

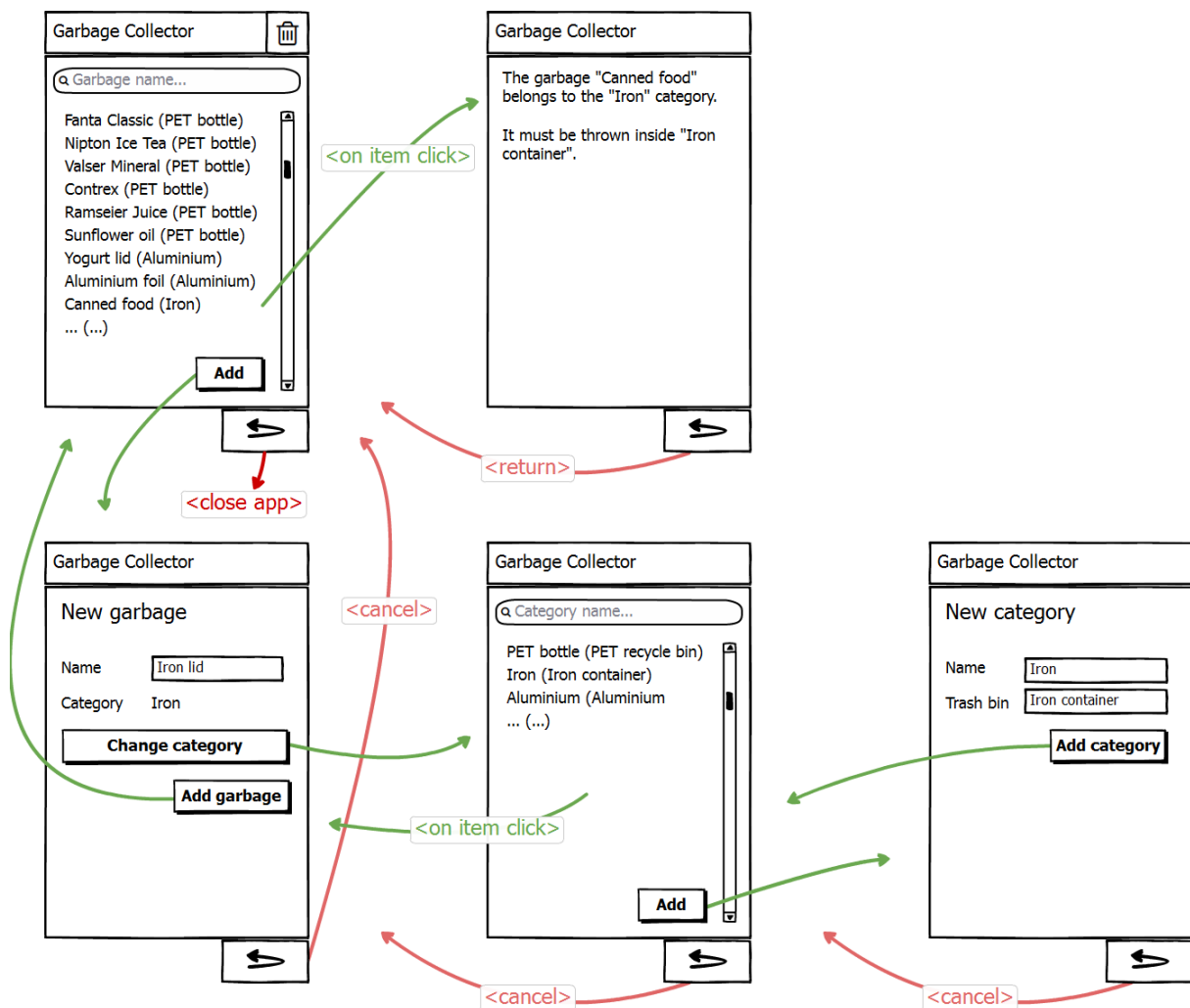
Android : Application « Garbage Collector »

Description de l'application

« Garbage Collector » est une application à développer de manière incrémentale par les étudiants DLM lors du cours Android.

Le but de l'application est de permettre à un utilisateur de déterminer dans quelle poubelle il doit jeter ses déchets. Pour cela, il pourra effectuer une recherche par nom ou par catégorie de déchets. Il aura également la possibilité de rajouter / supprimer des déchets ou catégorie de déchets.

Le flow général de l'application est le suivant (où l'activité de départ est en haut à gauche) :



Comme l'image en page précédente le montre, l'application sera composée de 5 activités :

- la liste des détritrus ;
- les détails d'un détritrus ;
- la création d'un nouveau détritrus ;
- le choix d'une catégorie pour un détritrus ;
- la création d'une nouvelle catégorie de détritrus.

La communication et la navigation entre ses différentes activités est visible par les flèches de navigation sur l'image.

En plus de ceci, l'utilisateur aura la possibilité de supprimer les détritrus et catégories de détritrus supplémentaires qu'il a ajouté dans l'application en cliquant sur l'icône d'une corbeille en haut à droite, sur l'activité principale.

Étapes de développement

Chaque étape de développement de cette application entraînera la découverte d'une nouvelle fonctionnalité primordiale du SDK Android. Au total, 10 étapes seront prévues ; la liste des étapes (avec la fonctionnalité découverte) ci-dessous :

1. L'environnement de développement Android Studio :
 - prise en main du logiciel,
 - *logging* avec « LogCat »,
 - qu'est-ce qu'une *activity*,
 - cycle de vie d'une *activity* ;
2. les *views*, les *layouts* et la classe de ressources « R » :
 - découverte des fichiers XML de ressources,
 - création de *views* dans une *activity* à l'aide du *designer* d'Android Studio,
 - manipulation d'une liste à l'aide RecyclerView ,
 - filtrage d'une liste à l'aide d'une classe « Filterable » ;
3. les *intents* :
 - navigation et communication de données entre les *activities* ;
4. les fragments :
 - création de vues réutilisables,
 - adaptation du *layout* des vues selon l'orientation du téléphone ;

5. les barres de menu :

- création d'une barre de menu,
- gestion des actions d'un menu ;

6. la programmation concurrente :

- comprendre la problématique d'une *activity* qui ne répond plus,
- résoudre le problème en utilisant des coroutines ;

7. les services :

- création d'applications sans interface graphique,
- gestion de longues tâches de fond récurrentes ;

8. les broadcasts :

- envoyer des messages lisibles depuis une autre application,
- retourner des informations depuis un service à une *activity* ;

9. la persistance de données :

- conserver des données même après l'arrêt d'une application,
- utilisation room ;

10. les capteurs :

- récupérer des informations physiques,
- changer le comportement d'une application selon une contrainte physique.