Expense Tracker

GitHub Username: Hemal Shah (www.github.com/hemal-shah/)

Expense Tracker

Description

Expense Tracker app provides means for users to record their expenses, individually or collectively with family members /roommates /colleagues and review their expenses anytime.

Intended User

Financially restricted family members, students attending university abroad, or students living together in an apartment can use this application to manage their expenses.

Features

Expense Tracker helps user in the following ways:

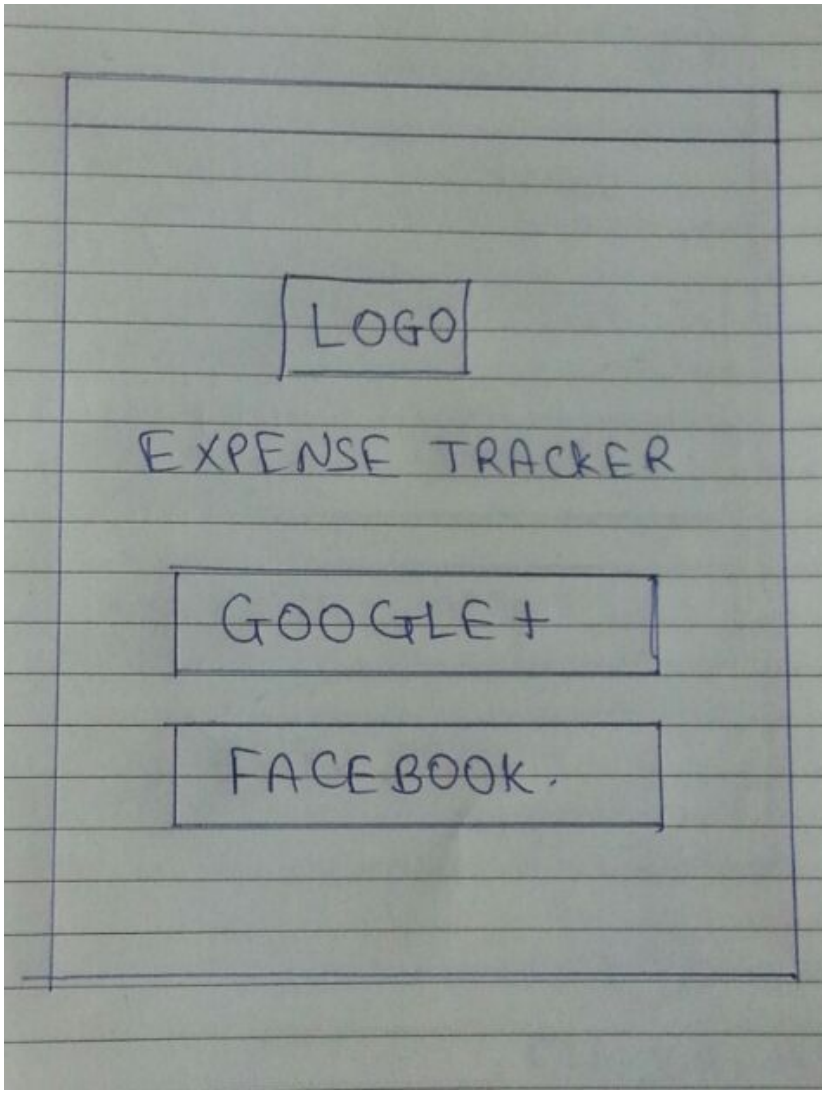
- Maintain a unified list of all expenses user has made in the past.
- Group the expenses into a cluster. For eg.
    - Food Cluster will contain all the expense related to food
    - Home Cluster will contain expense done for decoration, maintenance, or investments done in Home.

Expense Tracker

    - Self Grooming Cluster can contain expenses made in shaving, buying cosmetics, etc.
    - **NOTE: Clients can create their own personalized clusters.**
- Multiple people can contribute to same expense cluster. I.e. Family members can create their own clusters and add each member of family into it. That way they can monetize their expense.
- Personalized Pie Graphs for a solid representation of expenses made by each participant in the cluster.
- Monthly reminders to add repeated expenses such as electricity bills, water bills, phone bills.

