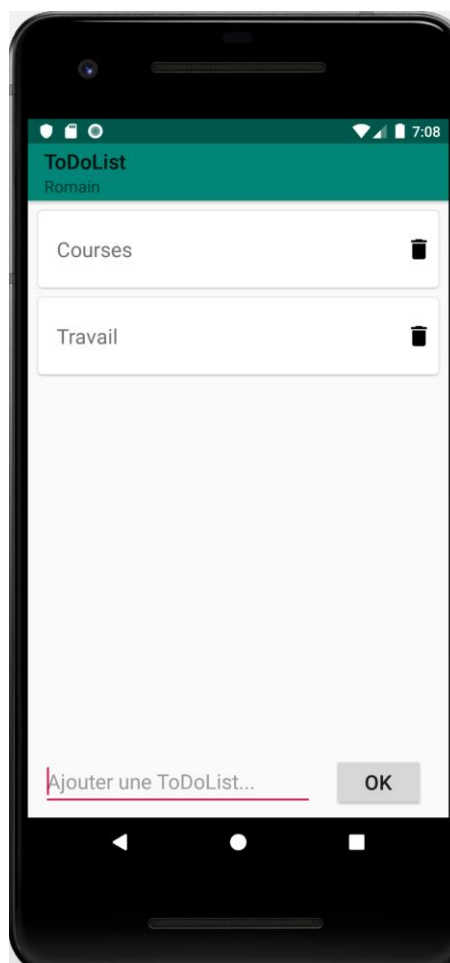




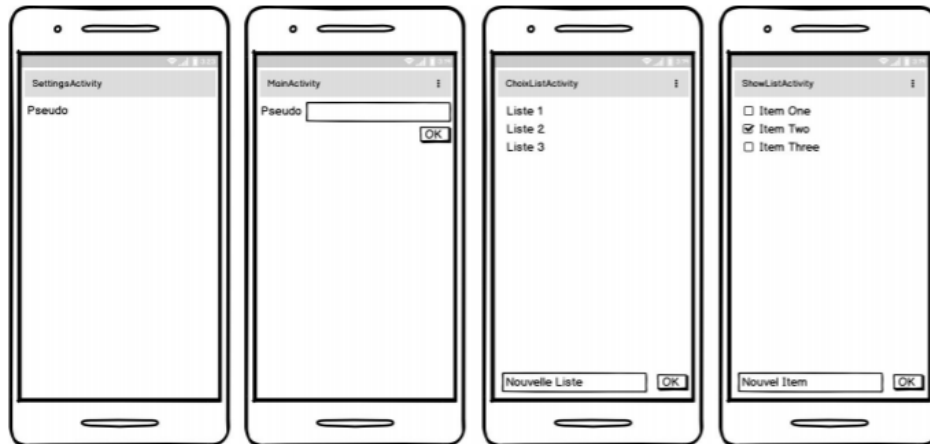
Première application Android

ToDoList



Objectifs

L'objectif de ce TEA était de créer une application mobile de type "To Do List". Le cahier des charges était le suivant :



Pour résumer, pour une même application, nous devons avoir plusieurs utilisateurs, pouvant insérer et supprimer des To Do Lists, dans lesquelles il serait possible d'ajouter et supprimer des Tâches. Ces tâches devaient pouvoir être mise en "Faites" ou "Non faites" à l'aide d'une checkbox.

Pour chaque pseudo rentré, l'ensemble des utilisateurs devait être enregistré dans les paramètres (item du menu).

Bien entendu, les To Do Lists et les Tâches devaient être enregistrées en mémoire, afin de ne pas perdre toutes les données une fois l'application fermée.

Equipe et organisation

L'équipe est composée de deux élèves :

- MABBOUX Romain
- GERVASI Pierrick

Dès le début, nous avons décidé de nous organiser afin de trouver une bonne répartition du travail pour être efficaces. Nous avons décidé de nous organiser de la manière suivante:

- Pierrick devait être responsable de toute la partie FrontEnd, avec notamment la gestion des layouts, des recyclerview, et des classes.
- Romain devait être responsable de toute la partie BackEnd, concernant notamment le passage d'objet d'une activity à une autre, et toute la partie Json et Gson.



Etant donné que nous devions travailler à distance la plupart du temps, nous avons décidé d'utiliser GitHub pour gérer le code de l'application.

Cependant, nous n'avions pas vraiment de connaissances pour gérer GitHub sur Android Studio (onglet VCS). Pour en apprendre davantage sur GitHub sur Android Studio (notamment pour anticiper le projet final), nous avons décidé de suivre une série de vidéo explicative sur youtube, disponible à ce lien:

<https://www.youtube.com/playlist?list=PLhO3X757HABdao5hv4HGQSWPX0tnN4ICO>

Nous y avons trouvé 1h50 de vidéos nous expliquant comment fonctionne GitHub sur Android Studio et des éléments supplémentaires sur les techniques de base (Pull, Push, Merge, etc...)

