phidex

Project Report

Team Skynet

Computer Science Project COMP[39]900

Due: Friday, 25th of May 2018

Elliot Griffiths	e.griffiths@unsw.edu.au	z3332776	Scrum Master / Developer
Emma Young	emma.young@student.unsw.edu.au	z5017860	Developer
Patryk Labuzek	patryk.labuzek@student.unsw.edu.au	z5198690	Developer
Cameron Hillman	c.hillman@student.unsw.edu.au	z5062541	Developer

1. Background

A stock exchange is an *"organized and regulated financial market where securities (bonds, notes, shares) are bought and sold at prices governed by the forces of demand and supply"*.[1] It dates back to the early 1600s where the Dutch East India Company became the first organisation to issue shares of stock and bonds to the general public. Since then the popularity and size of stock exchanges has grown significantly.

The stock exchange is constantly evolving and much has changed since its origins, especially in recent years with the emergence of cryptocurrencies. The price of crypto tokens is set just like in stock exchange, by supply and demand. As of 2017, over a thousand cryptocurrency coins exist and are turning the financial services space upside-down because of their growing influence in changing marketplace dynamics.

Some believe that cryptocurrencies are the way of the future, and invest in them due to their strong belief in its future value for society, while others invest for the opportunity to make quick short-term profits. Looking at the statistics in recent years, cryptocurrencies have proven to be a very good investment for those who got in early - in 2017 alone, there was an increase in value of 1,500% in Bitcoin and over 10,000% in Ethereum. More and more new "Cryptos" are emerging that offer even higher possible returns. With such a high potential, it's no surprise Cryptocurrencies attract both finance/investment professionals and general public.

It is because of the popularity and demand of Cryptos that our team aims to implement a Cryptocurrency Portfolio Management app (Phidex), rather than the stock-oriented option. With all the possible crypto coins on the market, buying multiple currencies to diversify your portfolio and manage risk is very common, hence managing and tracking your coins becomes an issue. A Cryptocurrency Portfolio Management application is built specially to overcome this obstacle by enabling the user to add and delete owned coins to a single portfolio.

[1] http://www.businessdictionary.com/definition/stock-exchange.html

2. Aim

Our goal was to create a minimalistic, user friendly and effective portfolio application for tracking cryptocurrencies. We strived to provide a high level of utility while also upholding sound usability and aesthetics.

Google Play Store offers several effective and usable interfaces. However the majority of which posses narrowly targeted goals while leaving additional features lacking in utility.

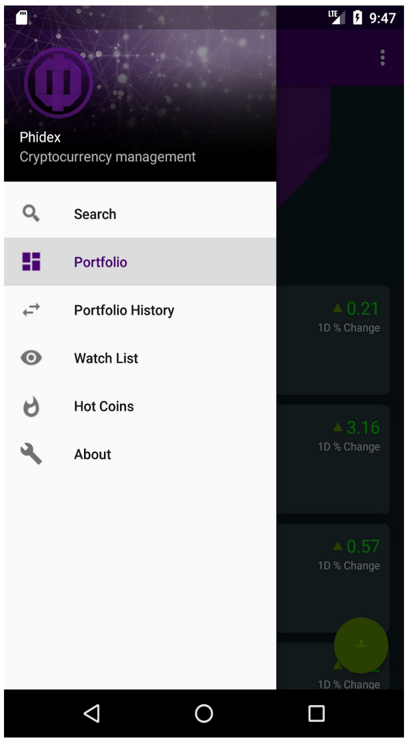
Our aim with Phidex was to address three of these prominent issues:

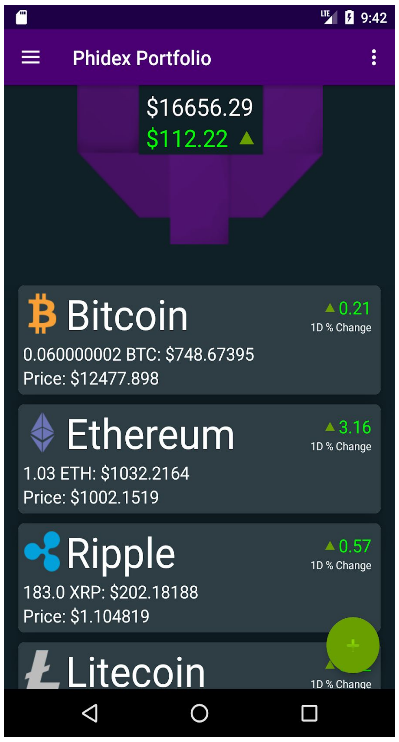
1. Community discussion
    - News and discussion will be outsourced from active user generated and moderated forums. This will assist with the rising issue of ‘fake news’ in today’s media.
2. Identifying early interest
    - Finding a coin, ICO or token early in its life cycle is extremely important. Phidex will be able to identify times of high interest through the user activity changes observed in the individual coin’s news section. This is made possible through the implementation of the highly popular user generated discussion board.
3. Comparing net price and identifying arbitrage
    - With hundreds of exchanges offering a wide range of coin prices, bid/ask spreads, percentage commision and transfer fees, it can be extremely difficult to know the best exchange to purchase and transfer out a specific coin. As such, Phidex will display a sorted list of the net total cost of purchase from several popular exchanges.

Although our initial aim was to address the above points, we found we needed to limit the scope of the project regarding price comparison and arbitrage which we will discussed in section 7.

3. Design Overview

As a team we have put every effort into delivering an eye-catching and user-friendly design. To achieve this we have closely followed Google's material design guidelines[2] which highlight the pros and cons of every design decision, as well as providing up-to-date, general design standards used by app developers. Following an existing set of design guidelines that are widely used across the Android app ecosystem which also makes our user interface intuitively easy to use. The most important design aspects implemented in the project include: the persistent side menu (discussed in section 8), a hybrid of Card View and Recycler View, which combined result in a user friendly and eye catching scrollable panel, the floating action button - when clicked opens a pop-up window. Putting all of these aspects together resulted in an intuitive design and user-flow.





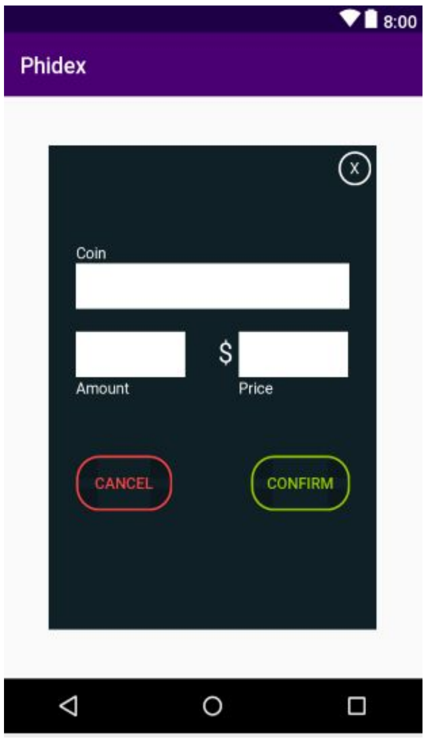
$16656.29
$112.22 ▲

Bitcoin ▲0.21 1D % Change
0.060000002 BTC: $748.67395
Price: $12477.898

Ethereum ▲3.16 1D % Change
1.03 ETH: $1032.2164
Price: $1002.1519

Ripple ▲0.57 1D % Change
183.0 XRP: $202.18188
Price: $1.104819

Litecoin 1D % Change



[2] https://developer.android.com/design/index.html

4. Functionality Overview

Upon opening the app, the user is first presented with a list of coins that they currently have in their portfolio. From this main menu, they may add a new coin via the bottom-right “+” icon, which allows the user to search for a coin from a predefined list and then enter the amount of coins that they bought as well as the price they bought it for.

