage d'AlertDialog pour rentrer un nouveau nom d'item ou modifier un nom d'item ou de liste.

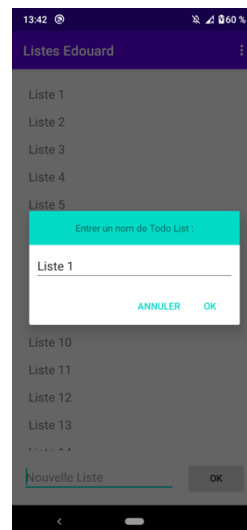
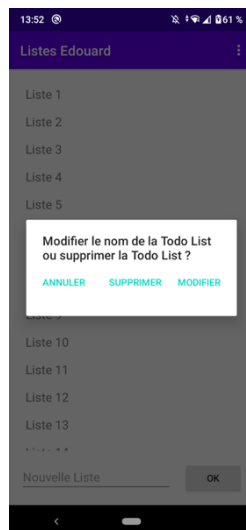
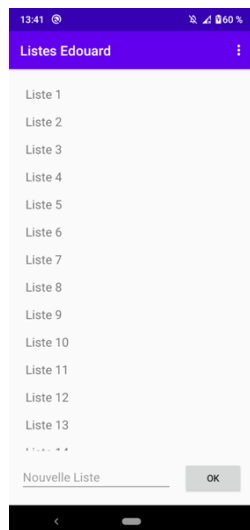
Ci-dessous des captures d'écran de l'application en fonctionnement :



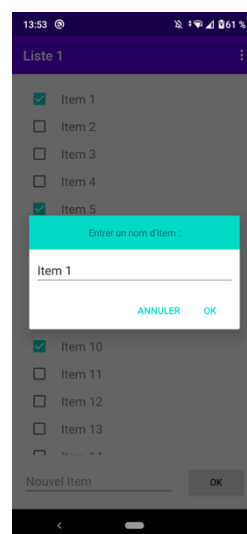
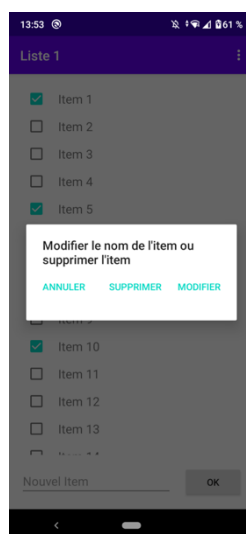
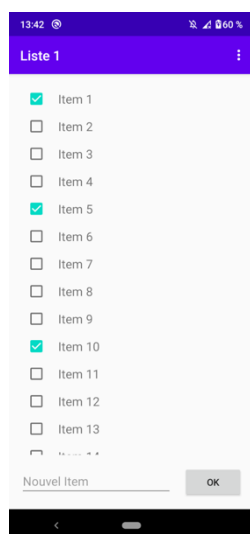
MainActivity



SettingsActivity



ChoixListActivity



ShowListActivity

3. Conclusion

Je suis satisfait du rendu de l'application et de son fonctionnement. Et la programmation de cette application m'a permis de me rendre compte de la complexité apporté par l'ajout d'une quelconque fonctionnalité sur la logique globale de programmation de l'application.

J'ai aussi été sensibilisé aux bonnes pratiques de programmation car si on ne les utilise pas il faudrait écrire plus de lignes de code que nécessaire et il serait difficile de se repérer dans le code pour le compléter.

Ainsi j'ai compris l'avantage qu'apporteraient certaines bonnes pratiques comme l'utilisation des ressources values, string et styles par exemple.

4. Perspective

Concernant l'activité Settings, je pourrais ajouter une préférence sur la taille de la police et une préférence sur la langue de l'application.

Concernant la disposition des listes et la disposition des items de ces listes, je pourrais ajouter une fonctionnalité permettant de les déplacer afin de pouvoir choisir l'ordre de priorité des listes ainsi que des items au sein d'une liste.

Concernant les items cochés dans une liste, je pourrais les faire apparaître au bas de la liste pour expliciter la réalisation de cette tâche.

5. Bibliographie

Developer Guides <https://developer.android.com/guide/>

Material Design <https://material.io/design>

Les cours et screencasts de la séquence 1 de l'électif PMR