App Description

Title: Crypt-O'-Currency

Developer: Robert Tunn, 2015065

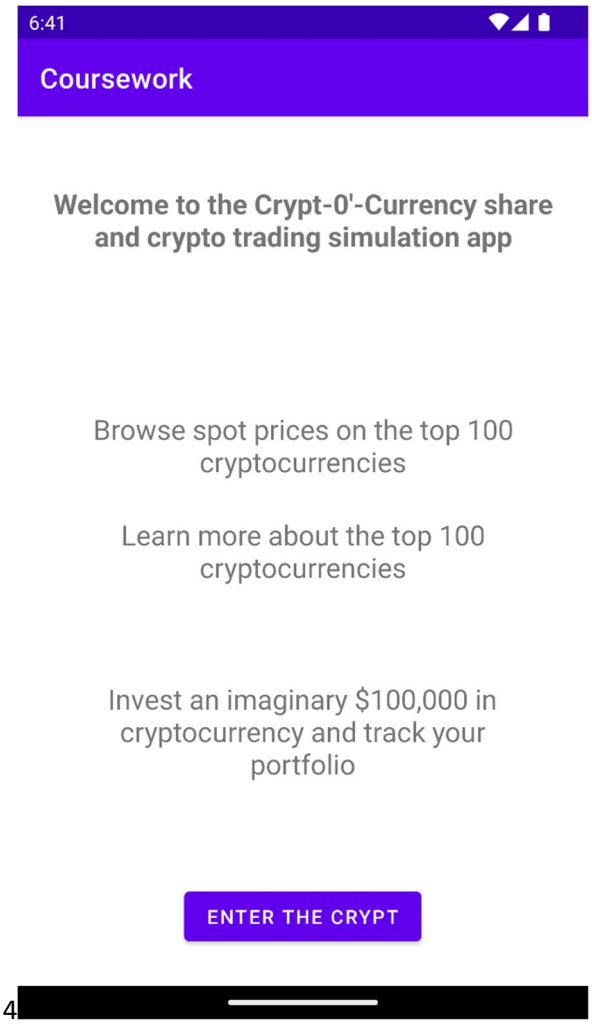
Promo: Cryptocurrency Trading Simulator

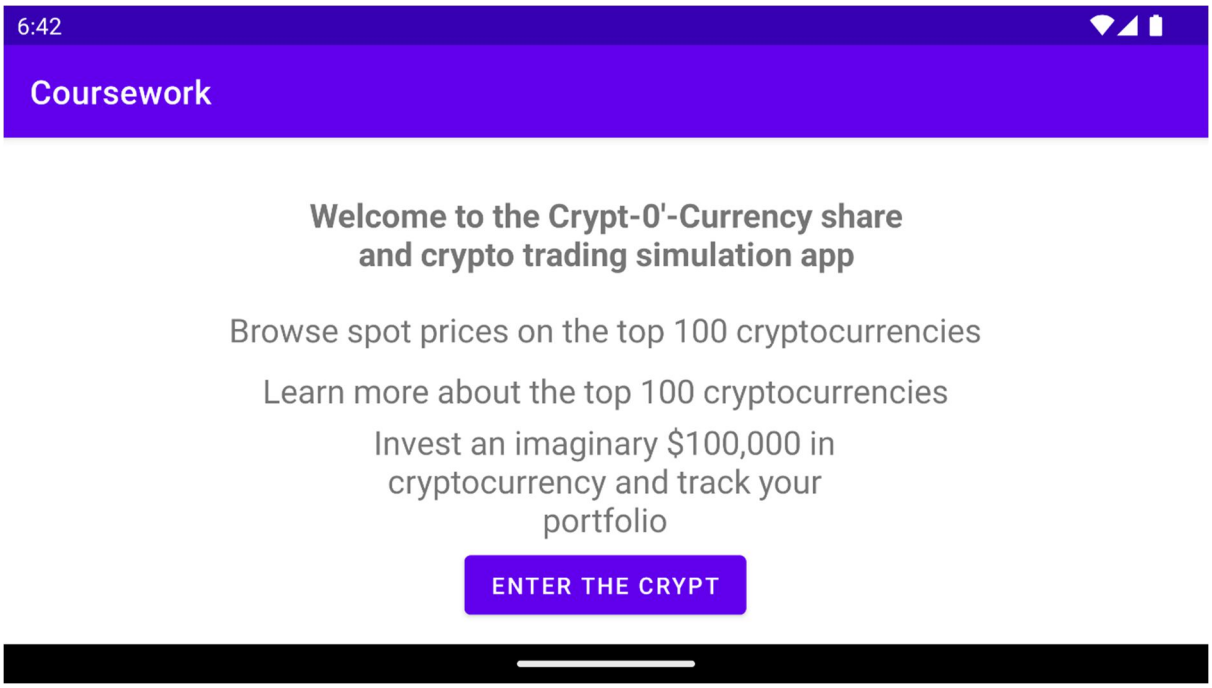
- Browse crypto spot prices (data taken from CoinMarketCap API);
- Learn about cryptocurrency; and
- Trade with an imaginary $100,000

App Design

I intended to create a design where the user entered a darkened crypt but later decided to concentrate on the code.