From the sidebar menu, the user may navigate to a number of pages:

- The search page, where a predefined list of coins can be filtered down using the search bar, and can be clicked on to take them to the coin’s detailed information page.
- The watchlist page, where a user may, similar to the way they add a coin to their portfolio, add a coin that they are “watching”, for coinst that the user may want to track the price of but does not want it in their portfolio.
- The hot coins page, where a user may see a list of currently trending coins based off of the increased frequency of news about that coin.
- The transactions page, where a user may see all of the transactions they have entered into the app.

From all of the above-mentioned pages, a user may navigate to the information page for a specific coin. Here they are able to access via a tab menu the coin’s statistics, as well as a price history chart, the transactions history, and list of recent news.

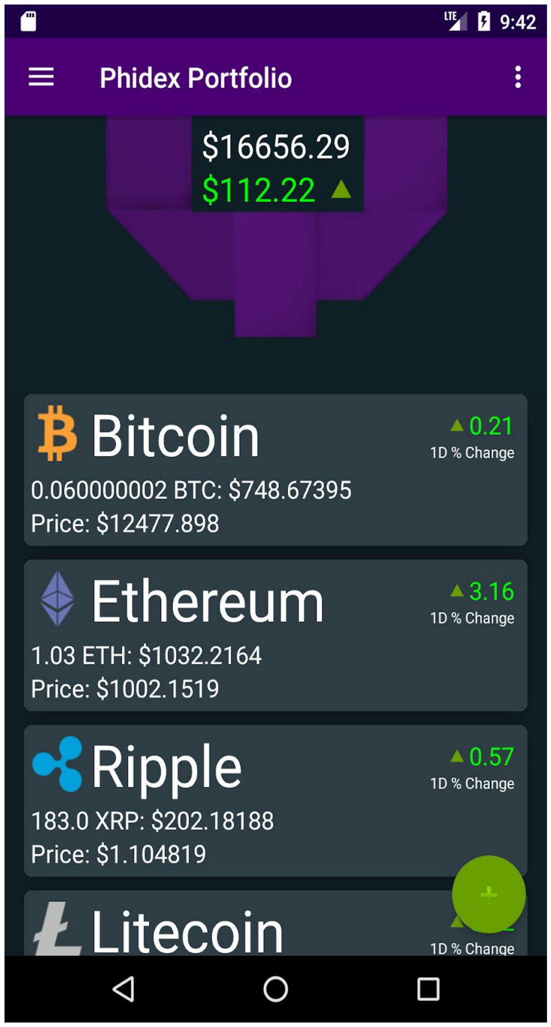
5. Implemented Functions

5.1 Portfolio

The portfolio screen displays an overview of the users current cryptocurrency holdings. This screen is the main page of our application and maintains a minimalist and aesthetic design while displaying the following information:

- Portfolio balance
- Profit & Loss
- Holdings list

Critical to the application, the portfolio page includes a manual refresh function for synchronising with the latest market data.
The persistent sidebar menu is accessible at any time from the Portfolio Activity. It provides a quick and easy navigation to the main components of our app, that is: Search page, Portfolio History, Watchlist, Hot Coins and About page.
Additionally, from the portfolio page users are able to add coins to their portfolio by clicking on the floating "plus" action button. Once a coin has been added it will appear as a card in the scrollable panel with the most important information about this coin.
All coins in the holdings list are interactive and take users to their respective coin page, that provides in-detail information about the coin.



9:42

Phidex Portfolio

$16656.29
$112.22 ▲

Bitcoin ▲0.21 1D % Change
0.060000002 BTC: $748.67395
Price: $12477.898

Ethereum ▲3.16 1D % Change
1.03 ETH: $1032.2164
Price: $1002.1519

Ripple ▲0.57 1D % Change
183.0 XRP: $202.18188
Price: $1.104819

Litecoin 1D % Change

5.2 Search

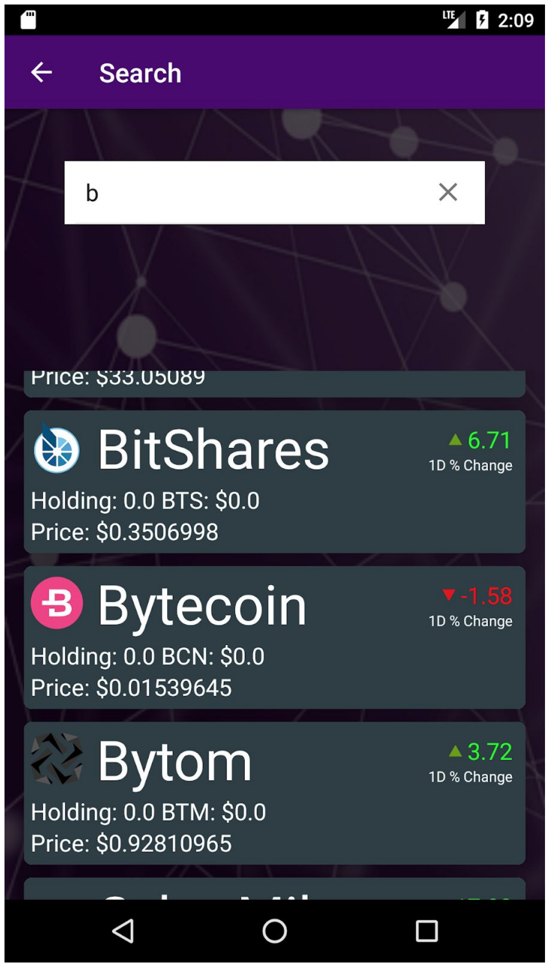
The Search page contains a scrollable list of the top 100 cryptocurrency coins. This list is generated from the room database which in turn pulls data from coinmarketcap.com.

Each coin has its own clickable card displaying:

1. Logo
2. Coin name
3. 24 hour percent change
4. Number currently holding in portfolio
5. Value currently holding in portfolio
6. Current price of the coin.

A single click on a card will take you to the View Coin page outlined in section 5.3.