User Interface Mocks

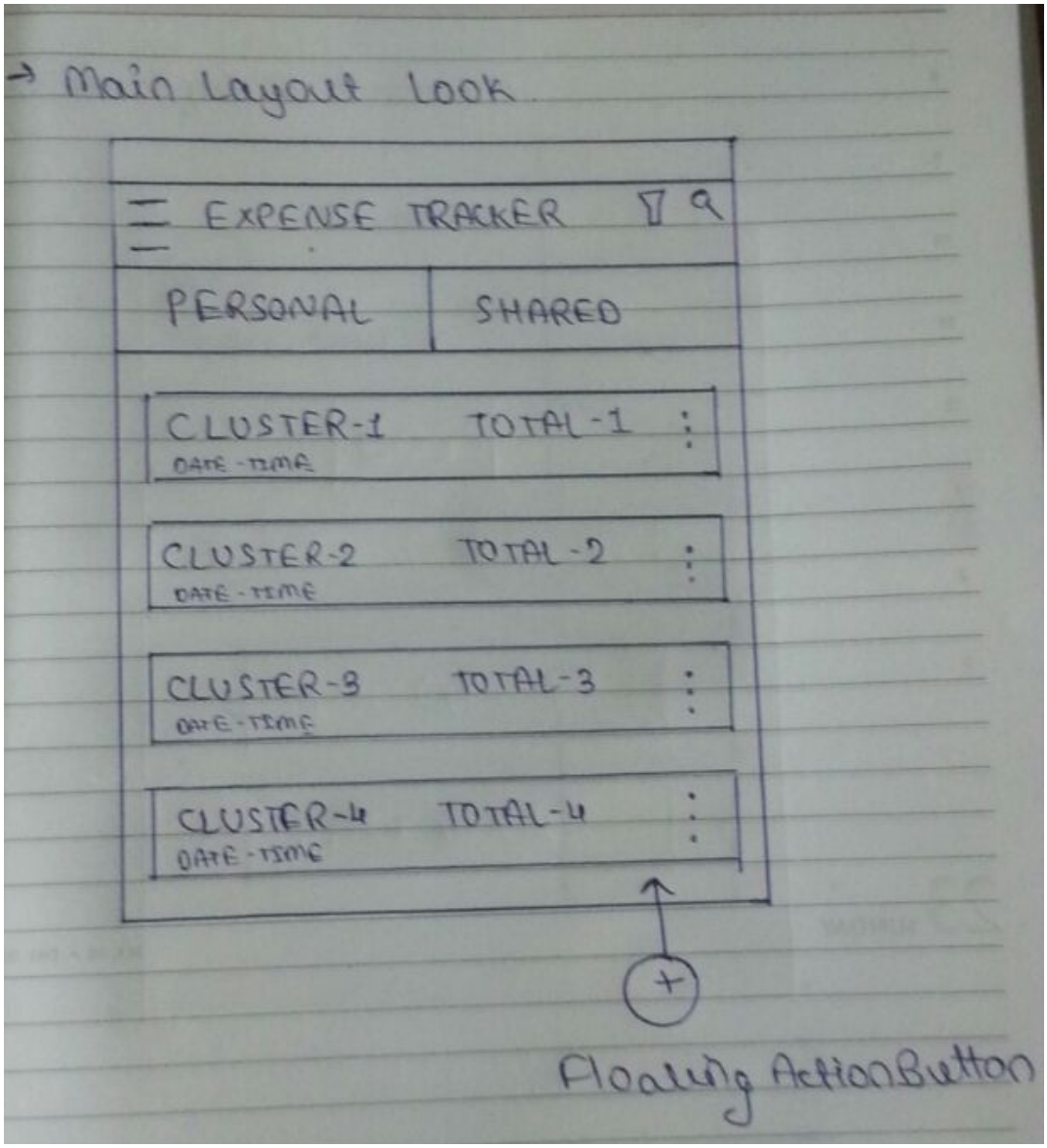
Screen 1 : Login Screen (Appears Only when not logged in)



Expense Tracker

The above screen would appear on the first time user opens the app, or alternatively logs out of the app. It would use Google Identity service to retreive name and email. Facebook would also be used for the same purpose.

