John Huynh -n9154566

CAB230: Project Web Driven Website

Website works on cHROME (Chrome Testing prefered)

Statement of Contribution

Stage 1:

|  |  |
| --- | --- |
| Name | Contribution |
| John Huynh | Individual Results Page, Registration Page |
| Kathrine Harijatna | Search Page, Search Results Page |

Stage 2:

|  |  |
| --- | --- |
| Name | Contribution |
| John Huynh | All of the backend design, Testing Plan and Report. |
| Kathrine Harijatna | Dropped out of subject (Zero contribution to backend) |

# Test Plan

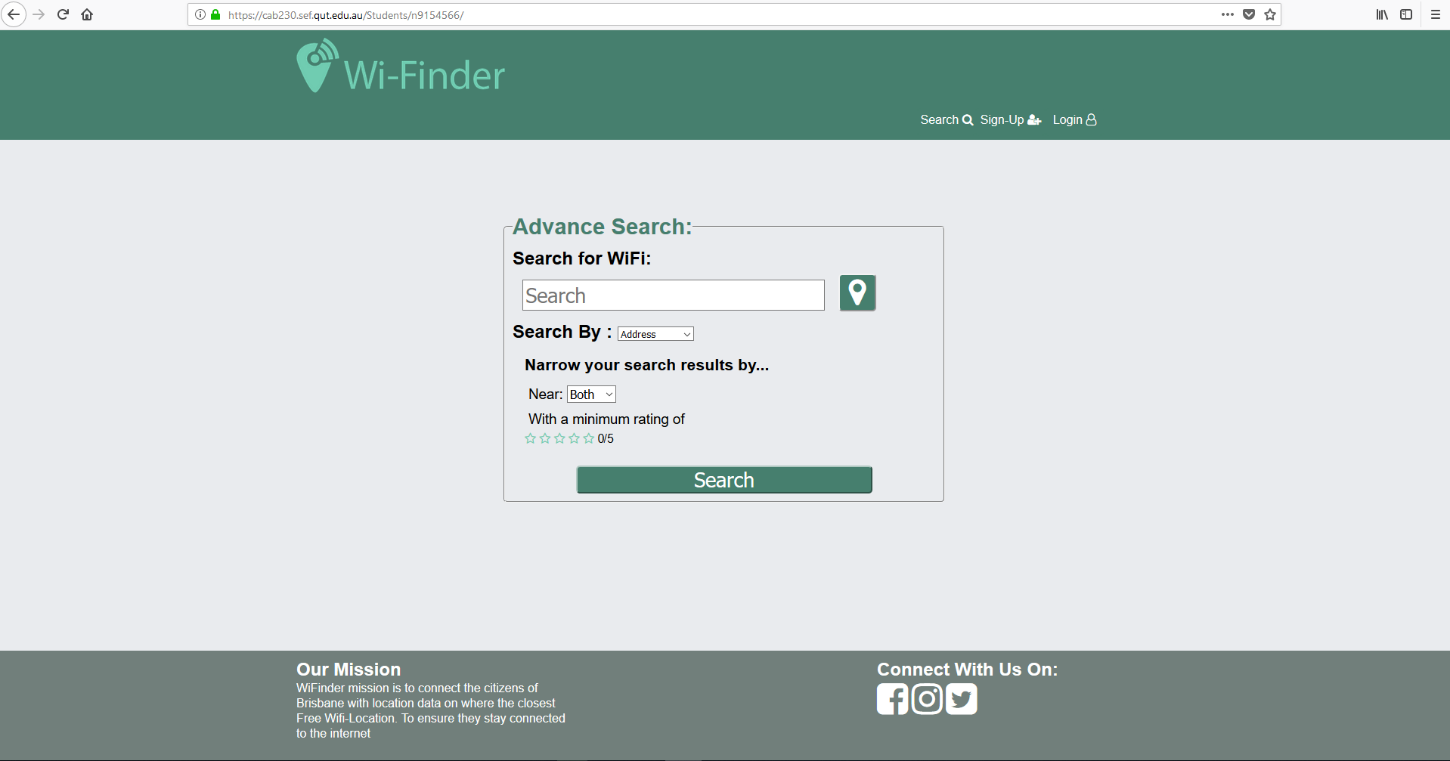
## Home Screen

The Home Screen for our webpage is the Search Screen to access it going to this website:

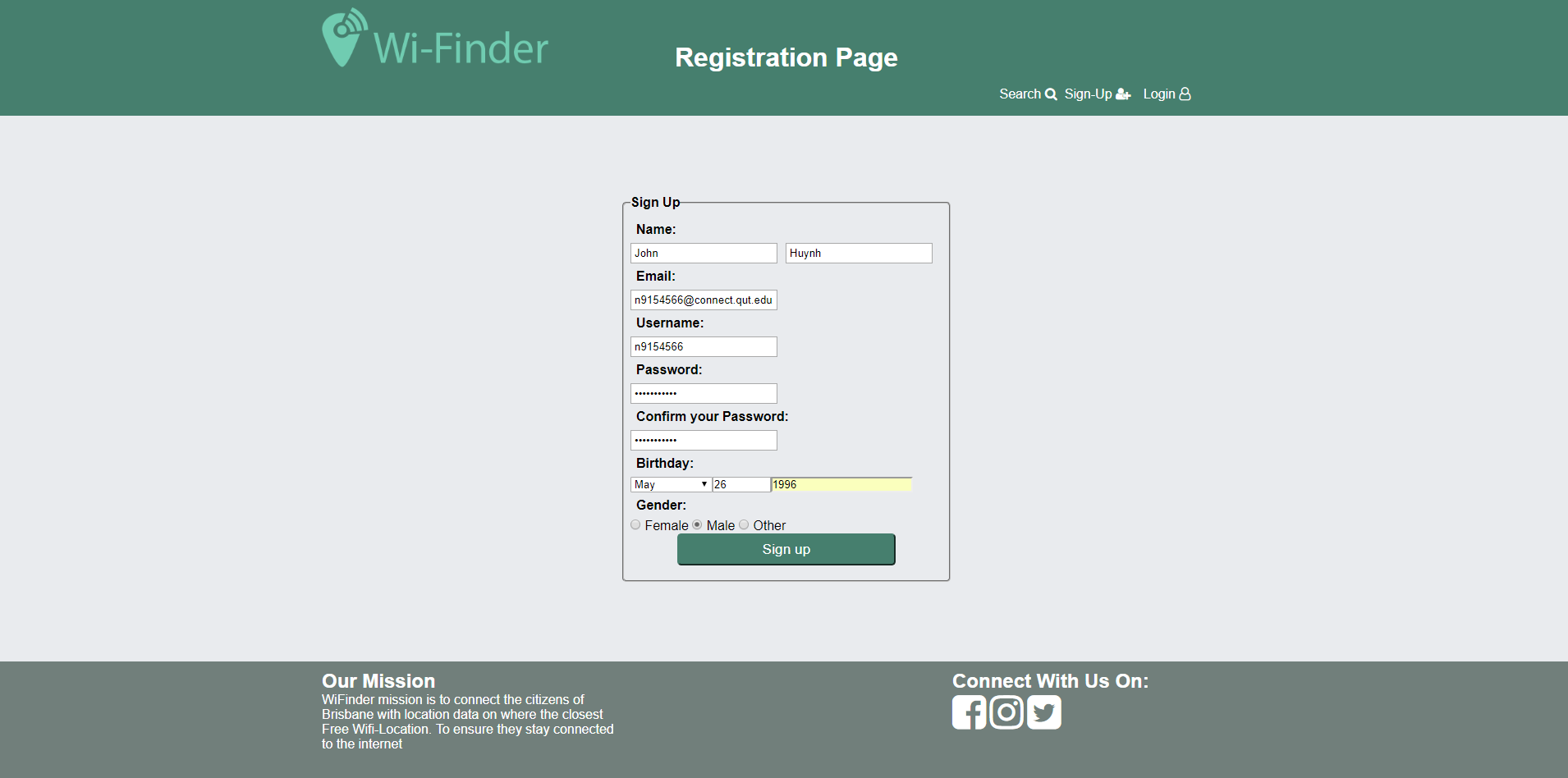
(added a web.config file which sets the default page to search\_page.php)