The main functionality of this page is the ability to filter the coin list down by entering text into the search box. The filtering is done by pattern matching the coin's name with the entered string. The typed substring can match with a series of characters starting at any point in the coin name. For example searching for "coin" or "bit" will both match with "bitcoin"



Price: $33.05089

BitShares ▲6.71 1D % Change
Holding: 0.0 BTS: $0.0
Price: $0.3506998

Bytecoin ▼-1.58 1D % Change
Holding: 0.0 BCN: $0.0
Price: $0.01539645

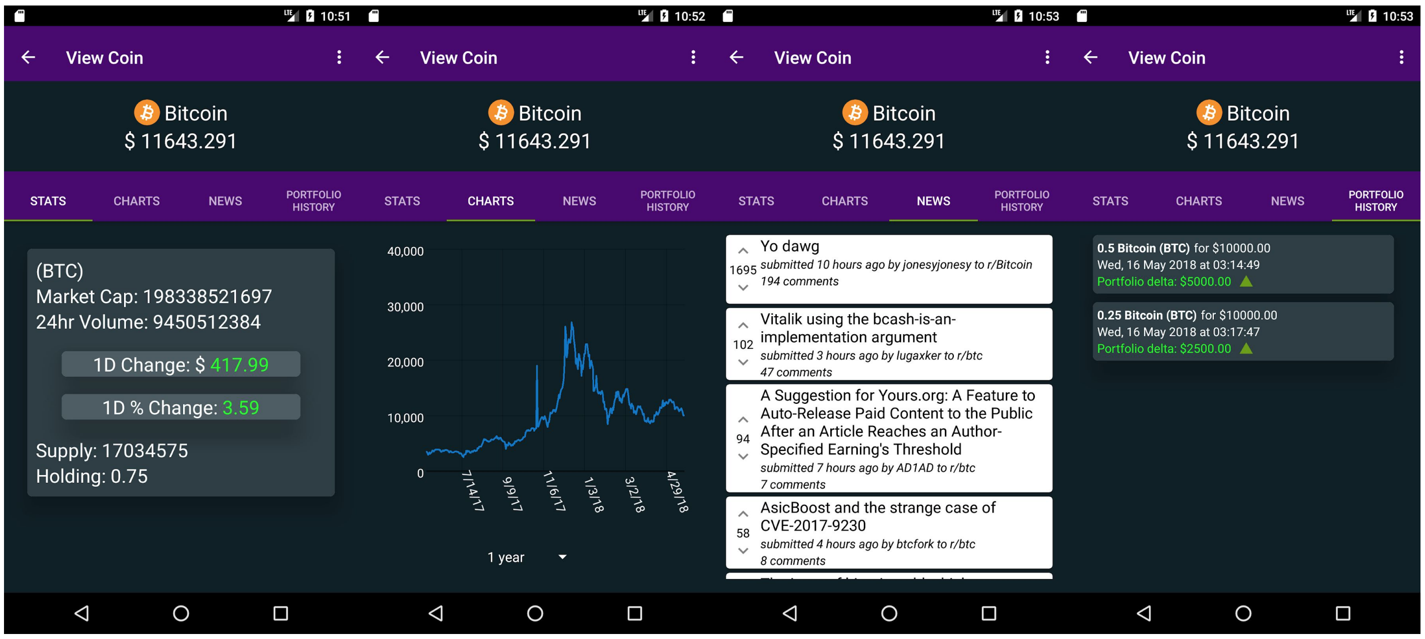
Bytom ▲3.72 1D % Change
Holding: 0.0 BTM: $0.0
Price: $0.92810965

5.3 View Coin

The “view coin” screen is accessed by the user to view useful information specific to a cryptocurrency coin.

It contains tabs to switch between the following screens:

- A stats tab, which shows statistics such as the coin’s market cap, 24hr volume, and 1 day changes. This data is sourced from the CoinMarketCap API.
- A graph of the coin’s value over time, which can be set to view in ranges of 7 days, 1 month, 3 months or 1 year. This data is sourced from the CryptoCompare API.
- A news feed taken from the social media site Reddit for the particular coin’s subreddit (made up of multiple subreddits (“multireddits”) where appropriate). If the coin’s subreddit cannot be found, this page will default to showing posts from /r/CryptoCurrency.
- A list of the user’s transaction histories for this coin, which may be deleted with a long press. If the user does not hold any of this coin, the page will be blank except for a “No transaction history to display” message.



(BTC)
Market Cap: 198338521697
24hr Volume: 9450512384
1D Change: $ 417.99
1D % Change: 3.59
Supply: 17034575
Holding: 0.75

1695 Yo dawg
submitted 10 hours ago by jonesyjonesy to r/Bitcoin
194 comments

102 Vitalik using the bcash-is-an-implementation argument
submitted 3 hours ago by lugaxker to r/btc
47 comments

94 A Suggestion for Yours.org: A Feature to Auto-Release Paid Content to the Public After an Article Reaches an Author-Specified Earning's Threshold
submitted 7 hours ago by AD1AD to r/btc
7 comments

58 AsicBoost and the strange case of CVE-2017-9230
submitted 4 hours ago by btcfork to r/btc
8 comments

0.5 Bitcoin (BTC) for $10000.00
Wed, 16 May 2018 at 03:14:49
Portfolio delta: $5000.00

0.25 Bitcoin (BTC) for $10000.00
Wed, 16 May 2018 at 03:17:47
Portfolio delta: $2500.00

5.4 Hot Coins

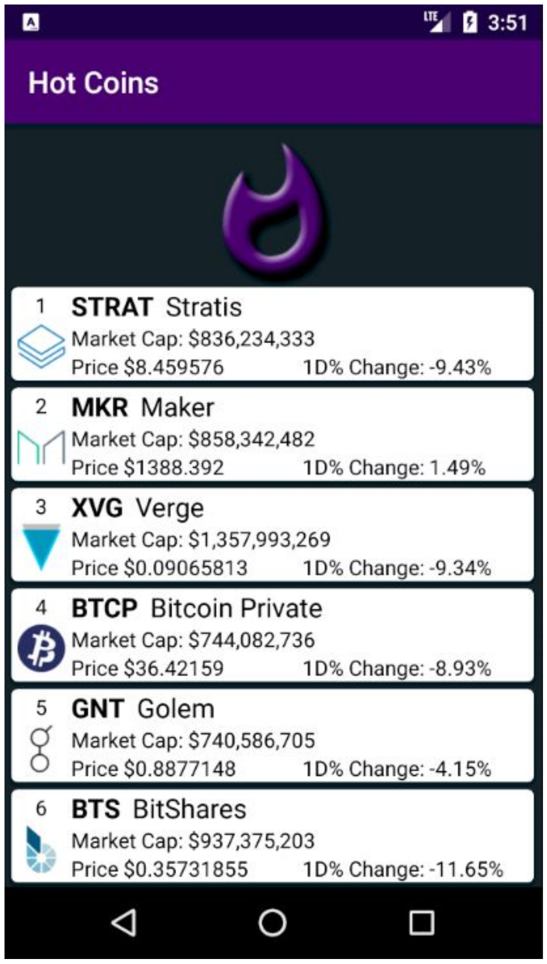
The Hot Coins screen is a composed of a scrollable list of coins that are scraped (when the screen loads) from a website that lists trending cryptocurrencies, based on “real-time scanning of various trading parameters, and news and social media chatter.”[3]

Each coin card in the list displays:

- its rank in the list of coins
- its coin code, name and logo
- its current market cap, and
- its price and daily change (%)

Clicking on any of the coins will take you to that coin’s View Coin page, where you can view more detailed statistics about that coin, some news articles and a list of your transactions involving that coin (as described in section 5.3).

It should be noted that “hot” or “trending” does *not* mean that the coin is having (or has had) an upward trending price - in fact it can often mean the opposite. The Hot Coins page tries to display coins that are being talked about in news and social media.