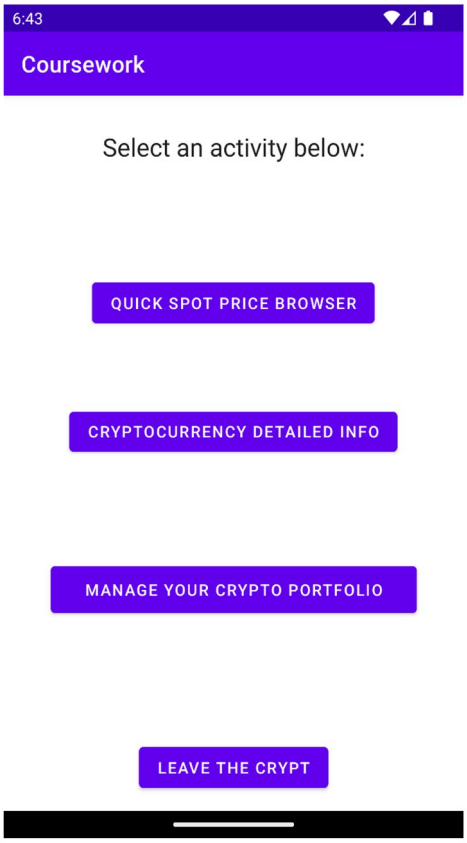
The user begins at the welcome page which describes what the app does:

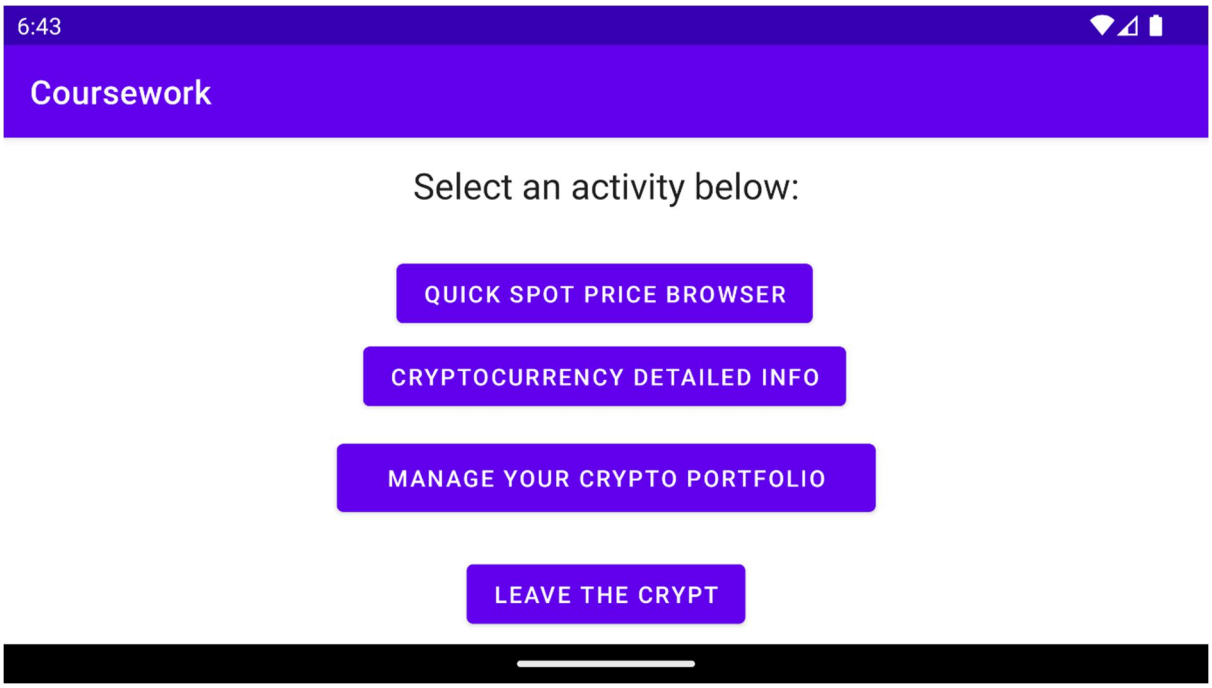




All pages were created in portrait and landscape format. Portrait always seems to look better.

