Capstone Project: Stage 1

Description

Intended User

Features

User Interface Mocks
- Main screen
- Category details
- Add/edit a spending/income
- Settings

Key Considerations
- How will your app handle data persistence?
- 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: Create Main Activity and Fragment
- Task 3: Create Content Provider and Populate main screen's list
- Task 4: Create category details activity and fragment
- Task 5: Create "Add/Edit" activity and fragments
- Task 6: Create Setup screen
- Task 7: Final touches

GitHub Username: thelsien

MoneyTrackR

Description

Do you want to track your incomes and spendings? Then this is the app for you!
Tracking all your spendings can be a challenging thing to do. When you buy a coffee, just open this app and make a note into one the categories of your choice. You got your salary? Just put it into the the respective main category, and you are done!
Are you curious about how much you spent in a category? You can access it from the main screen. The main screen will also tell you about how much your spendings and income was in this time interval.
Are you changing your phone and want to bring your data with you? Export your spendings to a file, then import it on your new device.

Intended User

The kind of users I'm targeting are the ones who wants to keep track of their spendings. Maybe they just want to save up money, but they don't know where they keep losing so much in a month.
There are other users who believes in the "measure everything" theory. They obviously wants to know about their spendings and incomes as well. This way they can make predictions for the future. And because the data is being saved into a CSV document, they can easily can import it into a database.

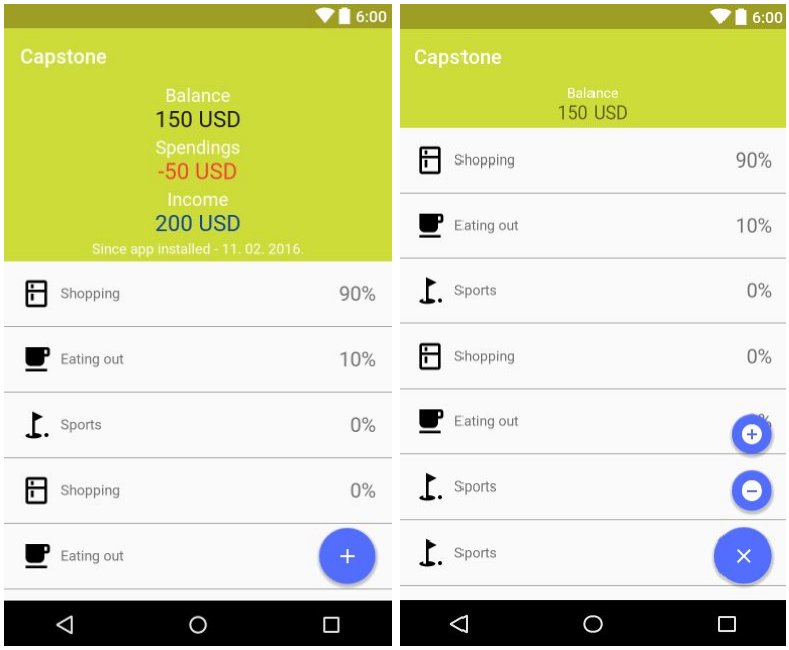
Features

- Add a spending to a categories.
- Add an income to a special 'incomes' category.
- Edit previously added spendings/incomes.
- See how much income and spending was done in a period of time.
- Select time interval to see spendings/incomes.
    - Month, year, all-time (since app installation).
- See spendings/incomes per category.
    - Selecting "Food" category in this case will show all the spendings for "Food"
- Exporting datas from the app into a CSV document.
- Importing datas into the app from a CSV document.

User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Main screen

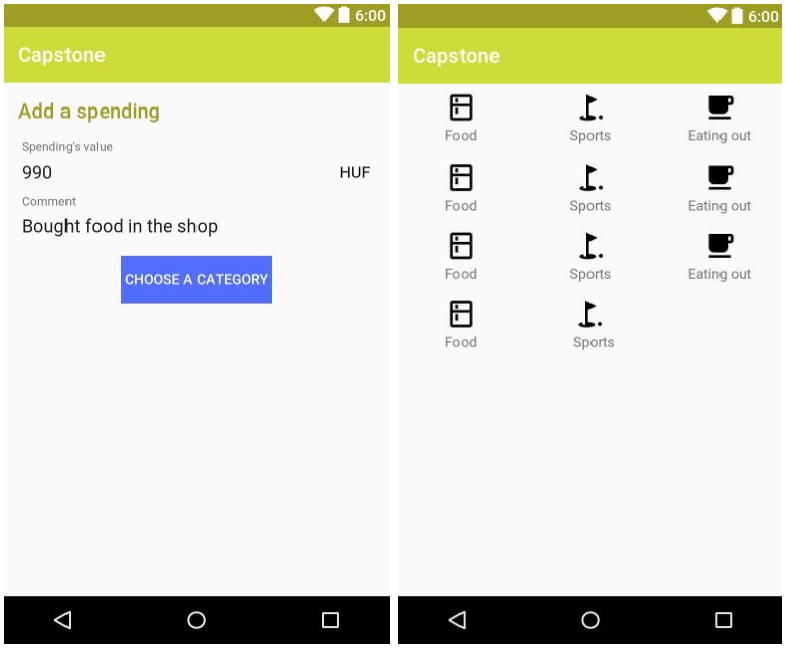


Please note that the fonts, colors, icons and namings are NOT FINAL!
The main screen will give information to the user about the spendings, the income and the current balance regarding the selected time interval. The user can see the spendings broken down to categories. In default, the values for categories are percent, however it can be changed in the settings menu.
The toolbar will work a parallax way, and when the user scrolls down, it will shrink, leaving the balance to be seen.
New spending or income can be added through floating action buttons, which are located at the bottom right of the screen.
The settings screen will be available from the top right corner (this is not visible on the design)

Category details

When the user clicks on one of the categories from the main screen, it will goes to this screen.
Top part is not parallax, it will scroll out of the view together with list items.
Every entry has a date, a value and an optional note. If the note is given, it will show up under the value. If not given, it will not show.
This screen will contain a FAB, so the user can add a spending to the given category (or income, if it is the ‘income’ category).
Clicking an entry in the list will bring the user to the edit screen, where he can change the value, note and category of the entry (or delete it altogether).