1 **STRAT** Stratis
Market Cap: $836,234,333
Price $8.459576 1D% Change: -9.43%

2 **MKR** Maker
Market Cap: $858,342,482
Price $1388.392 1D% Change: 1.49%

3 **XVG** Verge
Market Cap: $1,357,993,269
Price $0.09065813 1D% Change: -9.34%

4 **BTCP** Bitcoin Private
Market Cap: $744,082,736
Price $36.42159 1D% Change: -8.93%

5 **GNT** Golem
Market Cap: $740,586,705
Price $0.8877148 1D% Change: -4.15%

6 **BTS** BitShares
Market Cap: $937,375,203
Price $0.35731855 1D% Change: -11.65%

[3] https://stokz.com/cryptocurrencies/trending

6. Altered Implemented Functions

6.1 Settings

The Settings page, as outlined in our proposal, was an optional epic that enclosed a series of functions which users could adjust to modify the UI and some functionality of the application.

The task was initially intended to be partially implemented, as discussed in the proposal - "*this epic will be partially implemented at the end of the project. If time allows the page will be expanded on.*"

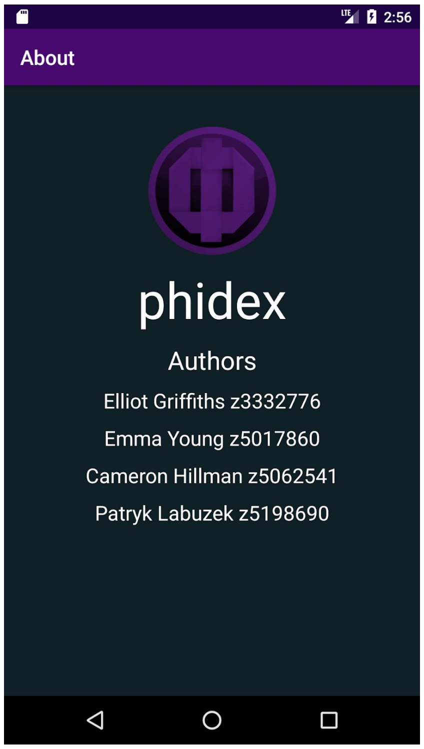
Final project delivery:

Implemented:

- **About page** - Phidex logo and authors

Unimplemented:

- **Day/night mode**
  Dark and light UI themes
- **Other theme options**
  Font, font size, theme colours
- **Export portfolio**
  Exports your current portfolio to a CSV format
- **Changes default trading currency**
  eg. to USD or EUR (locality)



Elliot Griffiths z3332776
Emma Young z5017860
Cameron Hillman z5062541
Patryk Labuzek z5198690

The features which were left out of the implementation were primarily aesthetic. With only the Export portfolio function adding utility to the application.
The removal of these features relates back to our initial vision of a minimalistic design with a high level of usability for this project

"*create a minimalistic, user friendly and effective portfolio application … upholding sound usability and aesthetics." (2.0)*

We came to the conclusion; adding additional bloat to the application was not necessarily inline with our goals and did not represent a valid expenditure of our limited time.

7. Unimplemented Functions

7.1 Coin Exchanges

The Coin Exchanges Tab outlined in the proposal was to be a separate tab of the View Coin screen. It was planned to display a list of exchanges where the selected coin is available for purchase. From this tab the user could also check the accepted currencies which could be used to purchase said coin on a particular exchange.

This epic also included an optional stretch goal. This feature would display the net cost of purchase and transfer of the coin for each of the displayed exchanges. This cost would be displayed in both ETH and AUD and calculated by summing the following:

1. Funds deposit fee
2. Percentage purchase fee
3. Price of currency
4. Withdrawal to personal wallet fee

The decision was made to not implement the Exchanges tab. This was due to several reasons, with the main issue being time limitations and lack of free to use API for gathering live data for fees / costs of all top 100 cryptocurrencies.

8. Implementation Challenges

8.1 Persistent Side Menu

From the design perspective, perhaps one of the biggest challenges we have faced was the overall aspect of the user-flow. During one of our first meetings we have established that our app will place a prominent focus on the user experience which includes the design and user interaction with the app. The challenge involved finding a way to make the user interaction as intuitive and therefore as hassle free for the user as possible. Following Google's material design guidelines we found a method to overcome this problem - a persistent side Menu (picture 1 of section 3). We decided that the home screen of our app - Portfolio Activity should contain this feature. The biggest issue was the implementation of the side menu specifically for that activity and included an overall refactoring of the activity. We had to create separate xml files for each feature and so the activity_portfolio.xml (holds the actual side menu), app_bar_portfolio.xml (toolbar design) and the content_portfolio.xml (overall activity design) were created, which when combined create a single activity with the persistent side menu.

8.2 Room Database

For the most part, implementation of the app's database was challenge free. At the time of development, the Android's Room Persistence Library seemed to be an industry standard for Android apps that needed to store data that persisted between sessions, and the library was exactly what we wanted - a well supported, well documented abstraction of SQLite made for Android apps written in Java.

We had some issues with *using* database early on in the project, while we were still deciding what the database was going to look like (ie. what tables? what fields?), but once we had that sorted, the database was easy to use.

About three quarters into our project we made a major change to the *role* of the database in our app. Previously, we had been using the database to store only coins that we were tracking in our portfolio, and whenever we needed data for pages that listed other coins, we did an API call (eg. for Hot Coins, Search, and related View Coin pages). This meant that we were sometimes making hundreds of calls every minute. We decided instead to take full advantage of our scope (the top 100 coins) and store all the data that our app might need with one API call whenever the app loads (this is described in more detail in 8.3).

8.3 Coin Data API

Using the CoinMarketCap API to source our coin data came with its challenges. We had to carefully consider the way in which we structured the usage of API calls to limit them as much as possible, as although CMC provided a free API, they did limit the number of usages and would block the app from making anymore if too many were made. Initially, we were making calls whenever we wanted to display data, but as this would lead to being rate limited, we eventually designed our app so that only one call would be made whenever the portfolio page of the app was loaded, and then have that data stored in our database, and for all other pages of the app we would source our data from that previously-made API call.

8.4 Dynamic View Generation

An integral element of developing an android application is the creation of xml files with enclosed 'views' (fields) for displaying images, text and calculated values. These views are then filled with data from the java code. Knowing the number of views required on the majority of our applications pages is an intrinsic issue which we had to solve. For example one user may have two coins in his portfolio while another may hold all 100 coins. Generating 100 card views and only filling a couple of them would be a large waste of resources.

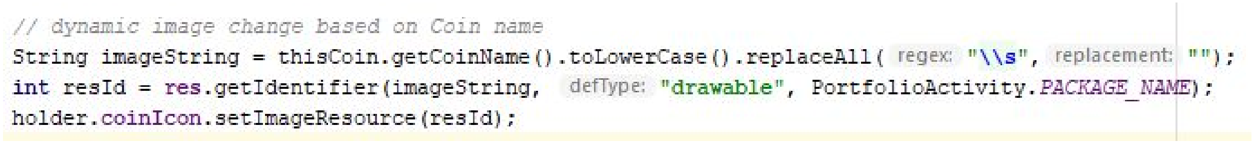
We first attempted to display the data using a 'ListView' which "*displays a vertically-scrollable collection of views, where each view is positioned immediately below the previous view in the list*" [1]

For the most part the ListView would handle this issue. However, as our application used dynamic fields we needed further unitality with the ability to update the view without reloading the entire page. For this we found we could use a RecyclerView.