Enter the Crypt button takes user to a navigation page:



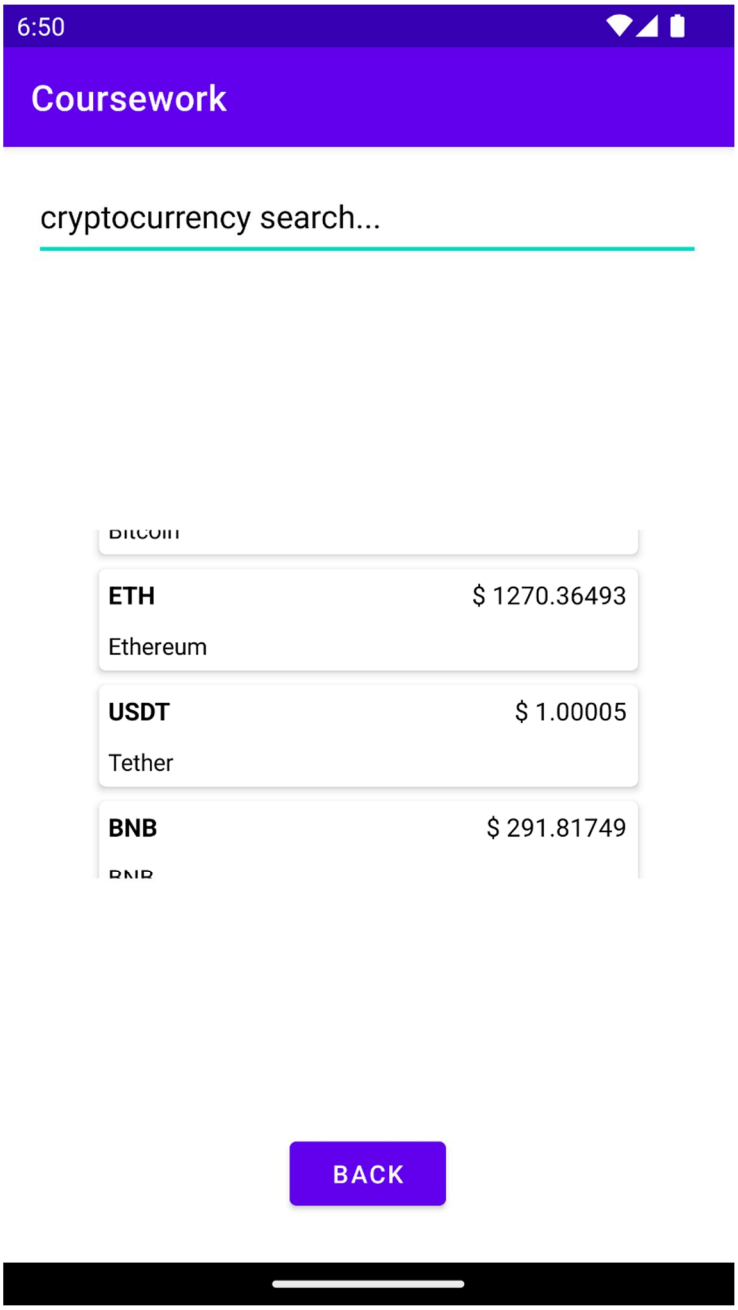


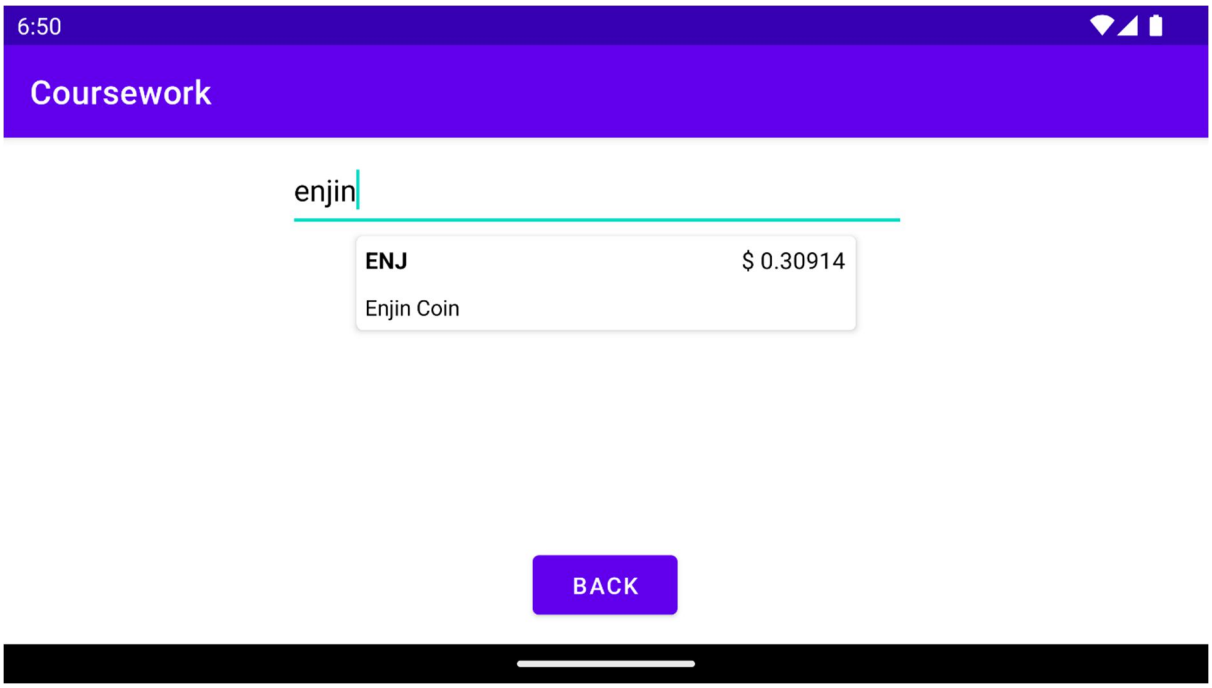
From here the user either decides to leave the crypt or proceed to:

- Look at the spot prices on the top 100 cryptos by market value;
- Learn more about a cryptocurrency; or
- Create and manage a cryptocurrency portfolio using their imaginary $100,000 that comes with the app. All amounts are in USD.

