

# Mini Projet Android

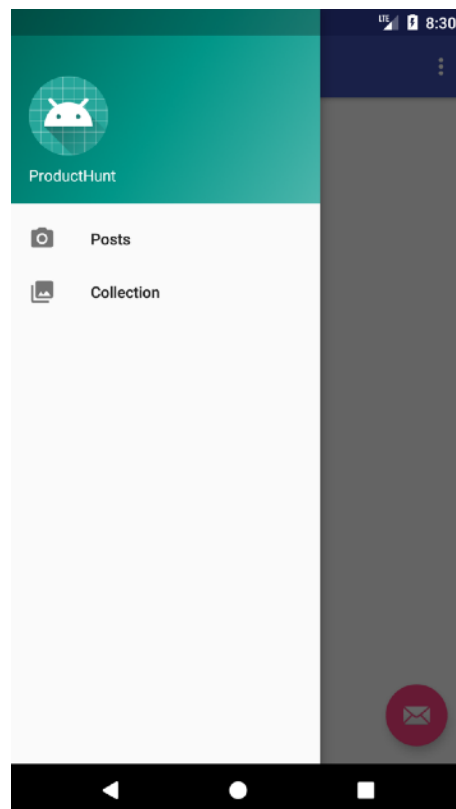
## Objectif :

Développer une application Android qui consomme l'api Product hunt pour afficher la liste des publications avec leurs commentaires ainsi que la gestion du mode hors ligne.

1. Module Écran Home.
  - a. Liste des publications
  - b. Liste des collections
2. Module Écran Détail de publication.
3. Module Écran Liste des commentaires.

## Module Écran Home:

Mettre en place un DrawerLayout qui contient un bouton pour afficher la liste des publications et un bouton pour afficher la liste des collections.



- **Liste des publications (Fragment)**

1. Charger et afficher la liste de publications triées par date depuis la base de données.
2. La première publication de la liste doit être mise en avant :
  - a. Taille de l'élément doit être plus importante que les autres.
  - b. L'image doit prendre toute la largeur de l'élément.
  - c. Le Titre et le sous-titre doit être affiché en dessous de l'image.
3. Pour chaque publication, il faut afficher un compteur avec le nombre des commentaires.
4. Au clic sur une publication il faut lancer l'Écran Détail de publication.
5. Au clic sur le compteur des commentaires il faut lancer l'Écran liste des commentaires.
6. Il faut pouvoir actualiser la liste des publications en faisant un Swipe To Refresh ([voir la doc](#)) :
  - a. Actualiser : Télécharger et sauvegarder les nouveaux Posts en suite les affichés dans la liste.

**NB:**

- Pour récupérer les nouvelles publications il faut faire l'appel web-service suivant :
  - [https://api.producthunt.com/v1/posts/all?newer=id\\_post](https://api.producthunt.com/v1/posts/all?newer=id_post)  
id\_post : est l'id de la dernière publication sauvegardée dans l'application.

- **Liste des collections**

1. Charger et afficher la liste des collections depuis le service web productHunt
2. Au Clic sur une collection de la liste il faut afficher la liste des postes qui font partie de cette collection dans l'écran DetailCollection ce dernier doit réutiliser doit réutiliser le fragments liste des postes
  - [https://api.producthunt.com/v1/docs/collections/collection\\_show\\_get\\_details\\_of\\_a\\_collection](https://api.producthunt.com/v1/docs/collections/collection_show_get_details_of_a_collection)

### **Module Écran Détail de publication:**

1. Afficher l'url du produit (redirect\_url) de la publication dans une web-view (voir le TP).
2. Ajouter un bouton au niveau menu pour permettre à l'utilisateur d'afficher l'url du produit dans le navigateur web.

### **Module Écran liste des commentaires:**

1. Afficher le titre de la publication au niveau de la barre d'action de l'écran.
2. Charger et afficher la liste des commentaires triés par date depuis la base de données par date.
3. Il faut afficher les réponses sur un commentaire directement en dessous de ce dernière voir la maquette Figure 1 (dans le Json child\_comments).
4. Il faut pouvoir actualiser la liste des commentaires en faisant un Swipe To Refresh ([voir la doc](#)) :
  - a. Actualiser : Télécharger et sauvegarder les nouveaux commentaires en suite les affichés dans la liste.
  - b. Mettre à jour le compteur des commentaires au niveau de la publication
5. Au retour sur le Module Écran liste des publications, il faut mettre à jour le compteur des commentaires de la publication qu'on vient d'afficher et synchroniser.

### **NB:**

- Pour récupérer les nouveaux commentaires d'une publication, il faut faire l'appel web-service suivant :

[https://api.producthunt.com/v1/posts/id\\_post/comments/?newer=id\\_last\\_comment&access\\_token=](https://api.producthunt.com/v1/posts/id_post/comments/?newer=id_last_comment&access_token=)

id\_post : est l'id de la dernière publication sauvegardée dans l'application.

id\_last\_comment : est l'id du dernier commentaire sauvegardé pour la publication de l'id : id\_post.

## Maquettes:

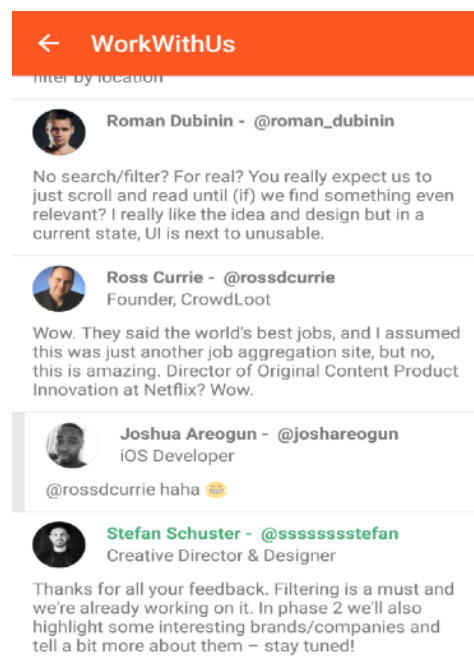


Figure 1 exemple d'affichage de la liste des commentaires