Ananya Jude

CS 443 - Mobile Applications

Final Project

Professor Bo Sheng

December 21st, 2021

CRYPTO CURRENCY TRACKER

Project Description Document

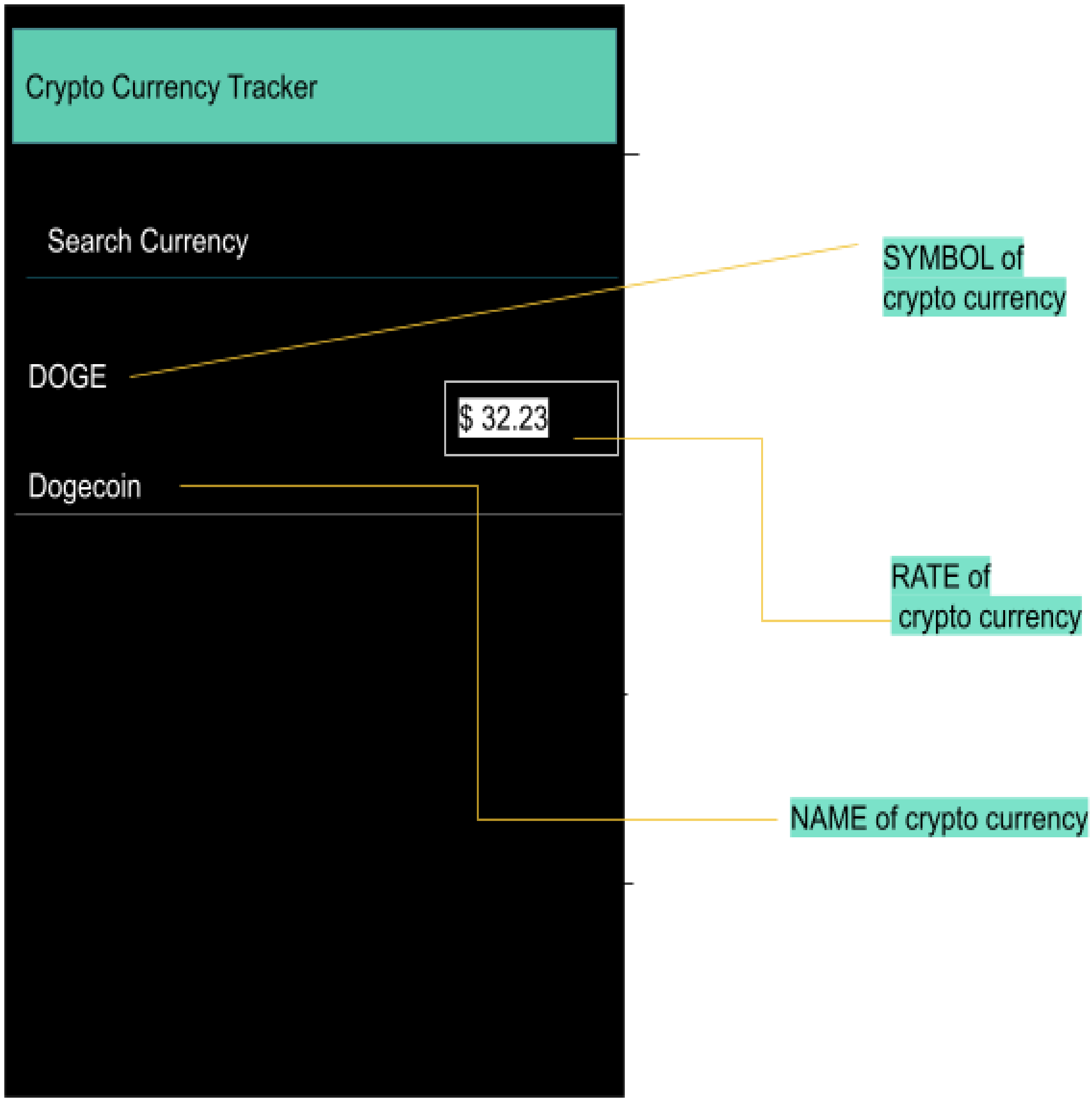
1. Project Statement :

