Landchecker Coding Challenge

This code test is your opportunity to demonstrate to us how you think in a relaxed and familiar environment. The test is derived from a real world scenario that we have in LC.

We will be looking to understand how do you architect or structure your code? format code? how you use Git? CSS? how to make use of your limited time? etc.

Please ensure to commit both repos api (rails) + client (reactjs) in github or bitbucket so we can see the commit history.

Time Limit

We expect you to spend between 2-6 hours to complete this challenge.

Delivery

Email us a link to the repos.

Challenge

Display properties as markers on a map using `rails` api and `reactjs`.

1. Import the CSV files into postgres

2. Create a `rails` application to create the `models` and expose a JSON api endpoint for an array of properties. At the very least it should expose the following keys below. i.e Lat/Long so we can point it on the map, etc.

```
{
  "id": 1525674,
  "lga_code": 311,
  "council_property_number": 2188100300,
  "full_address": "8 CLOVERLEIGH AVENUE EMERALD 3782",
  "council": "CARDINIA",
  "postcode": 3782,
  "latitude": 145.449895817713,
  "longitude": -37.9373447811622
}
```

3. Create a `reacts` front-end application which consumes the endpoint and display the markers on the map. You can either use mapbox or google maps.

4. When I click on a property marker, display the property details on the right hand side of the page. See http://landchecker.com.au/map/property/50071791

We don't expect to see all the property details section based on the example link. Showing the full_address, council, postcode based on the sample data will suffice for this test. If you want to mock other parts of the property details section, feel free to do so.

5. Filters. Bonus points if you can add the ability to filter by council on the left side of the page.