The RecyclerView *" introduces an additional level of abstraction between the RecyclerView.Adapter and RecyclerView.LayoutManager to be able to detect data set changes in batches during a layout calculation. This saves LayoutManager from tracking adapter changes to calculate animations. It also helps with performance because all view bindings happen at the same time and unnecessary bindings are avoided."* [2]

8.5 Dynamic Logo Presentation

Initially during the early stages of our project manually attaching an image of the coin to its content wasn't an issue, especially when our portfolio could consist of a maximum of 3 coins. However, as explained in the above section 8.4, one user may hold only those 3 coins in the portfolio, while another user may hold all 100 coins. This is problematic as it would require a lot of monotonous work. Another problem with this approach is the lack of flexibility for potential future updates such as extending our database of coins to maybe 500 coins. We decided to dynamically allocate an image with its corresponding coin, however this is easier said than done. It was difficult to find a way in which we can do that, as it wasn't just a simple match of the graphics name with the name of the coin. In Android development, all resources are stored with a unique id. In our problem, we had to deal with two seperate sets of id's, one set for the graphics, the other for the View that encapsulates all the data about a specific coin.
Below is the snapshot of the code that dynamically generates the image to the corresponding coin :



```
// dynamic image change based on Coin name
String imageString = thisCoin.getCoinName().toLowerCase().replaceAll( regex: "\\s", replacement: "");
int resId = res.getIdentifier(imageString,  defType: "drawable", PortfolioActivity.PACKAGE_NAME);
holder.coinIcon.setImageResource(resId);
```

The above is the final version of the code that deals with another issue we encountered towards the end of our project. As some coins in our database contain spaces such as Bitcoin Gold, or Ethereum Classic, converting these names into small letters and linking them with the id of the graphic (named after each coin using small letters and no spaces) didn't work for obvious reasons. To solve this problem we had to do some text editing, mainly replacing the space in a string with "", which combines the two parts into a single string.

The implementation of dynamic logo presentation has many benefits but the biggest one is the possibility for the apps future growth, without having to add or change any lines of code. If in the future the app is to be extended to hold more than 100 coins, the logo of each new coin must simply be added to the Drawable folder of the project and the coin has to be added to the database. The rest is achieved dynamically without the need of developers input.

8.6 Filtering Search

Filtering the RecyclerView on the search page proved to be quite difficult. First we needed to create a method for sorting a list of our Coin objects which were pulled from the room database. This method needed to be able to compare, insert and remove items and while doing so, update the coin adapter. We also needed a method for adding and removing coins from the adaptor. A Comparator method would compare the string entered into the search box with the names of the coins in the sorted list. Only items that matched would then be added to the coin list to display and the adaptor was notified of the change. For this to be dynamic a OnQueryTextListener was used to continuously apply the filter method whenever text in the search box was changed.
All of the above methods would allow the RecyclerView to dynamicly count the number of items in the filtered and sorted list and then display them correctly.

8.7 Charts

There were some challenges that came with building out the charts tab, as the use of third party libraries was required to get the functionality we needed, as well as the issues faced around async API calls. The two main chart libraries available for free use were MPAndroidChart and GraphView. Initially we chose MPAndroidChart for its better design, however the documentation available was not sufficient for our use case, and we struggled to get the chart displaying as we wanted to. The option we went with, GraphView, also had its own difficulties due to an existing bug in the library that had not been fixed by the library's owner. This bug was due to the graph needing to be instantiated as soon as the graph tab was opened, but we would need to wait a second for the API call to be returned with data before we could populate the graph, but the graph would not properly refresh once this data was given. A workaround for this was developed by creating a CustomGraphView class that implemented the original GraphView library class and made a few small changes, and using that instead of the original GraphView class.

8.8 News

The news section of the View Coin page went through many design iterations of problems and solutions before we arrived at our current implementation.

The problem.
How can we display news in a way that could be applied the same way to any coin in our scope (top 100)?

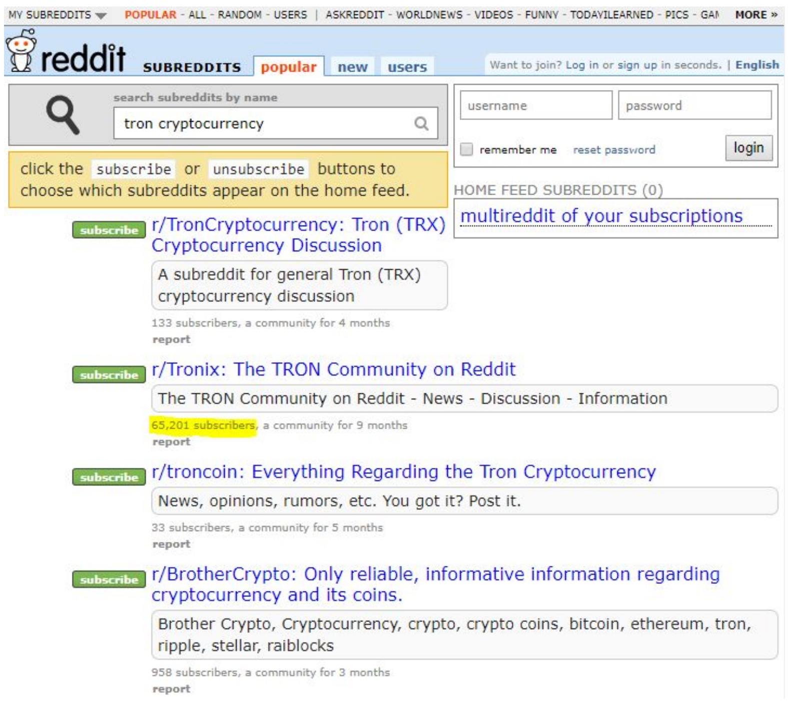
A solution?
Pull news articles from `reddit.com/r/{coin}`.
We decided early on to use Reddit as our primary source of news, because of a couple reasons:

1. Reddit has a json-based API that our app could easily understand.
2. Different subreddits are essentially the same under the hood (ie. the same methods applied to one subreddit could be applied to the rest).
3. A vast majority of the top 100 cryptocurrencies have popular subreddits that are updated regularly.

The problem.
Many subreddits at `/r/{coin}` weren't actually the subreddits dedicated to that coin.
For example, Bitcoin's subreddit is indeed `/r/Bitcoin`, but `/r/TRON` is dedicated to the popular science fiction franchise by the same name, rather the coin TRON (whose subreddit is actually `/r/Tronix`).
In addition, "Bitcoin Cash" maps to `/r/Bitcoin Cash`, which is an invalid URL.

A solution?
In order to map from {coin} to {coin's subreddit}, when loading a news page, use Reddit's API to search for "{coin} cryptocurrency" and choose the most popular (or first?) result. This result could be cached for later.

The problem.
Coins like "Tron" might now map correctly "Tronix", but "Bitcoin Cash" now actually maps to "Bitcoin" rather than "BitcoinCash", because Bitcoin's subreddit is more popular.
What's more, coins like Bitcoin actually have multiple active subreddits (`/r/Bitcoin` and `/r/BTC`) - how can we choose?


