

Take Home Coding Challenge

The coding challenge consist of building a simple two page/screen app used to explore prices of various cryptocurrencies available on the CoinJar platform.

Essentials

- Use either React or VueJS
- If using React you are welcome to use React Native rather than Web
- You will be relying on two different backend APIs
- The delivered app should be as close as possible to the designs
- Simple and easy to follow architecture
- Clean and readable source code
- Feel to use any npm package or framework
- Icons are provided as pngs in **Assets directory**, @2x version provided if using React Native
- Font are provided **Fonts directory**
- Rounding rule for **all prices**: Round to 2 decimal places for values between 100 and 999.99 and 4 significant figures for all others. For example:
 - 123.3333 rounds 123.33
 - 200.0012 rounds to 200.00
 - 8887.61 rounds to 8888
 - 0.211100 rounds to 0.2111
 - 0.433288 rounds to 0.4333
- Unit tests for tricky non UI logic

Nice to Haves

- For the Home Page list make it responsive for smaller screens (see design for compact size) think iPhone SE (width of 320px)

Home Page

- See home-screen.png
- Display all of the crypto markets for australian dollar in a list.
- Header is fixed
- Take note of the rounding of prices

- Clicking on a particular cryptocurrency will navigate to the trade history page for the selected cryptocurrency

API

- https://api.coinjar.com/v3/exchange_rates
- Filter for counter_currency 'AUD'
- Filter for base_currency: 'BTC', 'LTC', 'ETH', 'XRP', 'ZEC'
- Use 'midpoint' as the price to be displayed

Payload

```
{
  "exchange_rates": {
    "AUDUSD": {
      "base_currency": "AUD",
      "counter_currency": "USD",
      "bid": "0.69411",
      "midpoint": "0.69759",
      "ask": "0.70108",
      "roundtrip_spread": "0.9942"
    },
    "BTCAUD": {
      "base_currency": "BTC",
      "counter_currency": "AUD",
      "bid": "8608.67627527",
      "midpoint": "8679.4502",
      "ask": "8727.42",
      "roundtrip_spread": "1.3606"
    }
  }
}
```

Trade History Page

- See trade-history-screen.png
- List trades in descending timestamp order
- Ask for 20 trades so list of trades is a scrollable
- Fields to display: 'timestamp', 'price' and 'size'
- If 'taker_side' is 'buy' then the price is green and the arrow is up

- If 'taker_side' is 'sell' then the price is red and the arrow is down
- Format timestamp to the nearest second
- No need to round size
- Header is fixed and shows the current pair name in the title

API

- For example of BTCAUD <https://data.exchange.coinjar.com/products/BTCAUD/trades>
- Documentation:
<https://docs.exchange.coinjar.com/data-api/#/reference/products/product/trades>

Submission

Create a git bundle with following command in the root of your working directory:

```
git bundle create coinjar.bundle --all
```

Then send the coinjar.bundle file to michael@coinjar.com