

Projet réalisé en 5h.

Architecture MVVM Android standard

- 3 couches: UI, Presentation, Business
- 1 couche Common pour tout ce qui est mutualisable ou générique.

Android Jetpack:

- Databinding
- Navigation Component
- Room
- LiveData

Bibliothèques externes:

- Retrofit pour les appels réseaux
- CircleImageView: librairie open source pour afficher une image dans un cercle.
- Glide pour charger les urls des images

Si j'avais pu y passer plus de temps:

- Gestion d'erreur plus précise
- Créer un service pour la navigation plutôt que de la réaliser dans les fragments, car la logique de navigation n'a rien à faire dans l'UI.
- Faire une vraie architecture MVVM, pas celle de Google..., avec de la composition de View Model. Notamment dans la liste chaque cellule devrait avoir son propre View Model plutôt que d'utiliser celui du fragment principal.
- Ajouter un système de paging avec un « load more » quand on arrive en bas de la liste de manière à charger plus d'éléments.