MY SUBREDDITS
POPULAR - ALL - RANDOM - USERS | ASKREDDIT - WORLDNEWS - VIDEOS - FUNNY - TODAYILEARNED - PICS - GAM
MORE »
reddit
SUBREDDITS
popular
new
users
Want to join? Log in or sign up in seconds. | English
search subreddits by name
username
password
remember me
reset password
login
HOME FEED SUBREDDITS (0)
multireddit of your subscriptions
subscribe
report
tron cryptocurrency

click the `subscribe` or `unsubscribe` buttons to choose which subreddits appear on the home feed.

r/TronCryptocurrency: Tron (TRX) Cryptocurrency Discussion
A subreddit for general Tron (TRX) cryptocurrency discussion
133 subscribers, a community for 4 months

r/Tronix: The TRON Community on Reddit
The TRON Community on Reddit - News - Discussion - Information
65,201 subscribers, a community for 9 months

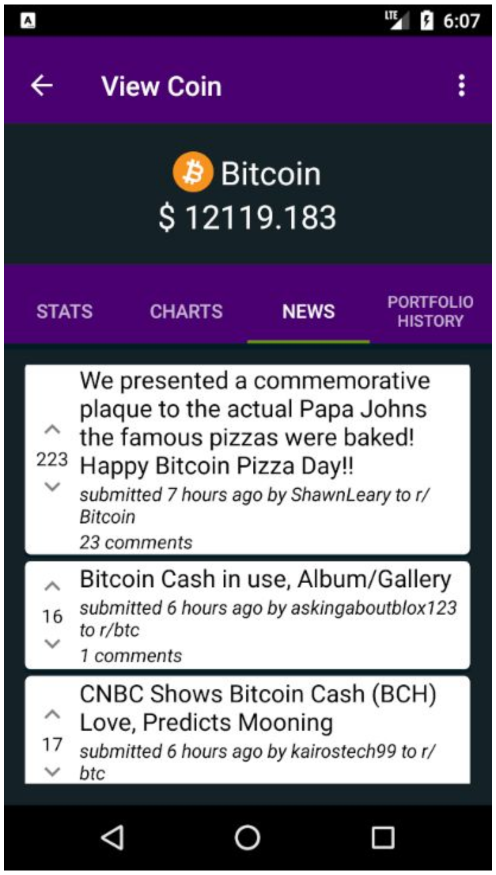
r/troncoin: Everything Regarding the Tron Cryptocurrency
News, opinions, rumors, etc. You got it? Post it.
33 subscribers, a community for 5 months

r/BrotherCrypto: Only reliable, informative information regarding cryptocurrency and its coins.
Brother Crypto, Cryptocurrency, crypto, crypto coins, bitcoin, ethereum, tron, ripple, stellar, raiblocks
958 subscribers, a community for 3 months

The final solution.
Considering that a coin's subreddits seldom change, it was actually practical to consider our scope of top 100 coins and statically store a coin's subreddit in the application itself. This means that:

1. The application doesn't need to do an expensive API call before doing the actual news API call to determine which subreddit to query. (Which means one less asynchronous network thread, less bandwidth usage, less waiting.)
2. The application doesn't have to interpret new data every time it needs to find out which subreddit to query.
3. Coins with multiple active subreddits (like Bitcoin) can be manually considered and stored as "bitcoin+btc" to actually call from *both* subreddits at the same time.
4. The application can have a stored default subreddit for coins that do not have a subreddit, instead of choosing the first result of a list of subreddits related only nominally.

This solution's primary disadvantage is that subreddits for new coins are not dynamically decided, and will need to be manually updated. However, we decided that this solution is well within our scope, and even increased our list of stored subreddits to 150 to account for any coins that move around the 100 boundary.



223 We presented a commemorative plaque to the actual Papa Johns the famous pizzas were baked! Happy Bitcoin Pizza Day!!
submitted 7 hours ago by ShawnLeary to r/Bitcoin
23 comments

16 Bitcoin Cash in use, Album/Gallery
submitted 6 hours ago by askingaboutblox123 to r/btc
1 comments

17 CNBC Shows Bitcoin Cash (BCH) Love, Predicts Mooning
submitted 6 hours ago by kairostech99 to r/btc

The Bitcoin news page pulls from both

`/r/Bitcoin` ***and*** *`/r/btc`*.

8.9 Hot Coins

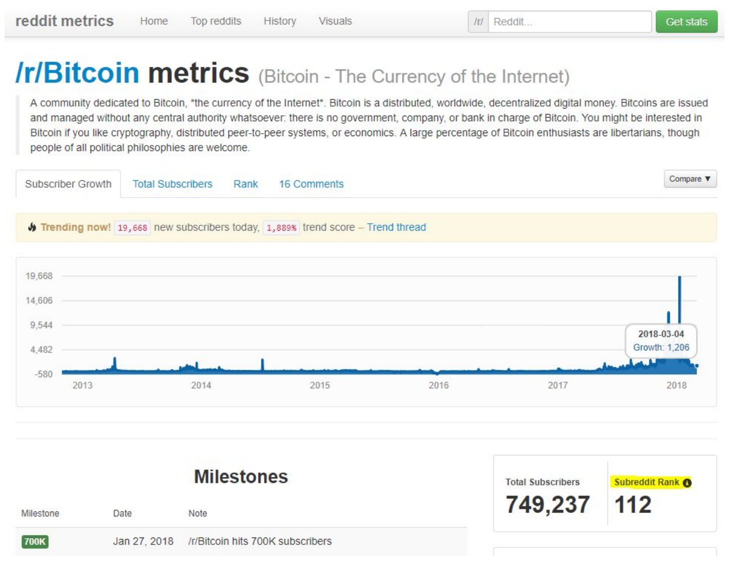
The Hot Coins page was a tricky page to approach. We had an idea of what we wanted the page to look like, but similarly to the news page, we ran into the problem of “where can we get the data we need?” We had a series of ideas that each had inherent problems before arriving our final solution.

Idea 1: choose the coins that had the biggest increase in hourly/daily/weekly price change.
We decided that inevitably, this measured coins that had *already* been talked about and missed the window of “here’s coins that you might want to invest in” - their price had already increased, so it was perhaps too late.

Idea 2: choose the coins that had had the most upvotes/posts on their subreddit in the last hour/day/week.
This approach didn’t work simply because Bitcoin was *always* the coin that had the most upvotes/posts due to the sheer size of the market. We wanted to target coins that had an *increase* in rate of upvotes/posts.

Idea 2b: choose the coins that had the most *increase (proportionally)* in upvotes/posts on their subreddit in the last hour/day/week.
This was an interesting solution to our problem: rather than coins like Bitcoin dominating the Hot Coins page, we could measure the total upvotes/posts in the last week, divide by 7 and compare with the total in the last day to find which subreddits had had the biggest spike in daily traffic. Unfortunately, this solution was too expensive to implement - we would need to make 200 API calls (and interpret/calculate based on data) every time the page was loaded.
In addition to this, it was too unreliable - some coins would shoot to the top of the list with one extra upvote simply because their subreddit isn’t very active, and some didn’t even have one at all.

Idea 2c: use 3rd party Reddit metrics to determine which subreddit had been most active.
Sites like redditmetrics[4] attempt to rank subreddits according to activity, but they have not currently implemented an API to access their data, so our application would have to query their site 100+ times and scrape our data from it. This ended up being, once again, too expensive to perform.



