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.
Next Steps: Required Tasks
    Task 1: Project Setup
    Task 2: Implement UI for Each Activity and Fragment
    Task 3: Your Next Task
    Task 4: Your Next Task
    Task 5: Your Next Task

GitHub Username: crlsndrsjmnz

Currency Alerts

Description

Get alerts on the exchange rate of every world currency. It offers live information obtained from Google Finance API and continually sync this information with the device to be able to get historical data and to convert any rate offline.

* LIVE CURRENCY RATES
* FREE
* Customized notifications
* Rates from Google Finance
* Currency conversions
* Support USD, EUR, GBP, CNY, COP, ZAR
* Monitor 1 currency
* Home screen widget

Intended User

This app is intended for people who travel and need to keep track constantly of the exchange rates It's also intended for people to do business with currencies and need be alert of sudden price rise or drops based on a customer defined percentage for each currency.

Features

- LIVE CURRENCY RATES
- Customized notifications
- Rates from Google Finance
- Currency conversions
- Support USD, EUR, GBP, CNY, COP, ZAR
- Monitor 1 currency
- Home screen widget

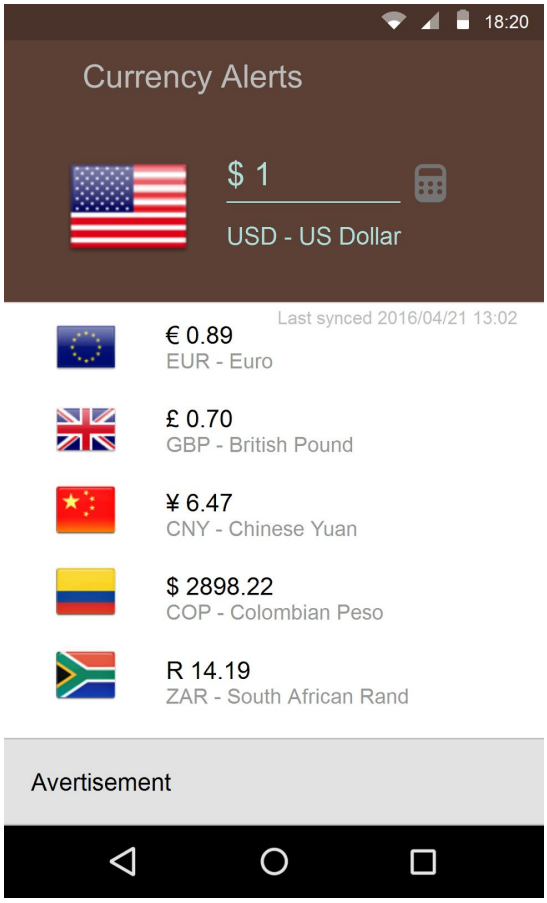
User Interface Mocks

Screen 1

This is the main screen of the app, here the user can select the main currency that will be converted to other currencies. When the user enter any amount and touch the calculator (or press enter on the keyboard), it will be converted to the other currencies based on the most recent rates synced by the adapter.

If the user selects any currency, it will open the detailed screen.

Advertisements will be present at the bottom of the screen.

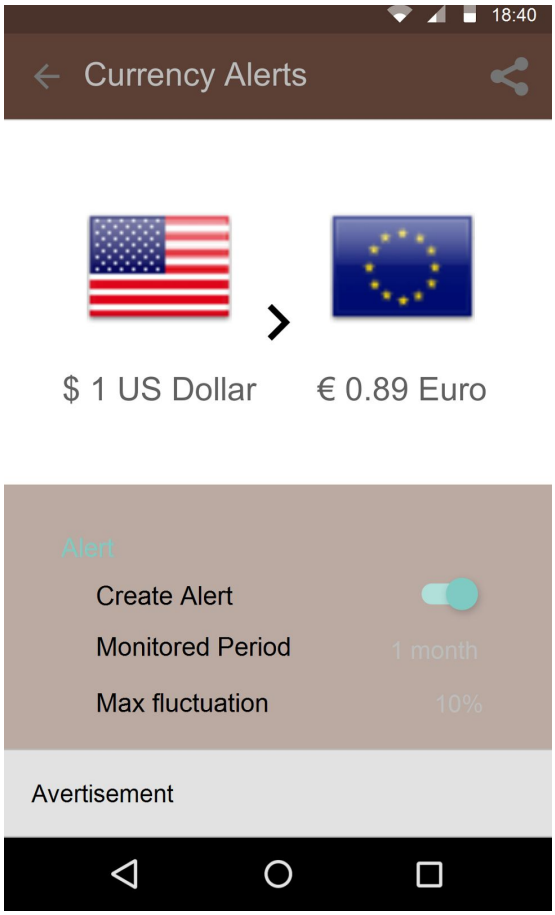


Screen 2

This is the detailed screen, here the user can activate the alerts and set the parameters for those alerts.

Here the user can also share the current exchange rate.

Advertisements will be present at the bottom of the screen.



Key Considerations

How will your app handle data persistence?

A ContentProvider will be created to store the historical info of the forex exchanges predefined, it will be upgraded to customized currencies on a future release.

Describe any corner cases in the UX.

If the user open a notification or any currency on the widget it will open the detail screen, to go back to the main screen, the user can touch the back button or back icon.

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

This app will include Picasso/Glide to handle the flag cache and visualization.

Next Steps: Required Tasks

Task 1: Project Setup

On this task, we will do the following subtasks:

- Create initial project
- Add Picasso/Glide and Google Play Services libraries

Task 2: Implement the data persistency mechanism

On this task, we will implement the mechanism to get the rates information from the API, and store it in the content provider in a way that we can easily access it from any place within the app.

Subtasks:

- Create the SyncAdapter
- Fetch information from the API
- Create the Content Provider
- Store the information on the Content Provider

Task 3: Implement UI for Each Activity and Fragment

On this task, we will implement the UI of the application, and display the information stored on the Content Provider we did on task 2.

Subtasks:

- Build UI for MainActivity
- Create a Loader to display the information from the Content Provider.
- Display the information on a Recicler View
- Add the calculator functionality
- Build UI for Detail Activity
- Add the Share functionality
- Add activity transitions
- Add accessibility support

Task 4: Implement Google Play Services functionality

On this task, we will add the ads on the Main Activity and Detail Activity.

Subtasks:

- Add ads to Main Activity
- Add ads to Detail Activity
- Add Analytics to the Main Activity
- Add Analytics to the Detail Activity

Task 5: Implement Alerts

On this task, we will implement the currency alerts that will be validated every time we sync the currencies.

Subtasks:

- Create the currency alert validation
- Create the alert if needed

Task 6: Implement Widget

On this task, we will implement the widget. This widget gets the information from the Content Provider and display the information on the Launcher.

Subtasks:

- Create widget layout
- Get the current information from the Content Provider
- Open the Detail Activity when the user clicks a currency.

Task 7: Release

On this task, we will clean the project and generate the release apk,

Subtasks:

- Clean the project
- Generate a new keystore for this app
- Generate a release APK using the new keystore