2020

Top Rated Restaurants Finder

CAB432 Assignment 1

<Sharafat Mir/s>
<n10015957/s>
9/8/2020

Contents

Introduction

Mashup Purpose & description

The Top-Rated Restaurant Finder is an application which allows people to search for top rated restaurants in any city of their preference. This application allows people to enter city name of their preference into the search box, then click on the submit button, and then click the city from the drop down locations. A list of top-rated restaurants with it's information are returned. The user can click on any restaurant to view it's location on the map and view further information about the restaurant from google apis. The purpose of this application is to allow people eat at better restaurants. This application uses multiple sources to get the information to it's users. The ratings and other information about the restaurants are correct and accurate.

Services used

A number of APIs and data sources are used to get information and display it to the users. All of the APIs that are used are mentioned below and what service or value they provide to the users.

Zomato API

Returns a list of top-rated restaurants in any given city. Additionally, it also return some information about the restaurant such as a picture, address, price, and ratings etc. Moreover, firstly the location is searched in this api and once the location is found to be existing in the API it will give you an entity id which is then passed to another Zomato endpoint which then fetches the top rated restaurants in the city.

Endpoint: `https://developers.zomato.com/api/v2.1/cities`

Endpoint: `https://developer.zomato.com/api/v2.1/search?entity_id`

Docs: https://developers.zomato.com/documentation

Google Places API

I have used this API to search for a place and see it's extensive reviews by a lot of people. The reason this API was chosen to get reviews, and rating from was because google is used by a lot of people. If we just used Zomato API's reviews and ratings that wouldn't have been enough. What this API does is it returns a place id, that place id is then used to fetch the place details form another endpoint in this api.

Endpoint:
`https://maps/api/place/nearbysearch/json?location={latitude,longitude}&radius`

Endpoint: `https://maps/api/place/details/json?place_id`

Docs: https://developers.google.com/maps/documentation

Google Maps API

I have used this API to find locations of restaurants and then locate them on the google maps. I have used the Zomato API's longitude and latitude to look for a location. Moreover, then locate that restaurant on the map.

Endpoint:
`https://maps/api/place/nearbysearch/json?location={latitude,longitude}&radius`

Mashup Use Cases and Services

Your User stories should go here. The basic structure is provided, and you should fill in the role and then the action and the good result that follows. Underneath the formal statement of the user story, you can then tell us how you have implemented this service – basically you would tell us how you get input from the user and then use this to get information from an API and then use those results in another one. This is at a semi-technical level – introduce it at a high level and then give more detail, but stop well short of code excerpts. You should then use screenshots to illustrate the process

To illustrate some of what we want, I will make up an example that might not make much sense in 2020:

Top Rated Restaurant Search

As a	Human
I want	To find a list of top rated restaurants with really good food
So that	I can have good food and not waste my money on bad food.

Top Rated Restaurants Reviews & Ratings from widely used source

As a	Human
I want	To find the reviews and rating of a restaurant from a very good source that are widely used
So that	I won't use my money on bad rated food.

Top Rated Restaurant on Map Location

As a	Human
I want	To be able to locate the top rated restaurant's location in a map
So that	I can see the location, I can go and find the restaurant and have a lot of nice food.