Quick Spot Price Browser

Getting the API parsed properly and working was a challenge. The call to the CoinMarketCap API is made within the onCreate() function meaning that the info is pulled as the page is loaded by the smartphone. This also ensures that prices are updated with each page load given the rapidly fluctuating spot prices.





The user can either scroll through the top 100 cryptocurrencies or use the search bar. The five decimal places are required for cryptocurrency trading.

The information is presented in a RecyclerView. Search results are limited to one item and can be made against the ticker or name.

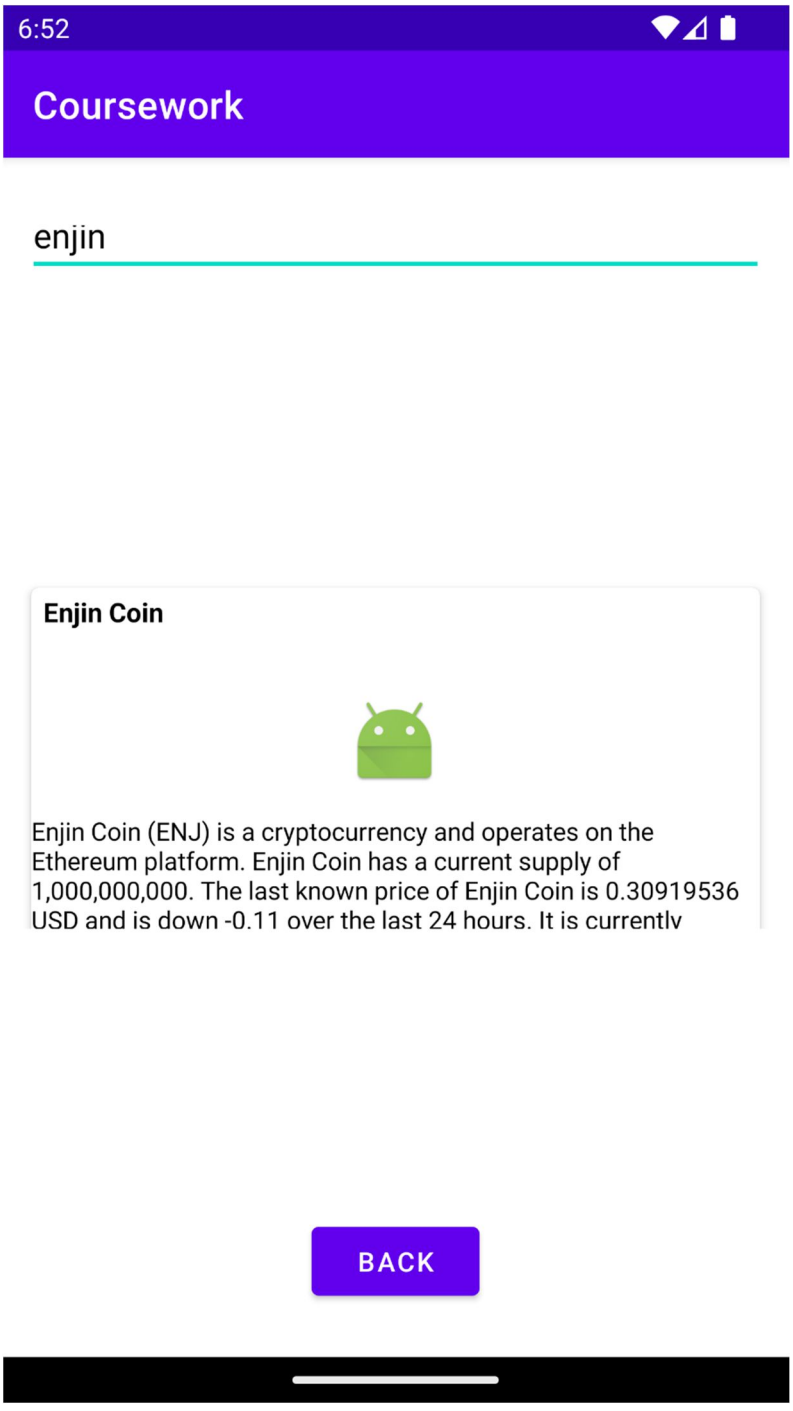
In order to achieve this functionality at the back end I created an object class namely Cryptocurrency with name, ticker, spot price etc as fields.