En revanche, n'étant que 2 pour cette application et pour des raisons de rapidité, nous avons travaillé uniquement sur la branche principale (master), et avons effectué des push et pull dessus.

Concernant les conflits et problèmes, nous les réglions ensemble. Voici le lien vers notre [Github](#) (une invitation en tant que collaborateur a été effectuée).

Présentation de l'application :

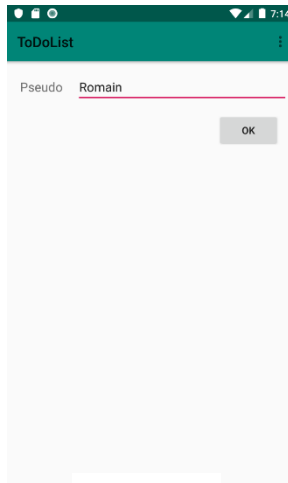


Figure 1

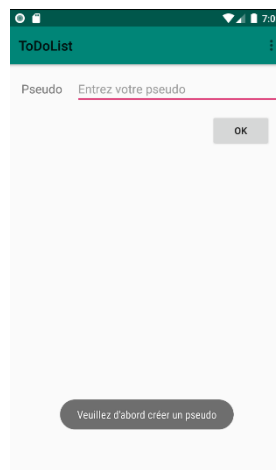


Figure 2

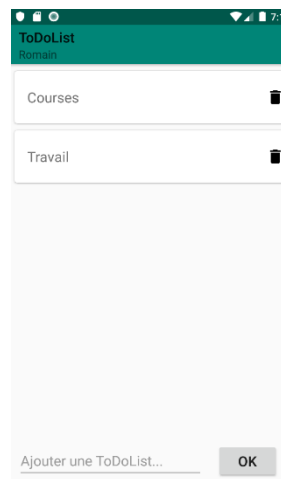


Figure 3

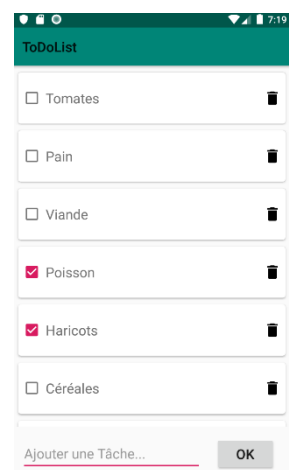


Figure 4

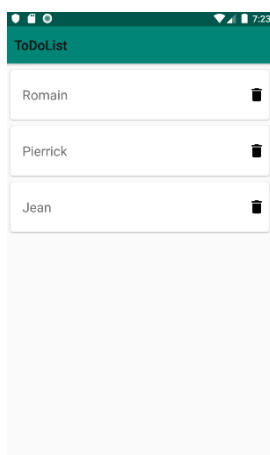


Figure 5

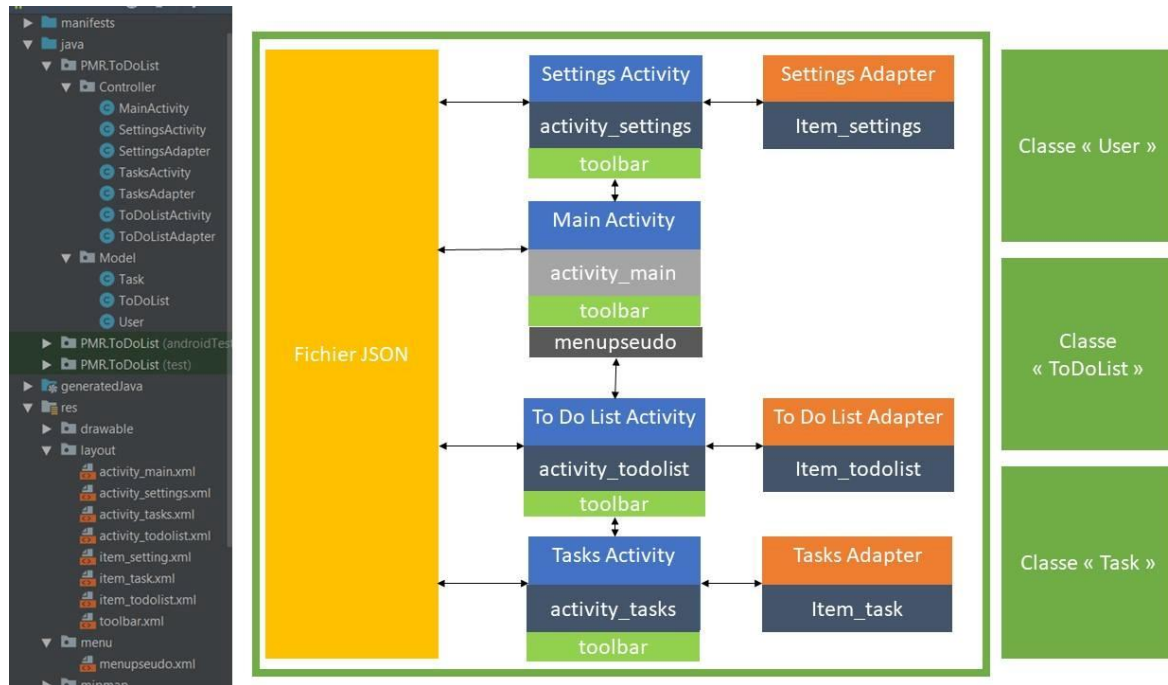
La première activité est une page permettant à un utilisateur d'entrer son pseudo (figure 1). Si ce pseudonyme a déjà été rentré, l'application réutilise les anciennes données (ToDo listes et tâches) déjà rentrées. C'est depuis cette activité que l'on peut accéder à la liste des utilisateurs, via l'option de menu « Settings » (figure 5). Depuis cette nouvelle activité, on peut contrôler la liste des utilisateurs, et en supprimer. Si la liste des utilisateur est vide, un toast prévient l'utilisateur qu'il doit entrer un premier pseudo afin de pouvoir accès à cette activité (figure 2).