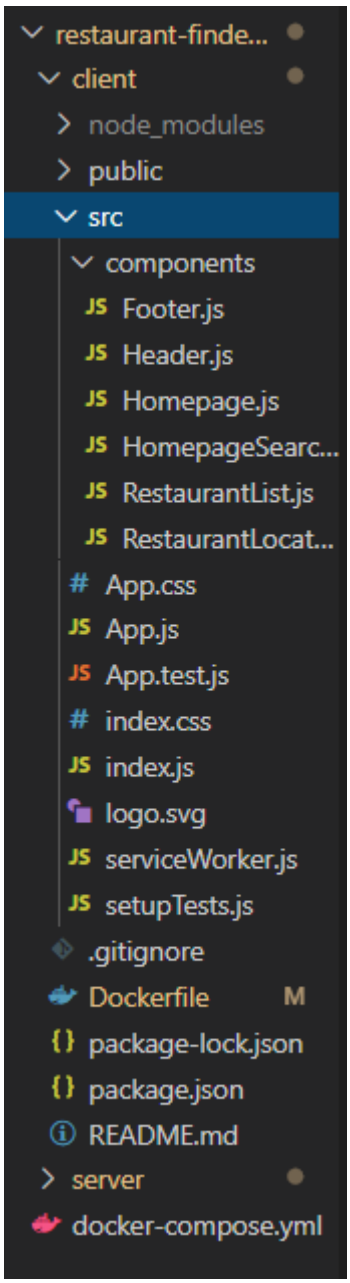
Technical breakdown

This is a deeper discussion of the architecture, the technology used on the mashup, any issues encountered, and overall, how you implemented the project.

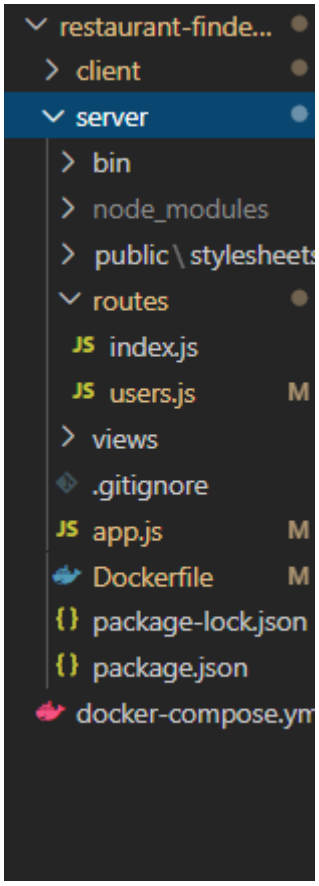
Architecture and Data Flow

My system operates by running two applications which are the client app and the server application. The client app is been developed using react js, material ui for styling, google maps for the mapping, axios, react-dom, react-router were used to as well and other libraries and packages. My client side has package.jsonn file and there is a src directory which contains most of the code for the client side

app. Inside src folder there is app.js which is basically the main function of the application as everything gets called here. Then I have another important folder inside the src folder which is named components. This folder holds the components and the api calls to the server. As well as holds all the styling and other functionality of the client application as can be seen in picture below.



Additionally, for the server application I have used nod and express js. As well as used the express generator to generate the application's architecture is as follows; inside the server directory there is a folder called routes which holds the routes and the api calls to the external sources. Moreover, there is the app.js which is like the main file which calls on all other functions for the operations of the application.



The data flows around the application. So the client app want to make a request to view top rated restaurants in the city of their preference. First the client app make a request to the server application. The server receives that a client wants to view the top rated restaurants in the city. They fetched that information from the external source and displays is to the user. If there are errors in the search or the city doesn't exists then the user will be displayed appropriate error message from the server application.

Deployment and the Use of Docker

I have built back end and front end used react and node, and express js. Therefore, I have a docker file for my client app. Then I have another docker file for my server app. I also have another docker-compose file which is located in the top directory of my project outside of the client and server directories. A copy can be seen in the appendix below.
Docker deployment is a two docker files one in each client and server folder.
Then another docker-compose file in the upper directory
To run the application go to the http://54.79.56.46:3001

Test plan

Manual testing is fine and our expectations are in line with the example grid below. You can show the results through a screen shot and point us to these from the table.

Your tests should include

- *Positive outcome cases*
- *Negative outcome cases (error scenarios)*

- *Edge cases*
- *Non-functional cases (ideally, but not required this time).*

Note that the grid below is unrelated to this application.

Task	*Expected outcome*	*Results*	*Screenshots Appendix*
Search for top rated restaurants in a city	*Top rated restaurants returned*	*PASS*	*01*
Search invalid location	*Throws an error "Please type in correct city name"*	*PASS*	*02*
Scroll down to see all the top rated restaurants	*Show a list of top rated restaurants*	*PASS*	*03*
Click on the restaurant to view it's location on the map and it's reviews	*Shows you the location of the restaurant on the google map and below the map reviews are visible*	*PASS*	*04*
Press go back to search button	*Takes you back to the search results*	*PASS*	*05*
View another restaurant or search in another city to view it's top rated restaurants	*Allow you to look for topo rated restaurants in different cities*	*PASS*	*06*

Difficulties / Exclusions / unresolved & persistent errors /

In this section, you should explain anything that caused you problems and how you overcame those problems. Tell us if there was any issue that prevented you completing the assignment to specification. Tell us about any assumptions or compromises that you have made. Those who worked with an API like Spotify, which presented particular concerns, should discuss the compromises here, and this is also where you can tell us about problems with API keys and responses.