I started off working on the final project for this class with a partner and the app was supposed to be a Trip Expense Manager. During the start of our semester we had gone through topics and tried to see what could be useful and something we could implement and have fully functioning as a project. Towards the end, due to different thoughts and opinions, I had to split from the team project and start an individual project by myself. I had to think carefully about what I learnt in class to be able to complete the project in a day. Earlier this year, I had taken a crypto currency class in Spring and thought about implementing a crypto currency tracker. The CryptoCurrency Tracker is a basic app which displays the name of the crypto currency, symbol of the crypto currency and the rate of the currency in dollars. This app allows the user to search for the particular crypto currency they are interested in and displays the above information in real time. I was able to execute this using an API call from the coinmarketcap website.

2. Application Design :

The first thing I did was go to the colors.xml file to set up the colors I want to use for the app. I added three black shades along with their hash numbers and set the pre-existing colors mostly to teal color, but I left the primary black and white the same way it was. The next thing I did was go to the build.gradle file and add Volley under dependencies. Volley enables the app to connect to the internet from the device. To make this work efficiently, I added the internet permission in the AndroidManifest.xml file.

In the activity_main.xml file which is under layout, I defined RelativeLayout which includes EditText for searching the currency, RecyclerView to display the list of crypto currencies and a progress bar that acts as a loading indicator. For each of these categories in Relative Layout, I modified the layouts, such as their height, width, margin, and added a prompt (through android:hint) to Edit Text. I also generated ids for each of them. I was able to display the information relevant to the crypto currencies with the help of APIs. I used an API call from the coinmarketcap website for the real-time information.



The above is a design for the app showing the crypto currency symbol, name and the rate in dollars.

3. Application Implementation and Evaluation :

This was the main part of my project, creating a new java file for the Currency RecyclerView Adapter (CurrencyRVAdapter) class. I created the RecyclerView in the activity_main.xml file which has the basic structure for acquiring the information on the crypto currency. This Adapter class is used for setting data to each item of our Recycler View.

In this class, I first created the adapter class (CurrencyRVAdapter) which extends the RecyclerView and passed the array list along with the ViewHolder class created. I created a filterList method which adds the filtered list to the array list and notifies if the data set has changed. I also created methods that inflate the layout file created for the recycler view, get the views and set the data to the particular item of the recycler view and the other views eventually and another method that returns the size of the array list. The CurrencyViewHolder method is created to initialize each view of our layout file, by initializing all the text views (symbolTV, nameTV, rateTV) along with their ids. The next step is extremely essential in making the app work and that is generating the API keys and getting the URL to fetch data through JSON. I generated the API key from coinmarketcap and with the help of postman, was able to add it to the code using JSON format.

In the MainActivity.java file, I first created variables for the array list, progress bar and recycler view and created an Array List of CurrentModel. After, I initialized the variables, the array list and the adapter class. Then, I set the layout manager to recycler view and adapter to recycler view. After getting the data from the API, I add the text watcher to edit text to check the data entered in edittext. The filter method has an array list that stores filtered data. Under the getdata() method, we create variables that store the URL and the request queue, along with the JSON object that gathers data from the API. In the onResponse method, I extract the data from response and pass it to the array list, during which I set the progress bar to gone. In the getHeaders() method, I pass the headers(postman) as the key, and the API key as the value.

4. References :

➔ Google

I used google to find the website for the API, for the app icon and for color hex numbers/ hash numbers.

➔ PostMan

- I used postman (website) to run the API calls to view the respective JSON file to access data from it in our MainActivity.java.

➔ Youtube

- I used youtube to refer to how to access the API keys using postman.

➔ Lecture and Homework code as a reference.

➔ GeeksforGeeks

5. Experiences and Thoughts :

I was excited about this project since the start of the semester because this class is all about android mobile applications. Towards the end, I did have a hard time with my partner which set me back a lot. I tried to do the best I can in this project and understand, learn and enjoy this process. I wish I could have done some more but because of this last minute change and time constraint, I was limited. I still think I have done the best I could given my situation and I hope my hard work pays off. I wanted to add a change in stock by indicating the increasing in price indicating it as green and a decreasing in price in red. Everytime, the page is refreshed it would show these changes on the spot. My app still shows the current values but not the way I had wished to depict it.