

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any corner cases in the UX.](#)

[Describe any libraries you'll be using and share your reasoning for including them.](#)

[Describe how you will implement Google Play Services.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Firebase Auth](#)

[Task 4: Create ContentProvider](#)

[Task 5: Store data in Firebase](#)

[Task 6: Get data from Firebase](#)

[Task 7: Wizard](#)

**GitHub Username:** mAlaliSy

## My Financial Assistant

### Description

This app for helping users manage their money and record every financial process they do and store it locally and on Firebase.

### Intended User

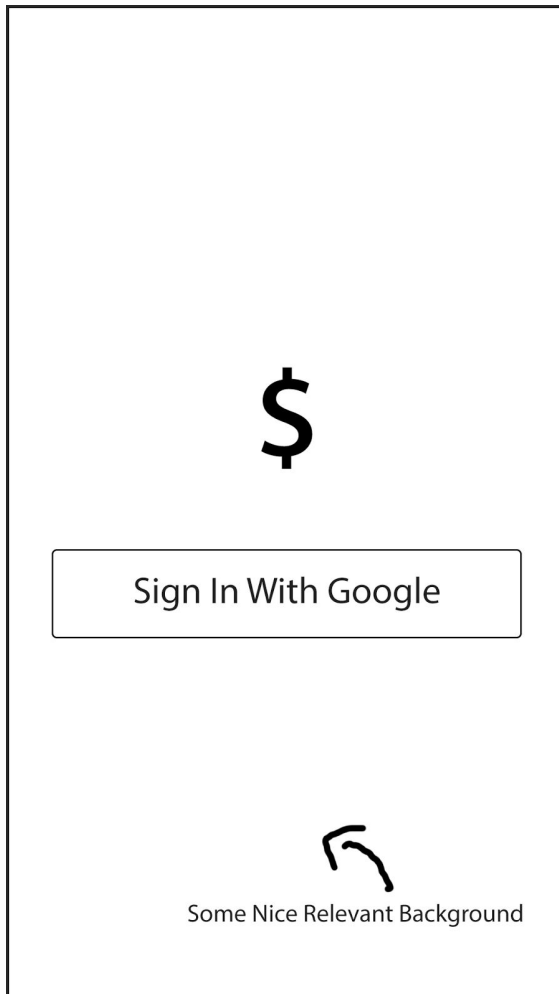
This app for everyone. Especially for those who want to record their incomes and expenses.

### Features

- The user can sign in with Google to store his data on Firebase so he can get it whenever he needs it.

- The app store any record entered locally with SQLite database and Content Provider and submit it to Firebase if the mobile has connection to the Internet.
- The app will check if there is a new data have to be submitted to Firebase when the user connect to the Internet.
- The app will show statistics and graphs for incomes and expenses.

## User Interface Mocks

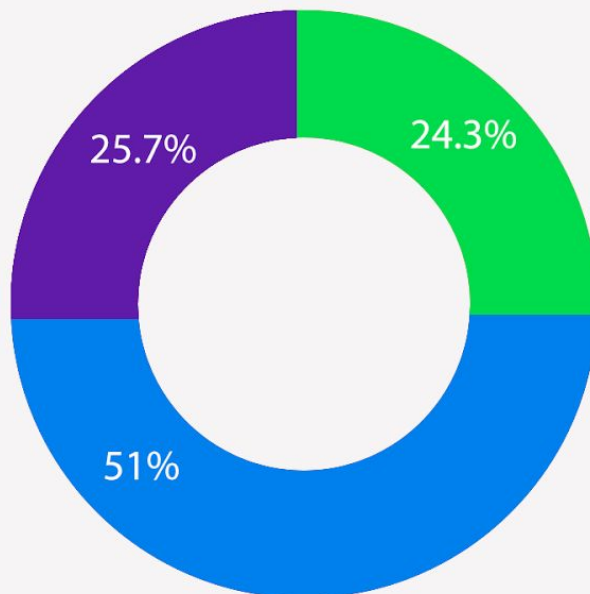


# ≡ \$ My Financial Assistant

## Overview

Total Balance	1050\$
Month's Incomes	1000\$
Month's Expenses	250\$
Month's Net	750\$