More generally, you might consider:

- *Your major roadblocks and how you resolved them.*
- *Any functionality you didn't or couldn't finish*
- *Are there any differences between your brief and what you delivered? If so, explain why.*
- *Are there any outstanding bugs*

Not required

Extensions (Optional)

In this section, you can tell us if you wish to how you might extend your app and make it better. This is an opportunity to tell us about good ideas that you had that you didn't have time to tell us about.

User guide

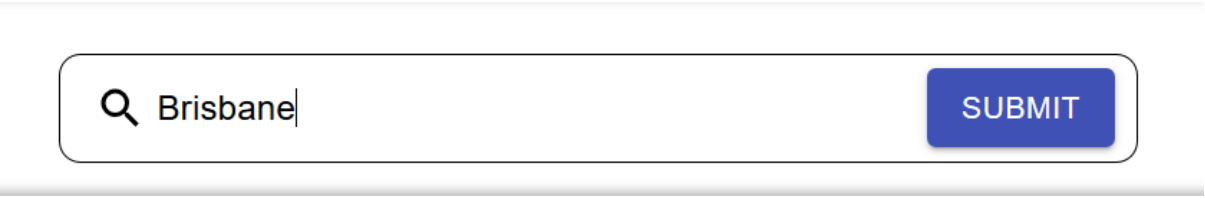
This is the user guide for the top rated restaurant finder application. Which shows you how to use the application.

Step 0: Go to the following link http://54.79.56.46:3001

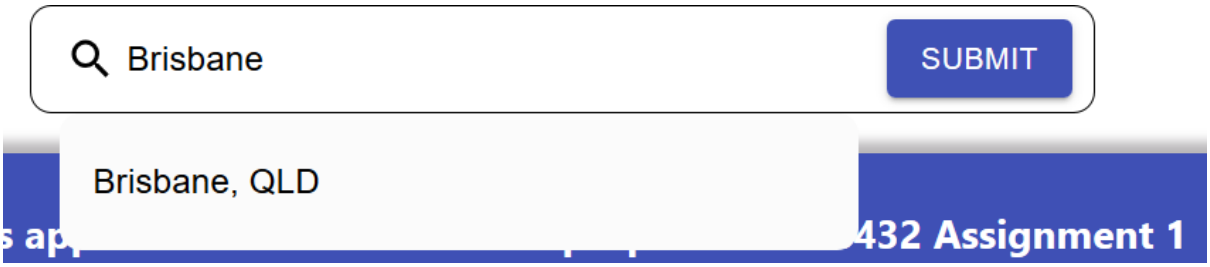
Step 1: Run the application:

Step 2: Go to the http://localhost:3001 port the application is to be deployed on the port and the domain name mentioned above.