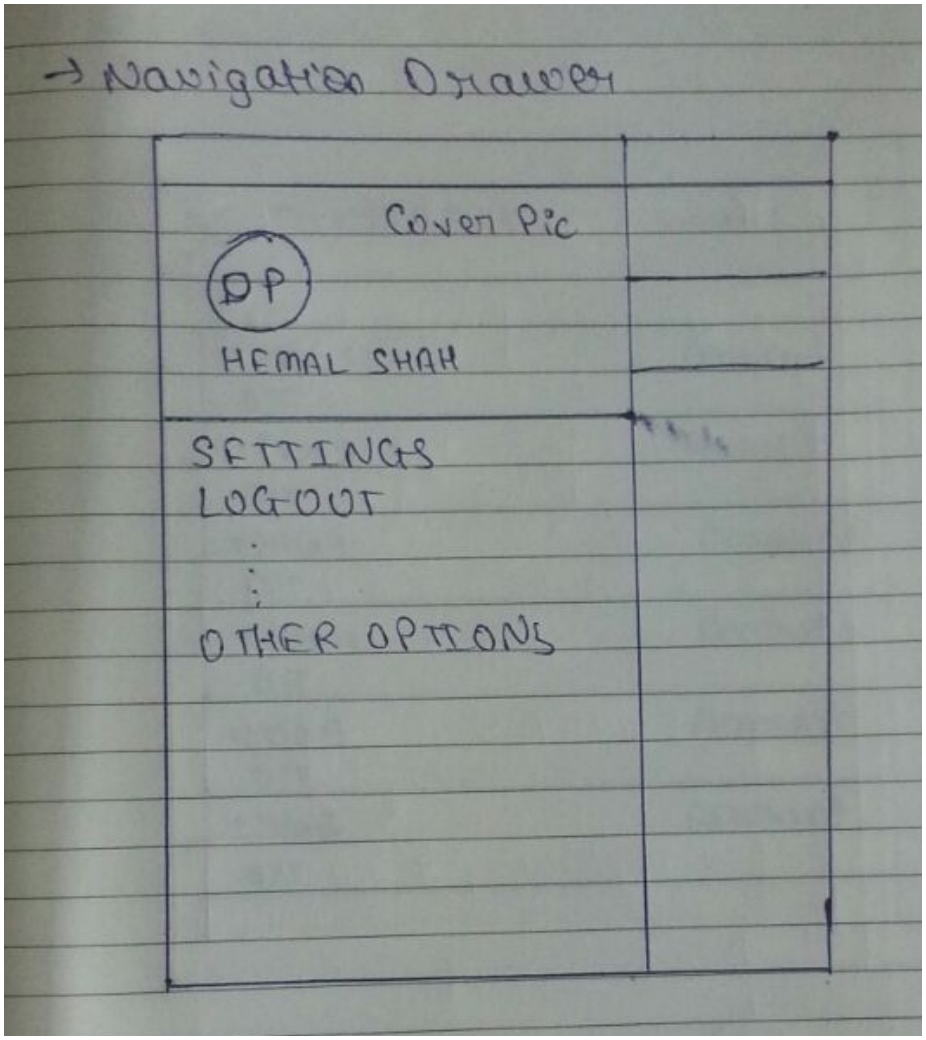
Screen 2 : Main Layout



After the user logs into the screen, two tabs would be provided to look for personal expense clusters and shared clusters. In each, name of the cluster, the total expense in that cluster, and option menu would be provided on the extreme right.

