

Marino Osmanllari – Developer

This Technical Document depicts the task of designing and implementing the "Stocks" app, a fully functional Android mobile application. The developed application provides users with real-time information about crypto currencies.

We focused on incorporating the main functionalities outlined in the assignment. In this technical document though are mentioned the rest of the functionalities.

- User Interface

The User Interface of the application is quite simple. The main page is where all the currencies are displayed by the API JSON response. They have four important pieces of data: id, name, symbol and current price. The favorites page is where the user has all the favorite currencies selected and the last page is the Details Page where more detailed information is displayed for the clicked currency by the user, including past data, image, 24h changes etc.

There are used 3 different Layouts:

- RelativeLayout, a view group that allows positioning and aligning child views relative to each other or even the parent view, used in every layout xml file of the project.
- LinearLayout, a view group that arranges child views in a linear orientation, either horizontally or vertically, used for the main page data.
- CardView container view that provides a visually appealing card-like appearance for its child views, with features like rounded corners, elevation, and shadows.

- RecyclerView

The RecyclerView is a powerful UI component in AndroidStudio that efficiently displays a collection of items in a scrollable list. The RecyclerView is connected to the Heroku API to get back the list of items displayed as a list, where in this case they look like cards.

The RecyclerView widget is in the activity_main.xml file where the list of currencies displayed and the layout manager defined is RelativeLayout. The CurrencyRVAdapter specifies the custom ViewHolder class and has the necessary methods such as onCreateViewHolder(), onBindViewHolder(), and getItemCount(). In the onCreateViewHolder, the layout is inflated for each item view and returns a new instance of the ViewHolder. The onBindViewHolder binds the data from the API to the views in the ViewHolder based on the item position. And the getItemCount returns the total number of currencies/items in the list. Thereafter, since the API is ready to fetch data, the adapter and data are set to the RecyclerView and the fetched currency data are passed to the adapter to populate the views in RecyclerView.

- Navigation bar

The navigation bar consists of a rectangle which has been designed to be with round corners and stay at the bottom for ease of use. The 3 pages consist of Home Page, Favorites and Details. The icons are downloaded as Vector Assets from AndroidStudio in the drawable folder where each one of them is linked to the mentioned pages. First 2 pages are navigable which means the user can click them to navigate through to the other page, however the Details page is used in the navigation bar as a reload button and makes a new request to the API for a new update. So the user can click on it to refresh the Home Page or click to any of the currencies to see even more details about the currency clicked.

- Search Functionality

In the CurrencyRVAdapter class is provided a filter list to update the dataset of CurrencyRVModal objects. It takes in an ArrayList of filtered items, replaces the existing dataset with the filtered list, and notifies the adapter that the data has changed. This triggers a rebind and refresh of the RecyclerView to display the filtered items.

To search a specific currency in the MainActivity java file it filters the currencies based on this specific keyword. It takes in a currency parameter, which represents the keyword to filter by and displays the currency or all of them that have the same keyword.