Step 3: Start typing the name of the city in search box

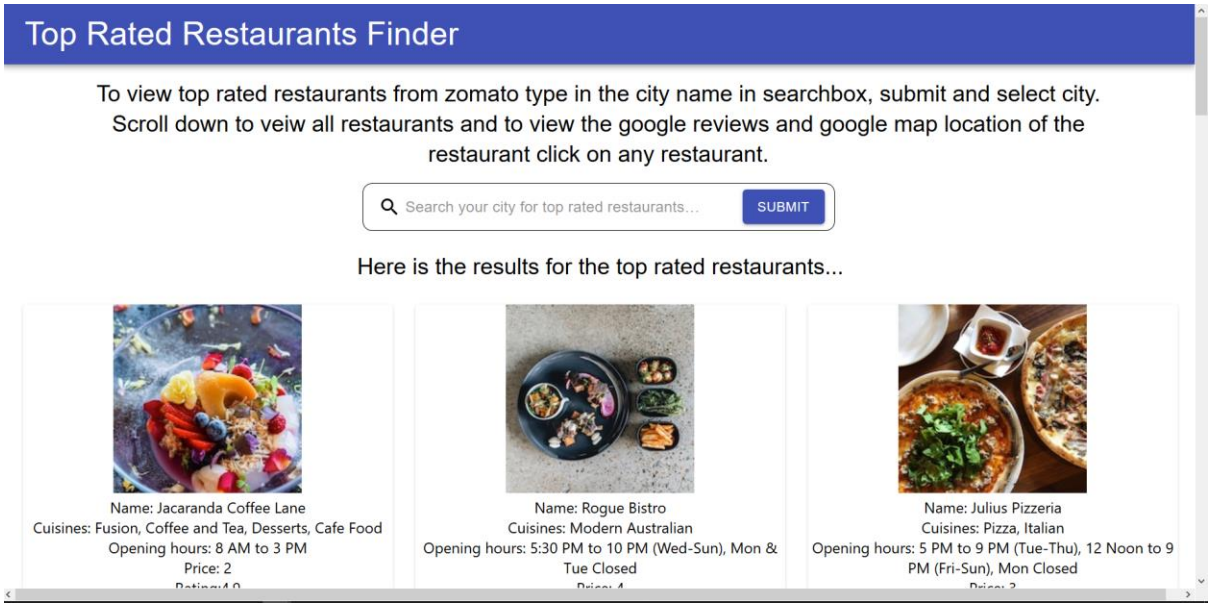


Step 4: Click the submit button and select the city name from the drop down



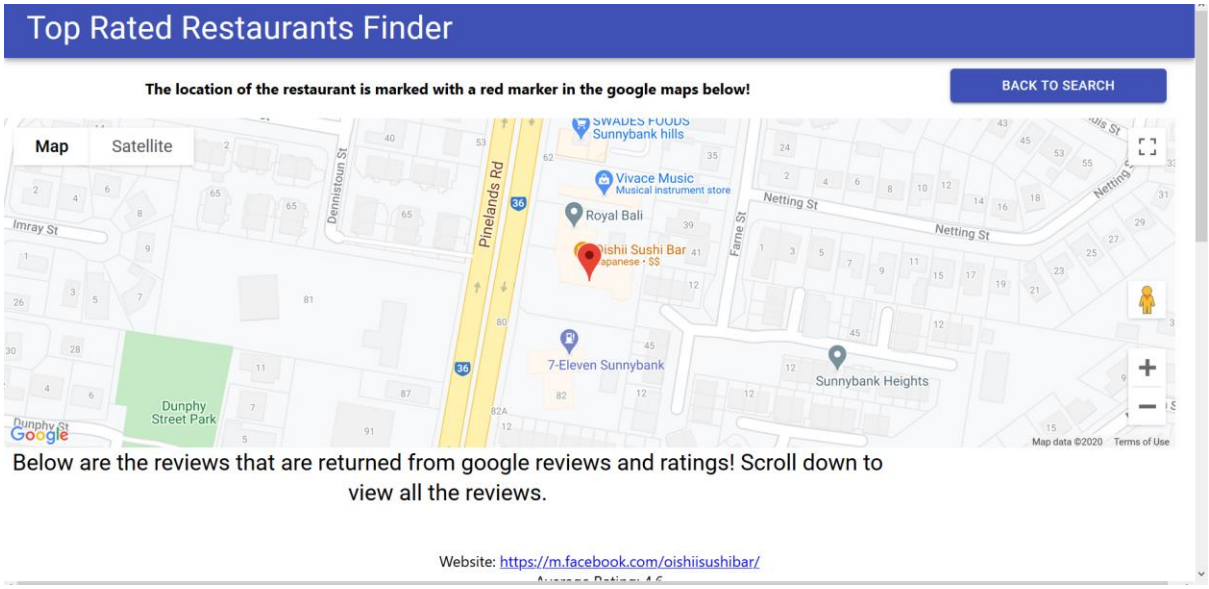
Step 5: A list of restaurants will appear.