La deuxième activité (figure 3) est une page correspondant aux ToDo listes liées à l'utilisateur rentré dans l'activité principale. Elle permet d'ajouter (via l'editText) ou bien de supprimer (via un clique sur l'image de poubelle) une ToDo liste. Un clic sur une ToDo liste permet d'accéder aux tâches associées.

La troisième et dernière activité (figure 4) correspond donc à la liste des tâches associées à la ToDo liste qui a été cliquée. On peut ajouter et supprimer des tâches simplement, de la même

manière que l'on peut ajouter des ToDo listes. Un case à cocher permet de visualiser une tâche « effectuée ».

Fonctionnement de l'application



Au niveau du fonctionnement de l'application, voici ci-dessous sa structure :

Nous avons préalablement créé les fonctions "User", "To Do List", et "Task", qui représentent les objets que l'on considère et manipule dans toute l'application.

Nous avons les 4 Activités : Settings Activity, Main Activity, To Do List Activity, et Task Activity.

Chacune de ces Activités possède un layout « activity_nomdelactivité » qui inclut la toolbar. Etant donné qu'il est possible de supprimer des utilisateurs enregistrés depuis le menu « Settings », nous n'avons mis le layout du menu « menupseudo » que dans la main Activity. Dans le cas contraire, un utilisateur pouvait accéder aux todolists d'un compte, supprimer ce compte depuis ces todolists, et continuer à modifier ces todolists alors que le compte associé était supprimé.

D'autre part, chacune des activités « Settings », « ToDoList », et « Task » doit afficher une liste. On utilise alors des RecyclerView gérés par des Adapter pour chacune de ces activités. Tous les Items présents dans les RecyclerView de ces activités sont des CardView, que l'on a pu implémenter dans gradle. Ce choix a été fait par pur choix d'esthétisme.

Durant toute l'utilisation de l'application, c'est-à-dire lorsqu'on crée un utilisateur non rentré, une nouvelle to do list, ou une nouvelle tâche, le fichier JSON ainsi que les objets considérés sont mis à jour au fur et à mesure.

D'autre part, une fois que l'on relance l'application, le fichier JSON est de nouveau lu pour recréer tous les objets créés auparavant afin de conserver les données.

Conclusion

En conclusion, nous pouvons dire que l'élaboration de cette application a été très enrichissante. Elle nous a premièrement permis de maîtriser les outils de bases que sont Android Studio et Github. Elle nous a ensuite donnée un bon aperçu de ce que représentait le développement d'une application interactive, ou les stockages des données et un point essentiel. Bien que notre application ne soit pas optimisée, elle est fonctionnelle et applique les principes vus en cours et en TD.

Perspectives

FrontEnd

- Suggestions pour les pseudos déjà existants (editText MainActivity)
- ToDo listes et tâches supprimables par swip

BackEnd

- Code davantage commenté
- Code plus propre (supprimer les redondances...)

Bibliographie

- Site sur les [HashMap](#)
- MOOC [Github](#)
- Site pour [putExtra](#) des objects
- Chaîne Youtube pour [recyclerview](#)