Add/edit a spending/income

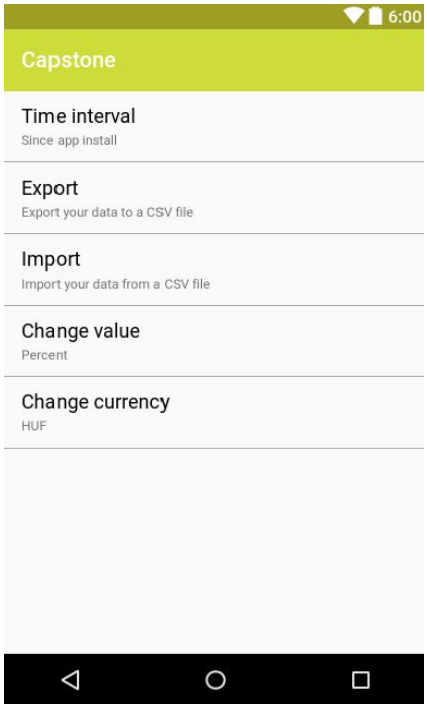


These two screen are considered to be used together, so I present them as one activity's 2 fragments.
The first screen can be accessed multiple ways:

- From the main screen's FAB (add income/spending)
- From a category details' FAB (add income the 'income' category, or add an entry to a category)
- Clicking an item on the category details (thus letting the user to edit chosen entry).

There are 2 edit texts. The first one is the value of the spending / income. Next to it the user will see the current currency. The other edit text is the note, which is optional. The value field can only contain positive integer number.
Clicking the "choose a category" button will bring the user to the other screen, where the user will select the category where he wants to add the spending. If this is an income, then the button will say "Add to incomes" and this screen will not appear. After choosing a category, the entry will be saved and goes back to the main screen.
If this is an edit screen, then a save icon will appear at the top right. The button will say "change category" (only if this is a spending, an income cannot be changed to be a spending). Values from the chosen entry will be filled into the edit texts.
If this is an edit screen, then a delete icon also appear at the top right, giving the option to the user to delete the selected entry.

Settings



This screen can be accessed from the main screen, clicking the icon on the top right.
The user can change multiple things here:

- In which the time interval he wants to see the current balance, spendings and income.
    - Year, month or from since app install
- The value shown on the main screen (percent or the actual value)
- Currency
    - I will use CurrencyLayer API to convert values from one currency to another one. Only the major currencies will be supported. (https://currencylayer.com)

The other two menu items are used to export / import data from / to a CSV file.

Widget

The widget for this app will be an Information widget. It will show the balance, spendings and incomes for the user, just like the main screen's top part. It is 2 spaces wide, and 1 spaces height minimum. If only 1 space is provided, then it only shows the current balance (for the chosen time interval). On 2 spaces height, it will show the spendings as well. On 3 spaces height, it shows the incomes as well. Clicking on the widget brings the user to the main screen for the app.

Key Considerations

How will your app handle data persistence?

I will use a content provider to store data on the phone. This app does not have a cloud endpoint, however the user has the option to import (or export) a CSV file into (from) the app. Loaders will make sure that the data is updated accordingly to the changes the user make.

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

I will use SimpleSQLProvider (https://android-arsenal.com/details/1/2512) to generate a Content provider.
I will use OkHttp (http://square.github.io/okhttp/) for making network calls easier (for the currency exchanging).

Describe how you will implement Google Play Services.

I will use the following Google Play Services:

- Analytics (track views and events)
- AdMob (interstitials only, when the user adds or modifies an entry)

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

First I will create a base project and add the library dependencies, play services. When the project builds and runs with a simple activity, I'll go to the next step.

Task 2: Create Main Activity and Fragment

- Create the Main activity
- Create the fragment for it
- Add Activity to the manifest (if necessary)
- Create layout for main activity (containing the fragment)
- Create the fragment layout.

Task 3: Create Content Provider and Populate main screen's list

- Using the library I have chosed, create the content provider.
- Create loaders to load data into the main fragment's list.

Task 4: Create category details activity and fragment

- Create new activity (add it to manifest if necessary)
- Create fragment.
- Create layout for both activity and fragment.
- Create loaders that can load data into the fragment's list.

Task 5: Create "Add/Edit" activity and fragments

- Create new activity and fragment for it (add activity to manifest if necessary)
- Create layouts for both of them.
- Create " select category" fragment and add layout for it, then populate it with items.
- Add logic for Adding an entry to a category.
- Add logic for Editing an entry.
- Add logic for deleting an entry.

Task 6: Create Setup screen

- Add icon for main screen.
- Create PreferenceFragment
- Add logic and refresh data accordingly on main screen.
- Import major currencies.
- Add networking for currency change.
    - This is for asking for currency exchange rates from CurrencyLayer API. This will happen in an AsyncTask.
- Add logic to change currency based on the values from the API call.
- Add logic for export / import data
    - The exporting and importing will happen in the background, for this I will use AsyncTask.

Task 7: Create widget

- Create layout for widget
- Declare widget in manifest

- Create AppWidgetProviderInfo
- Create an AppWidgetProvider
- Make sure that when a new entry is created or an entry is edited, the widget refreshes accordingly.

Task 8: Final touches

- More testing
- Clear hard coded strings, dimensions, colors, etc.