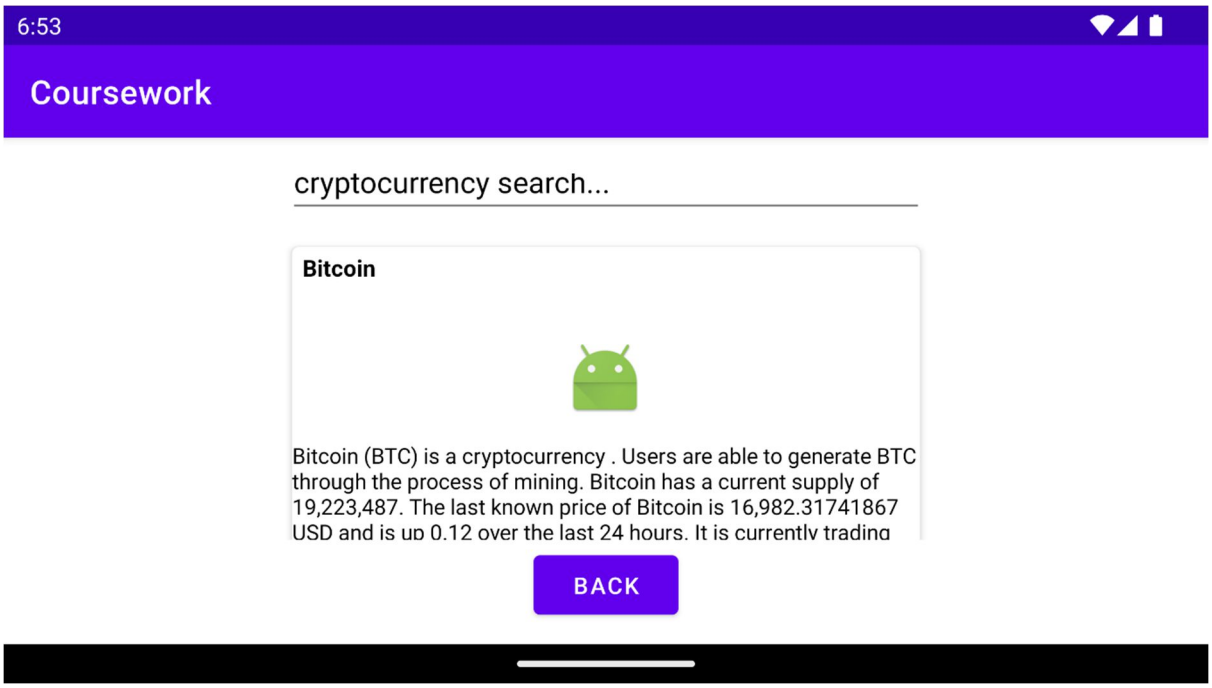
This page provides the user with a quick reference to current spot prices only.

Cryptocurrency Detailed Info

The CoinMarketCap API gives around a paragraph worth of information on each cryptocurrency including a url link to its logo, its website, metrics e.g. number of coins in circulation and market cap, pre-mined or mined etc.

The user can learn more about a cryptocurrency before making a decision on whether to invest or not:



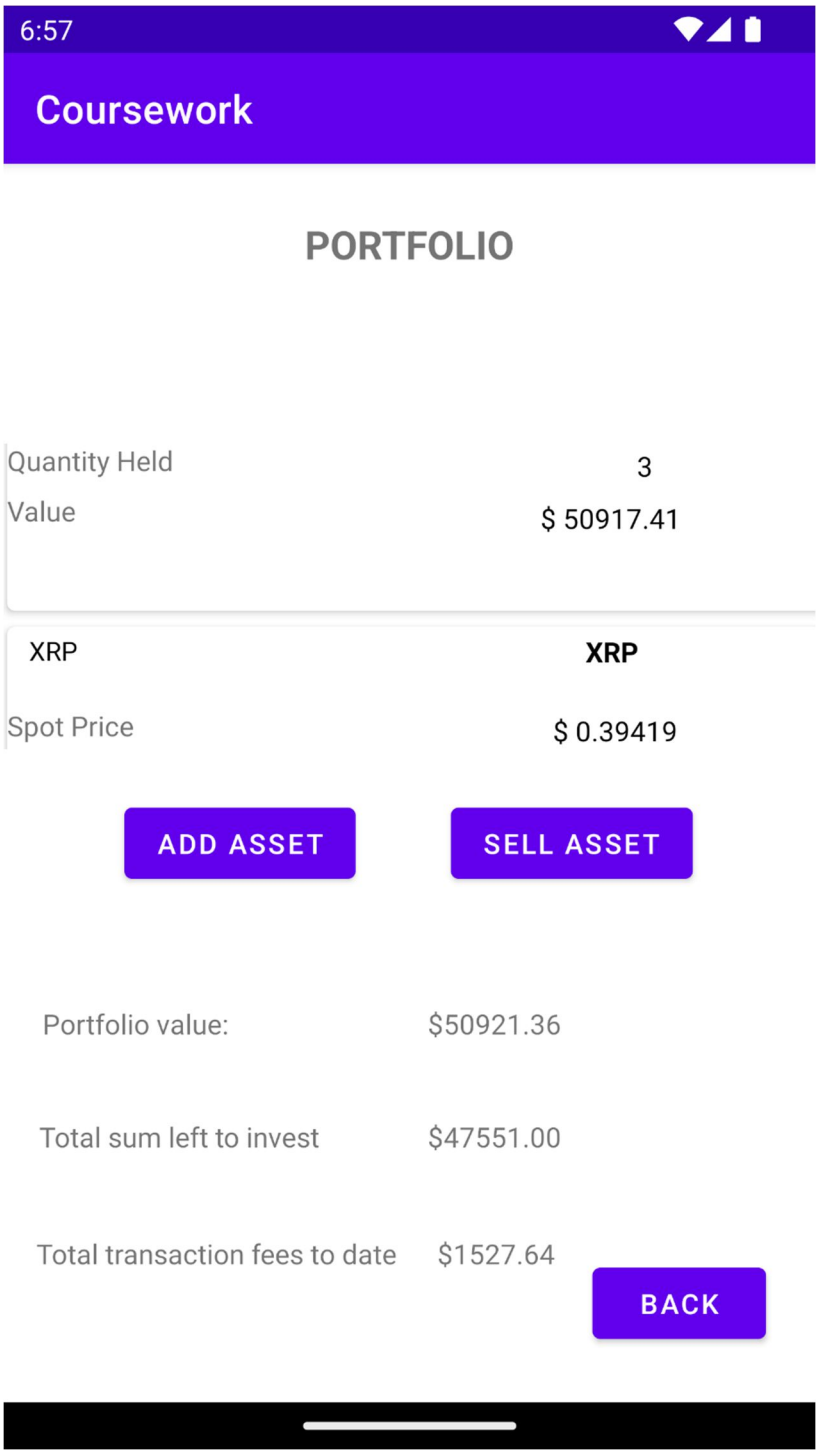


Again, a RecyclerView is used and you can search for the cryptocurrency that you want more info on. Results are again limited to one item. The code for this page began to get out of hand so I restricted the results to a handful of cryptocurrencies, I do not list 100 as stated on the welcome page.

Also, I did not get as far as divining a way to pull the url for the logo by API call and automatically populate the RecyclerView with the image associated with the url, hence the multiple alien's heads.

Manage Your Crypto Portfolio

The coding begins to get more involved at the back end. I created the object classes of Portfolio and PortfolioObject to keep track of what the user currently has in their portfolio, portfolio metrics and each PortfolioObject.



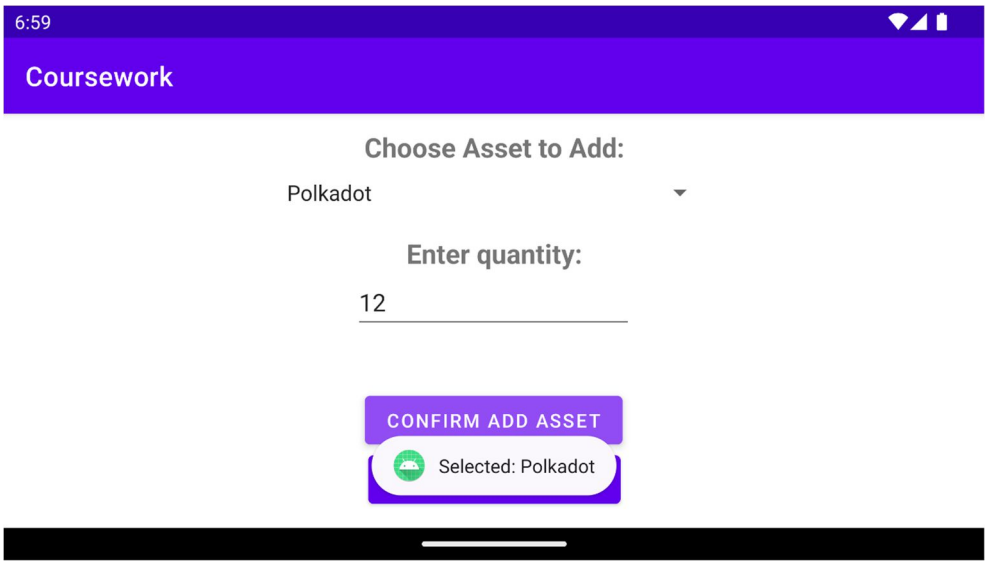
Portfolio items are listed in RecyclerView. The user has purchased three Bitcoins and some XRP. Portfolio value, based on current spot price, is $50,921.36. They have $47,551.00 left to invest and they have spent $1527.64 on transaction fees.