Step 6: Scroll down to see all the returned restaurants

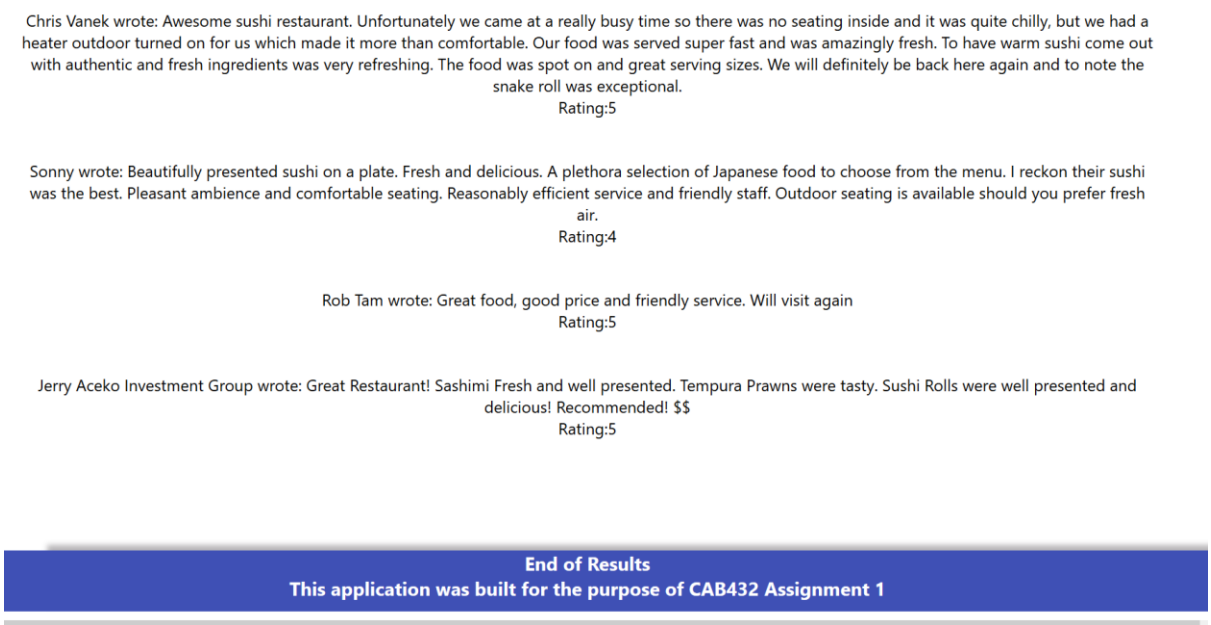


Step 7: Click on any restaurant of your preference to view it's location in the map and further reviews and information about the restaurant.

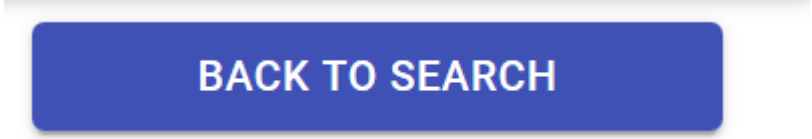
Step 8: Use zoom in and out if you need.



Step 9: Scroll down to see all the reviews and further information about the restaurant.



Step 10: Go back to the results by click the Back to search button.