/r/Bitcoin metrics (Bitcoin - The Currency of the Internet)

A community dedicated to Bitcoin, *the currency of the Internet*. Bitcoin is a distributed, worldwide, decentralized digital money. Bitcoins are issued and managed without any central authority whatsoever: there is no government, company, or bank in charge of Bitcoin. You might be interested in Bitcoin if you like cryptography, distributed peer-to-peer systems, or economics. A large percentage of Bitcoin enthusiasts are libertarians, though people of all political philosophies are welcome.

Trending now! 19,668 new subscribers today, 1,889% trend score – Trend thread

Milestones

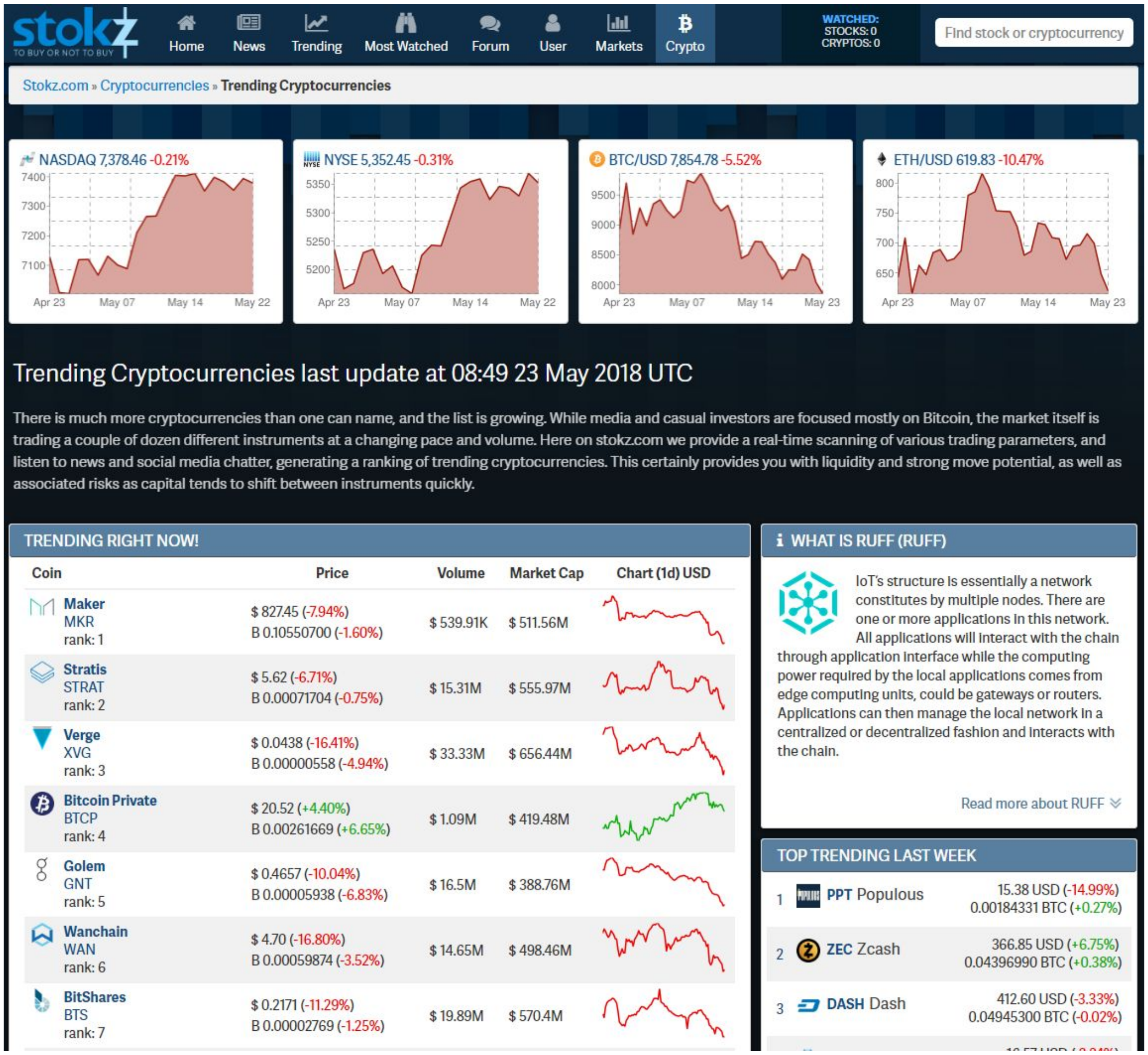
Milestone	Date	Note
700K	Jan 27, 2018	/r/Bitcoin hits 700K subscribers

Total Subscribers	Subreddit Rank
749,237	112

[4] https://redditmetrics.com/

Idea 3: do the news/social media scraping ourselves.
This idea was quickly discarded as being totally out of scope - an entire COMP[39]900 project on its own.

Idea 4: find someone who's already done idea 3 and use their publicly released data.
This solution seemed to be the most viable for our project. The hunt was on to find a website that tries to index coins that are trending in news and social media, and we settled on stokz[5]. Our application performs a GET request and then uses HTML parsing and regular expressions to scrape relevant data from the site in order to provide a list of top 50 trending cryptocurrencies in real time.



Trending Cryptocurrencies last update at 08:49 23 May 2018 UTC

There is much more cryptocurrencies than one can name, and the list is growing. While media and casual investors are focused mostly on Bitcoin, the market itself is trading a couple of dozen different instruments at a changing pace and volume. Here on stokz.com we provide a real-time scanning of various trading parameters, and listen to news and social media chatter, generating a ranking of trending cryptocurrencies. This certainly provides you with liquidity and strong move potential, as well as associated risks as capital tends to shift between instruments quickly.

TRENDING RIGHT NOW!

Coin	Price	Volume	Market Cap	Chart (1d) USD
Maker MKR rank: 1	$ 827.45 (-7.94%) B 0.10550700 (-1.60%)	$ 539.91K	$ 511.56M	
Stratis STRAT rank: 2	$ 5.62 (-6.71%) B 0.00071704 (-0.75%)	$ 15.31M	$ 555.97M	
Verge XVG rank: 3	$ 0.0438 (-16.41%) B 0.00000558 (-4.94%)	$ 33.33M	$ 656.44M	
Bitcoin Private BTCP rank: 4	$ 20.52 (+4.40%) B 0.00261669 (+6.65%)	$ 1.09M	$ 419.48M	
Golem GNT rank: 5	$ 0.4657 (-10.04%) B 0.00005938 (-6.83%)	$ 16.5M	$ 388.76M	
Wanchain WAN rank: 6	$ 4.70 (-16.80%) B 0.00059874 (-3.52%)	$ 14.65M	$ 498.46M	
BitShares BTS rank: 7	$ 0.2171 (-11.29%) B 0.00002769 (-1.25%)	$ 19.89M	$ 570.4M	

WHAT IS RUFF (RUFF)

IoT's structure is essentially a network constitutes by multiple nodes. There are one or more applications in this network. All applications will interact with the chain through application interface while the computing power required by the local applications comes from edge computing units, could be gateways or routers. Applications can then manage the local network in a centralized or decentralized fashion and interacts with the chain.