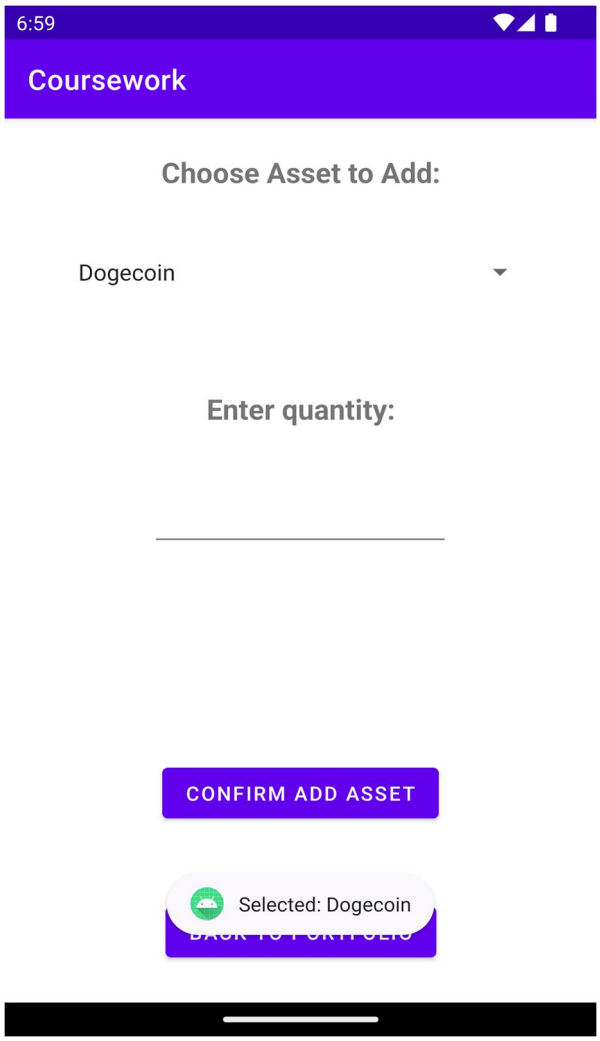
Please note that these transaction fees are not realistic rather a flat rate of 3% on the purchase and again on the sale is hard coded into the page Adapters. I cannot justify a massive monthly subscription for the full CoinMarketCap API for the actual fees so 3% will have to do for the coursework! This functionality would be easy achieved if I decide to take the app to market. I would

also add a system for alerting the user of big price moves and allow them to buy more imaginary credit to make it profitable.

This page leads to the Add Asset and Sell Asset Pages.

