

# Synaptop Front-end Developer Challenge

Your mission is to create a simple **React.js Web application** to promote Toronto. Take 2-4 hours and include as many of the features listed below in the app as possible. In completing these sections, care should be taken so that your code is easy to follow, commented where necessary, and well structured. Feel free to add anything that you feel will improve the app and user experience.

Package the final app, upload it to GitHub and send a link back.

If you can deploy the code to Heroku or AWS please do so. When you arrive for your interview, we will review the application with you.

Happy coding!

## App Features

### 1. Login

Use Firebase to login the user into the webapp.

### 2. Browse Points of Interest in Toronto

Create a view to browse the points interest.

I. The app must present the Points of Interest with the ability to switch between different views, i.e. tabular, map, etc...

A. The places to include are as follows:

- Aquarium
- Zoo
- ROM
- AGO
- Yorkdale
- Eaton
- City Hall
- Hockey Hall of Fame
- ACC
- CN Tower

B. Images for these places are provided and can be found in the Images folder.

C. Links and place Titles can be found in the places.plist in the project.

### 3. Find a Bixi Bike

Create a view with a map and overlay Bixi bike locations.

I. Have the app retrieve this JSON data from this url:

<http://www.bikesharetoronto.com/stations/json>

II. Display your current location on the map, and add the ability to have your current location updated regularly. Feel free to add any features that you feel would improve the user experience.

### 4. Favorites

Add favoriting of places and create a view for managing favorited items.

- I. Add the ability to Favorite an item in the Browse Places view. Clicking on the item should display a web page.
- II. Incorporate a gallery to view the photos of the particular favorited place.
- III. Include functionality to remove the item from favorites and persist favorites list between app launches. Download more photos as required.

## **5. Recently Browsed**

Create a section that displays places recently browsed in Part 1 and add annotations for these locations to the map in Part 2.

- I. GPS coordinates for the places can be found in the places.plist in the project.
- II. Selecting an item from the history should take you to the Bixi Map and center its position on the map using a reasonable scale.
- III. The history should be persisted between app launches.

## **6. Add a Location Filter to Bixi Map**

Create filters for the Bixi map.

- I. Scale the map view to show the filtered results without being zoomed out too far.

## **7. Create Tests**

Create tests to validate your code.

- I. Create two unit tests of the most error prone part of the application
- II. Create mock redux tests
- III. Create one selenium test