Read more about RUFF

TOP TRENDING LAST WEEK

1	PPT Populous	15.38 USD (-14.99%) 0.00184331 BTC (+0.27%)
2	ZEC Zcash	366.85 USD (+6.75%) 0.04396990 BTC (+0.38%)
3	DASH Dash	412.60 USD (-3.33%) 0.04945300 BTC (-0.02%)

[5] https://stokz.com/cryptocurrencies/trending

9. User Manual

9.1 Setup

There are 2 methods to set up our app on any Android phone. Our recommendation is to follow Method 1 which describes the easiest and fastest way to get Phidex running.

9.1.1 Method 1:

Requirements: Android Phone with access to the internet.

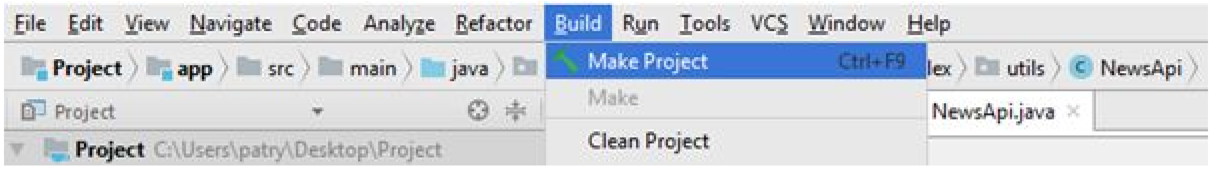
1. On your Android phone, download the compiled and packaged APK from the following link: https://drive.google.com/open?id=1SuaW0BrIOaOsjV1czbjGye4LTMUINHis
2. Open the location of the downloaded APK
3. Click on the ”app-release.apk” icon.
4. A pop up will appear saying the following: “Open with Package Installer”, click on “JUST ONCE”.
5. This will bring you to the installation window. Simply click “INSTALL”.
6. It will take a couple of seconds to install Phidex on your phone. When the “App installed” message shows up, click “OPEN”.
7. The app has been successfully installed and is ready for use.

9.1.2 Method 2:

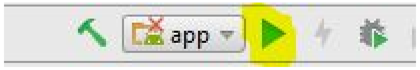
Requirements: Computer with access to the internet and Android Studio.

1. Download the Project file here https://www.dropbox.com/sh/dmcl7mo95ik8ze6/AAA0v7d2ox7_IQyhFeVqDxuLa?dl=0
2. On your computer, download and install the newest version of Android Studio from the following link: https://developer.android.com/studio/?hl=ja
3. Follow the video from the link below, which will guide you through the installation process. Use all the recommended settings. https://developer.android.com/studio/install
4. Set up the Emulator in Android Studio as it is done below: https://developer.android.com/studio/run/emulator
5. Import project into Android Studio by clicking on “Open an existing Android Studio project”.
6. Locate the project file we provided and click OK. This may take few minutes.

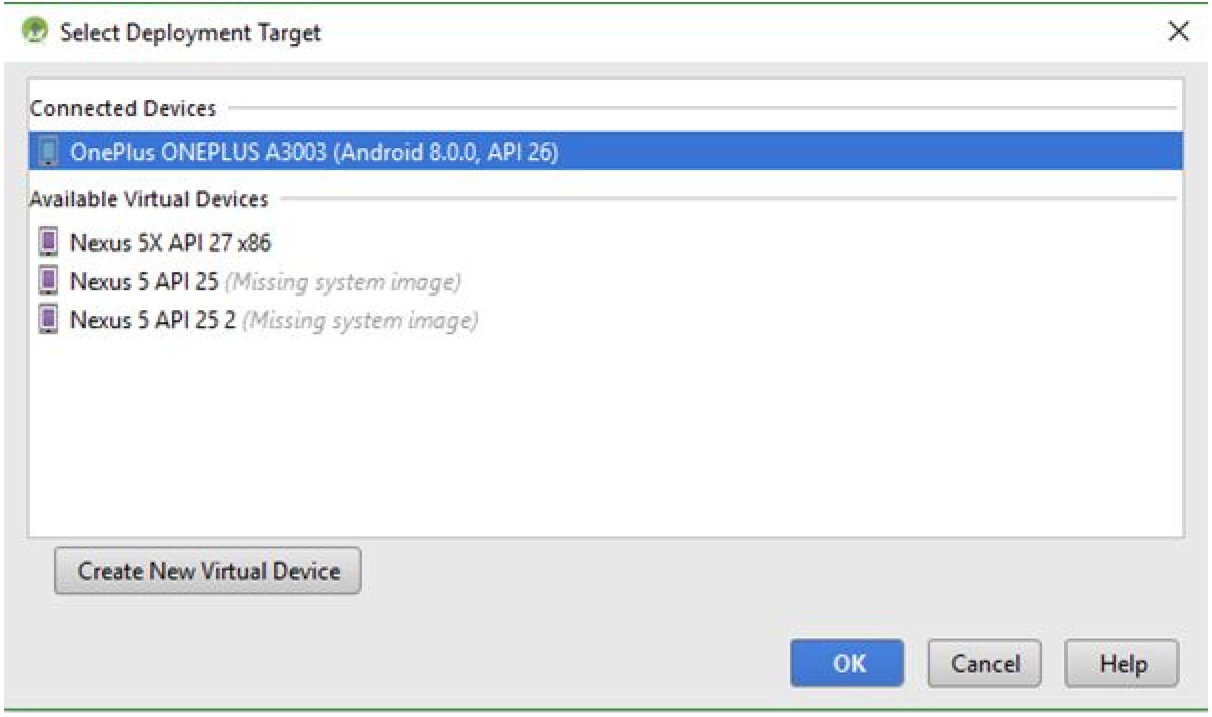
7. Once the project loads and the Gradle finishes building the application is ready to be run.
8. As a precaution, we recommend to go into “Build” in the top menu and click on “Make Project”. The project should build.


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Project app src main java
Make Project Ctrl+F9
Make
Clean Project
lex utils NewsApi
Project
NewsApi.java
Project C:\Users\patry\Desktop\Project

9. Click on the green play icon in the top menu.



10. Select your emulator. Alternatively, you can connect your Android device to the computer and select it. Note that the phone must be in Developer mode. Click OK.


Select Deployment Target
Connected Devices
OnePlus ONEPLUS A3003 (Android 8.0.0, API 26)
Available Virtual Devices
Nexus 5X API 27 x86
Nexus 5 API 25 (Missing system image)
Nexus 5 API 25 2 (Missing system image)
Create New Virtual Device
OK
Cancel
Help

11. The project will compile and install the APK on your device or emulator. This will take few minutes.
12. Phidex is ready for use.

10. Reference List

[1] Android Developers. 2018. *ListView | Android Developers*. [ONLINE] Available at: https://developer.android.com/reference/android/widget/ListView

[2] Android Developers. 2018. *RecyclerView | Android Developers*. [ONLINE] Available at: https://developer.android.com/reference/android/support/v7/widget/RecyclerView

Stokz - to buy or not to buy. 2018. *Trending Cryptocurrencies*. [ONLINE] Available at: https://stokz.com/cryptocurrencies/trending

Reddit Metrics. 2018. *Reddits trending*. [ONLINE] Available at: https://redditmetrics.com/