Add Asset



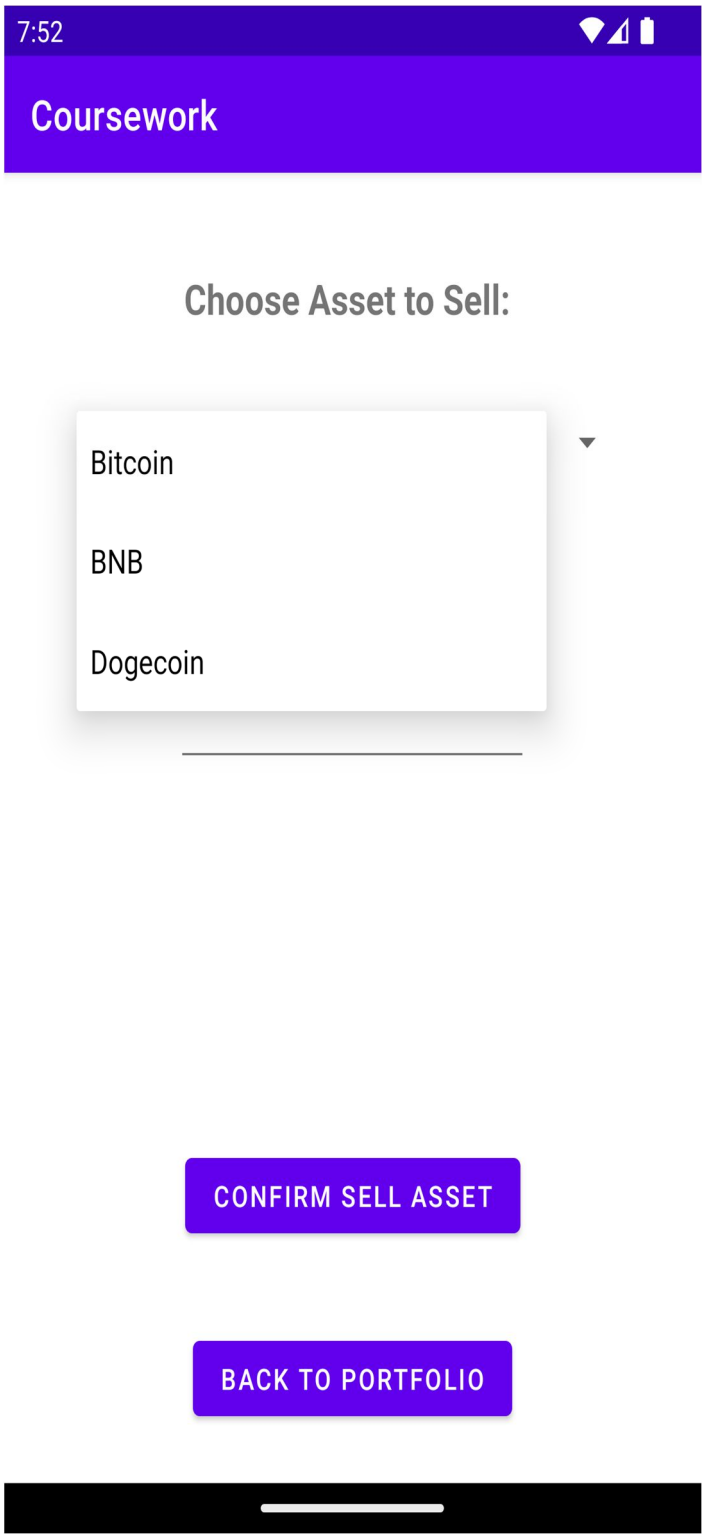


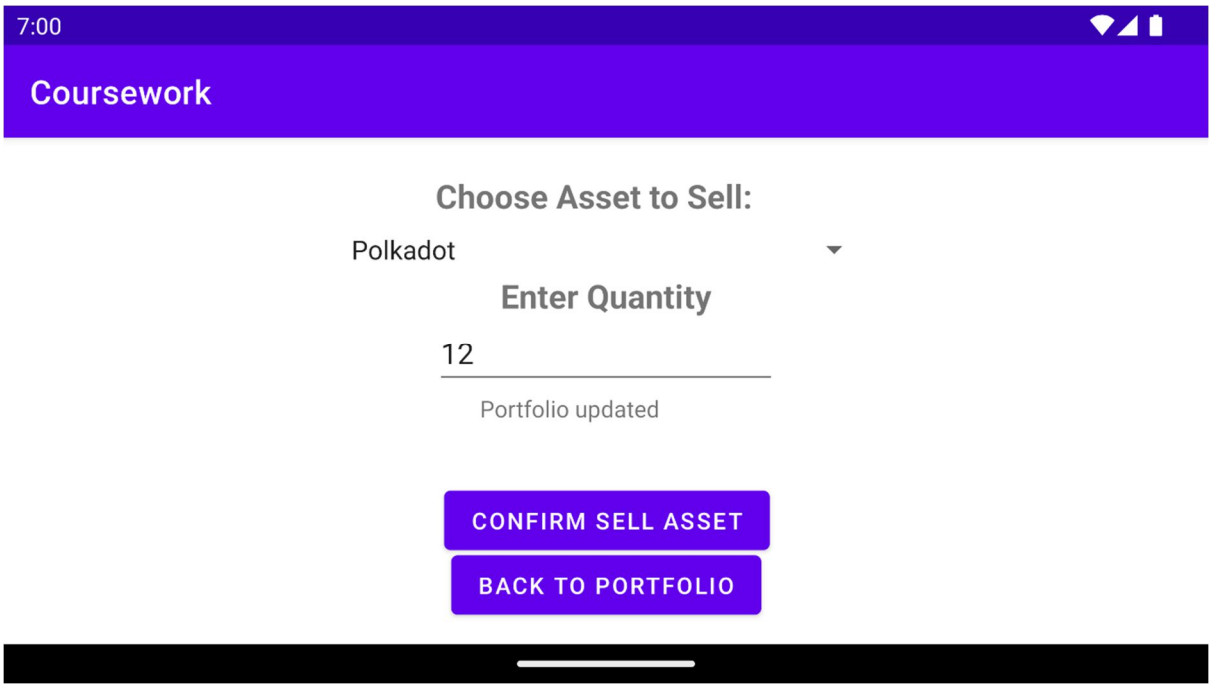
The spinner is automatically populated with the names of the top 100 cryptocurrencies which hopefully the user has researched before adding to their portfolio. The cryptocurrency is chosen and the amount entered in the EditText. The user is given a message by updated TextBox if the purchase was successful and, hence, the portfolio updated or not.

The page Adapter is coded to prevent the user from buying more than their current balance allows. When the user returns to the portfolio screen their assets and portfolio metrics are updated.

One limitation is that the amount to be purchased is limited to integer values. I had intended to make this a floating point as most cryptocurrencies can be bought and sold in portions. Alas the EditText box in the layout file only handles integers which was realised too late.

Sell Asset


7:52
Coursework
Choose Asset to Sell:
Bitcoin
BNB
Dogecoin
CONFIRM SELL ASSET
BACK TO PORTFOLIO



If the portfolio is empty the spinner will say that there are no assets to sell blocking further progress on this page.

As in the top view, a list of cryptocurrency assets are given in name only. The user must choose one and enter the quantity that they wish to sell before confirming by button click. As with Add Asset, this amount is limited to integer values. Also, the Page Adapter code prevents more of an asset from being sold than the user holds.

Upon return to the portfolio page the portfolio assets and the metrics of portfolio value, amount left to invest and transaction fees accrued to date are updated.

Reflection

All code has been tested and apart from the limits on functionality everything works as expected. I have certainly downloaded worse apps than Crypt-O'-Currency.

The app is supposed to be a fun and educational experience and I think that is achieved. I wish I had more time for the design. I have come up with something that I am reasonably proud of. I could go on forever with this and I can think of many more functionalities that could be achieved with the app in terms of putting it into real world use. It could even be addictive…

Robert Tunn, 2015065

Total word count: 999