Expense Tracker

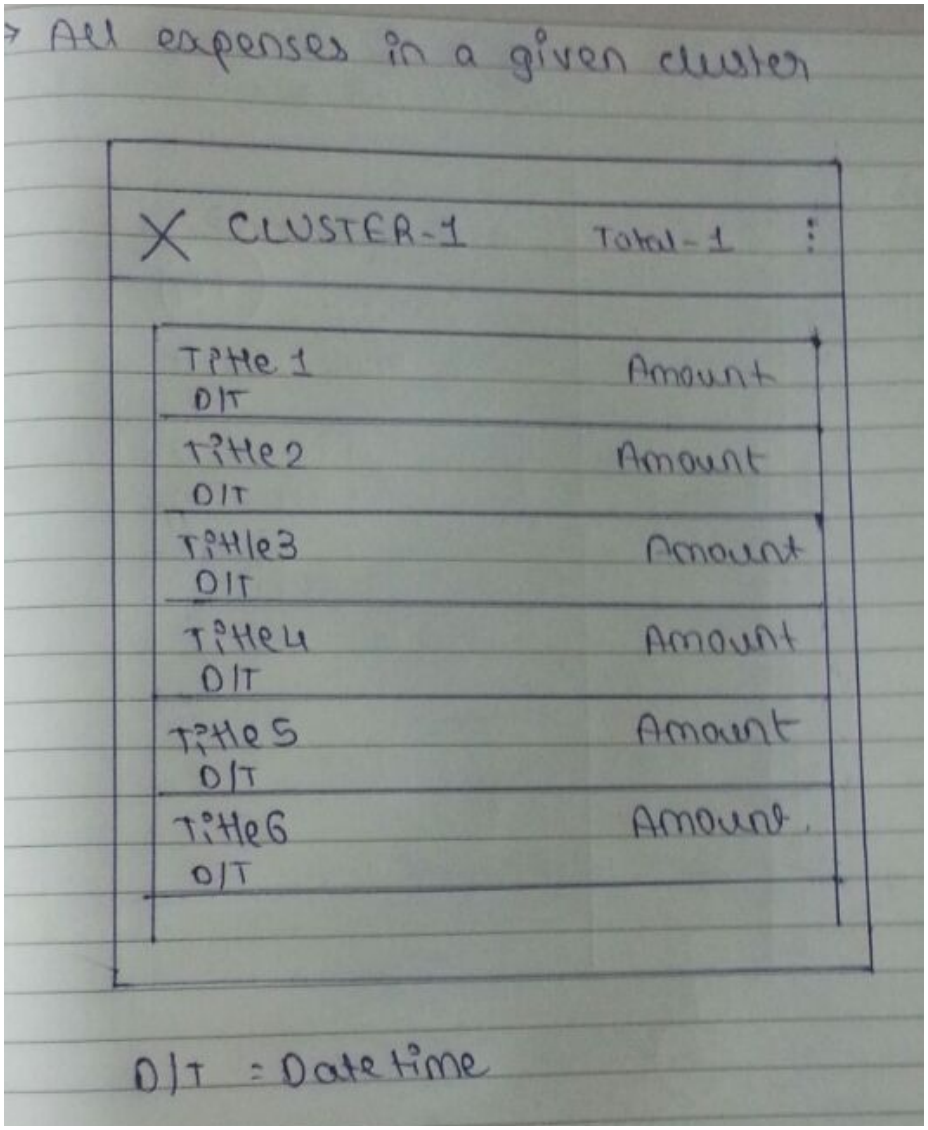
Screen 3: Navigation Drawer



Using the Google+ login / Facebook Login, I would obtain name, profile picture and cover photo, and display them in the Navigation Drawer.

Expense Tracker

Screen 4: Inside Each Cluster



Whenever clicked on a cluster, all the expenses made in that cluster, would appear here with appropriate title, time and date, and the amount. If it is a shared cluster, the name of the user who made the expense would be shown. FloatingActionButton would be provided to add new expense into a cluster.

Expense Tracker

Key Considerations

How will your app handle data persistence?