Statement on Assignment Demo

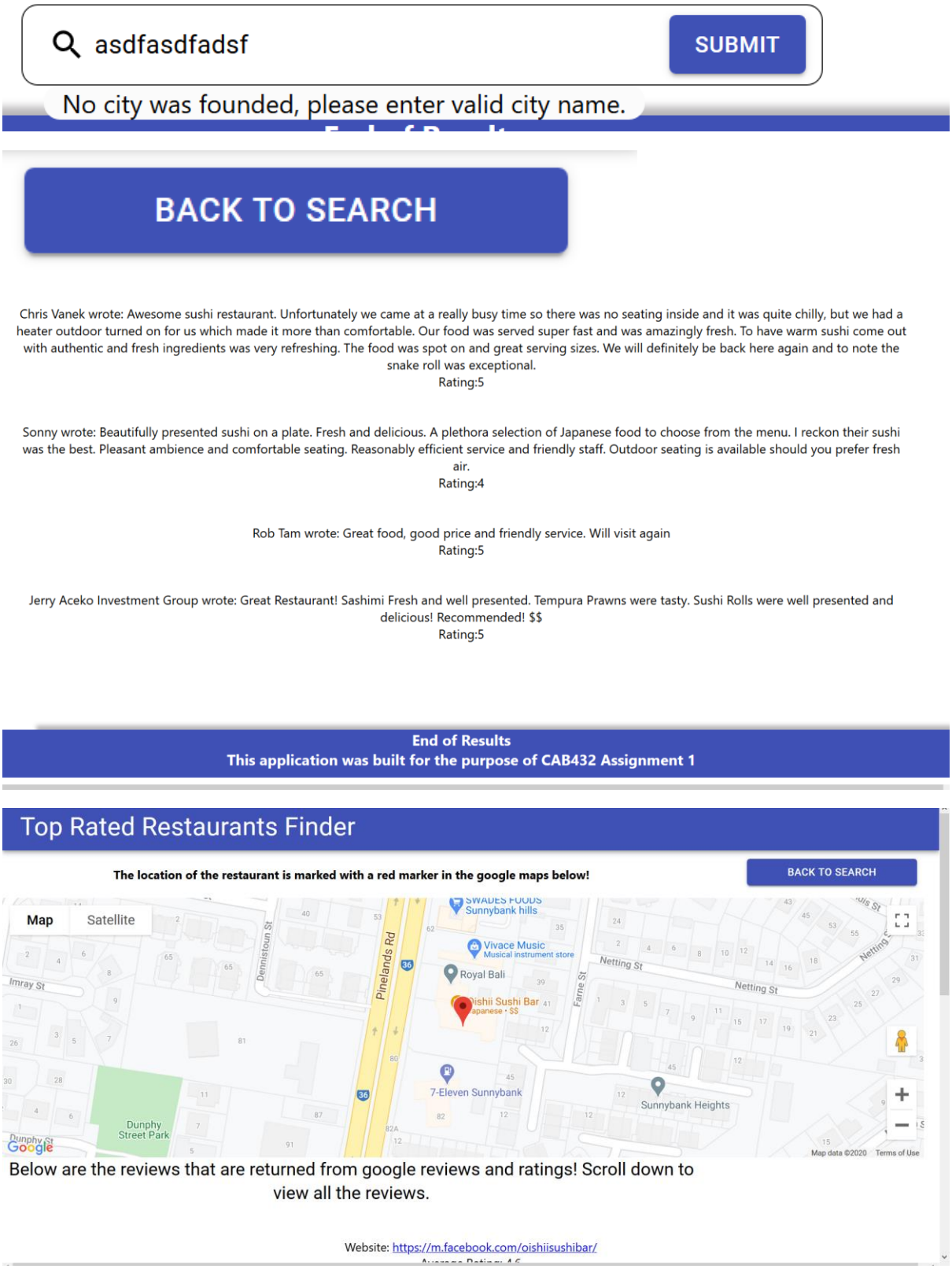
I will do the FACE TO FACE demo.

References

Use a standard approach to referencing – see the guidance at https://www.citewrite.qut.edu.au/cite/.

Appendices

Stuff you want to include, but is too long or too complex to include in the main report text. The full Docker file, some longer excerpt from API docs. Whatever helps.



Top Rated Restaurants Finder

To view top rated restaurants from zomato type in the city name in searchbox, submit and select city. Scroll down to veiw all restaurants and to view the google reviews and google map location of the restaurant click on any restaurant.

Search your city for top rated restaurants... SUBMIT

Here is the results for the top rated restaurants...

Name: Jacaranda Coffee Lane
Cuisines: Fusion, Coffee and Tea, Desserts, Cafe Food
Opening hours: 8 AM to 3 PM
Price: 2

Name: Rogue Bistro
Cuisines: Modern Australian
Opening hours: 5:30 PM to 10 PM (Wed-Sun), Mon & Tue Closed

Name: Julius Pizzeria
Cuisines: Pizza, Italian
Opening hours: 5 PM to 9 PM (Tue-Thu), 12 Noon to 9 PM (Fri-Sun), Mon Closed

Brisbane SUBMIT

Brisbane, QLD

s ap 432 Assignment 1

Brisbane SUBMIT