## This Month Expenses



Car	127\$
Home	60\$
Clothes	64\$



## Add Income

Source

Salary

Amount

0.0\$

Date

9/9/2016



Add

## Add Expense

Category

Home

Amount


0.0\$

Date

9/9/2016





Add

 History

Septemper 2016

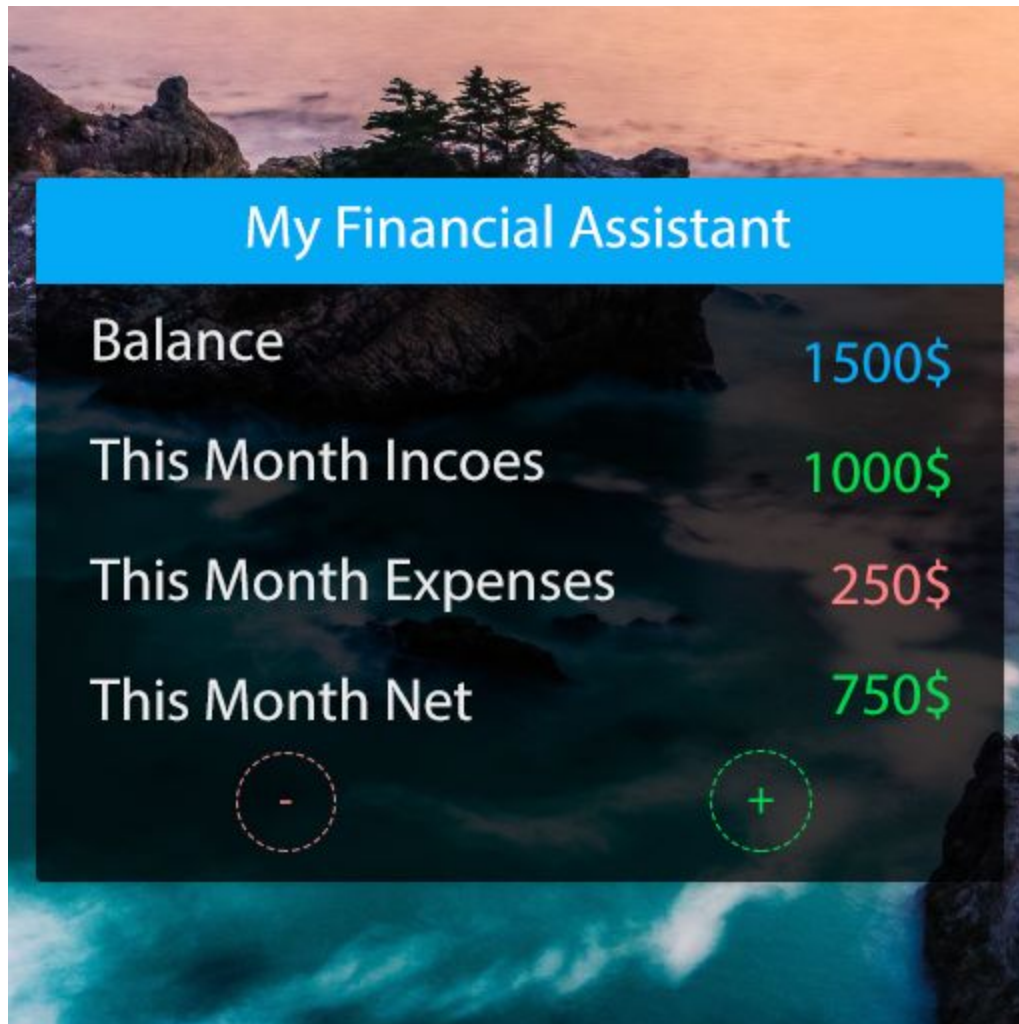
Income

Source	Salary
Amount	750\$
Date	9/9/2016



Expenses

Category	Home
Amount	750\$
Date	9/9/2016



## Key Considerations

**How will your app handle data persistence?**

The app will use SQLite database and Content Providers to store data locally and Firebase to store it on the cloud.

**Describe any corner cases in the UX.**

The app widget will make it easy for the user to add incomes and expenses. And display overview of balance and month information.

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

- [MPAndroidChart](#) to view charts and statistics.

Describe how you will implement Google Play Services.

- Firebase database to store data.
- Firebase Auth to sign in with Google.

## Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

### Task 1: Project Setup

- Create Android Studio Project.
- Initialize a repository with Git and link it to github.
- Add Libraries dependencies.

### Task 2: Implement UI for Each Activity and Fragment

- Build UI for Main Activity.
- Build UI for add income and expense.
- Build UI for History.

### Task 3: Firebase Auth

- Implement Firebase Auth to sign in with Google.
- Store account information in SQLite database.

### Task 4: Create ContentProvider

- Create ContentProvider to handle SQLite operations.



- Implement adding and editing incomes and expenses with ContentProvider.
- Fetch overview data from ContentProvider and display it in the MainActivity.
- Fetch month expenses data from ContentProvider and display it in MPAndroidChart PieChart.
- Fetch History data from content provider and display it in RecyclerView.

### **Task 5: Store Data In Firebase**

- Store new data in Firebase.
- Update any changes in data to Firebase.
- When the device connect to Internet, check if new data saved locally only and send it to Firebase.
- 

### **Task 6: Get Data From Firebase**

- When the user sign in, check if he has old data stored in Firebase and load it locally using ContentProvider.

### **Task 7: Wizard**

- Add a wizard when the user open the app for the first time.