I will create my own custom Content Provider to connect to SQLite. Also I will maintain online database on Firebase, so that whenever user logs into another device, every past data can be available.

Describe any corner cases in the UX.

I am considering two corner cases in my UX:

1. I want to implement Inbox App style “swipe down to close” feature, which will close the cluster.
2. There are two options available for user to delete/archive any expense:
    a. Swipe to dismiss(Inbox/Gmail style app)
    b. Selectable cards and remove them all together.

Maybe I will provide options for bothe 2-a and 2-b in the application, depending upon how much time it would take to implement.

Describe any libraries you’ll be using and share your reasoning for including them.

I would be using following libraries, and it may change while I’m implementing the code:

1. Butterknife Library: To bind views and avoid writing boilerplate.
2. RecyclerView: To use the handy features of swipe to dismiss or runtime layout manager(LinearLayoutManager or GridLayoutManager).
3. Design Support Library: To make the app material by using FloatingActionButton, Coordinator Layout, DrawerLayout, etc.
4. Firebase Libraries/GCM Libraries: To store data online and also provide push notification whenever user makes any changes.
5. Google Plus and Facebook Sign in: To authenticate users and obtain their Names and Profile Photo.
6. Google Admobs library: For advertisement revenue generation.

Describe how you will implement Google Play Services.

1. **Google Identity**: Used to sign-in into the application and help to look at all the expenses made by user on any device, user chooses to sign into.
2. **Google Admobs**: To display advertisements, mostly full screen advertisements for better UI.

Expense Tracker

Required Tasks

Task 1: Project Setup

I will set up the android studio project with minimum support library mentioning, and also initiate git repository to commit changes to github.com. Along with that, I would add all the above mentioned libraries in the gradle file, so I can use them alongside in the project.

Task 2: Implement UI for Each Activity and Fragment

There is need for creating for following activities/fragment (including they look nice on tablets too):

- Main Cluster look which will contain all the main expense clusters(Listview of cards).
- Expense View inside each cluster that contains all expenses of a single cluster
- Settings menu
- Summary screen, that displays all expenses in graphical format.(Pie Chart)

Task 3: Creating Database

During this, I will not only construct the database, but also ContentProvider also to get data from the database and continuously sync it with my views.
NOTE: I haven't really studied how to use firebase, but I am surely going to use that to make the user's data ubiquitous on each device.

Task 4: Creating Widget

Creating a widget for the user, so that user can see contents of any one cluster on the launcher screen per one widget. More widgets would be needed to add to view multiple clusters.

Task 5: Displaying Notifications for wear

Expense Tracker

This is an app that I personally think would invade user's personal space by creating and packaging a apk for android wear. Hence, if through the settings menu, user dictates that notifications should appear on Android Wear Device, then and then only, new notifications would be displayed on user's Android wear.

Task 6: Displaying Advertisements:

Adding advertisements so that revenue can be generated.

Task 7: Adding firebase shared cluster connectivity:

I would be using firebase database, to connect all the users into same clusters, and provide the functionality for multiple users to contribute their expenses simultaneously while learning the Freindly Chat app from Udacity's Firebase in a Weekend course. Hence, whenever user makes a new expense, it would be shown in all the connected shared users.

Stage 2 Requirements :

1. Content Providers: Yes, I would be implementing Content Providers, which will act as an interface between my database, and UI of the app.
2. AsyncTask/Volley: I would be placing listeners on event of new expense/cluster being added. Hence, to sync them with Firebase Database, I would either use AsyncTask or Volley.
3. SyncAdapter: I am completely not sure on how Firebase manages new data request, but if required, I would be using SyncAdapter or Google Cloud Messaging to notify users of new expenses added to any given shared cluster.
4. IntentService: There is no requirement of IntentService.
5. Loader: I'm going to use (https://gist.github.com/skyfishjy/443b7448f59be978bc59) as a CursorAdapter for RecyclerView and then initialize and use Loaders for UI modifications based on changes in data.