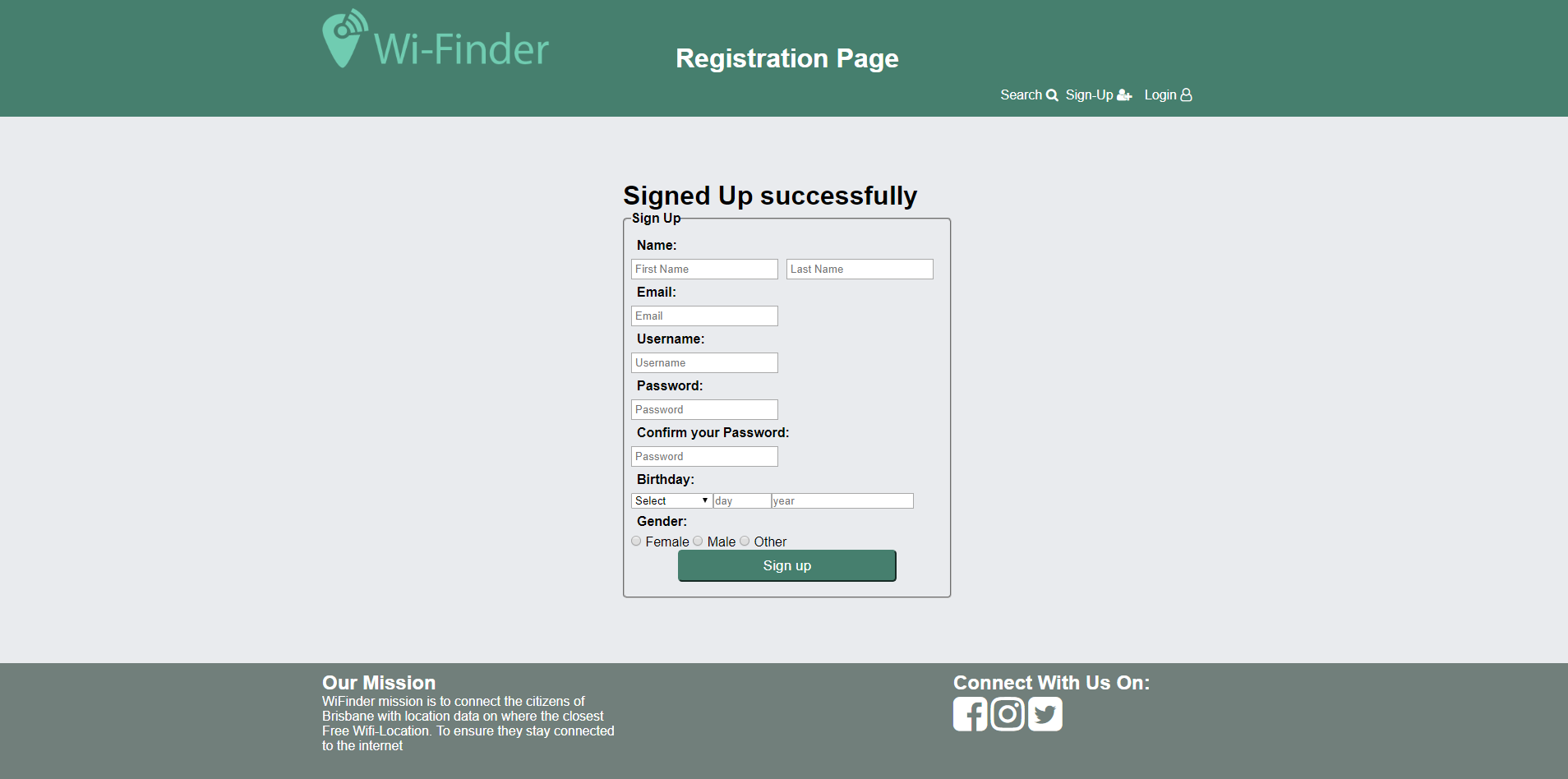
https://cab230.sef.qut.edu.au/Students/n9154566/

if that doesn’t work then use this website:

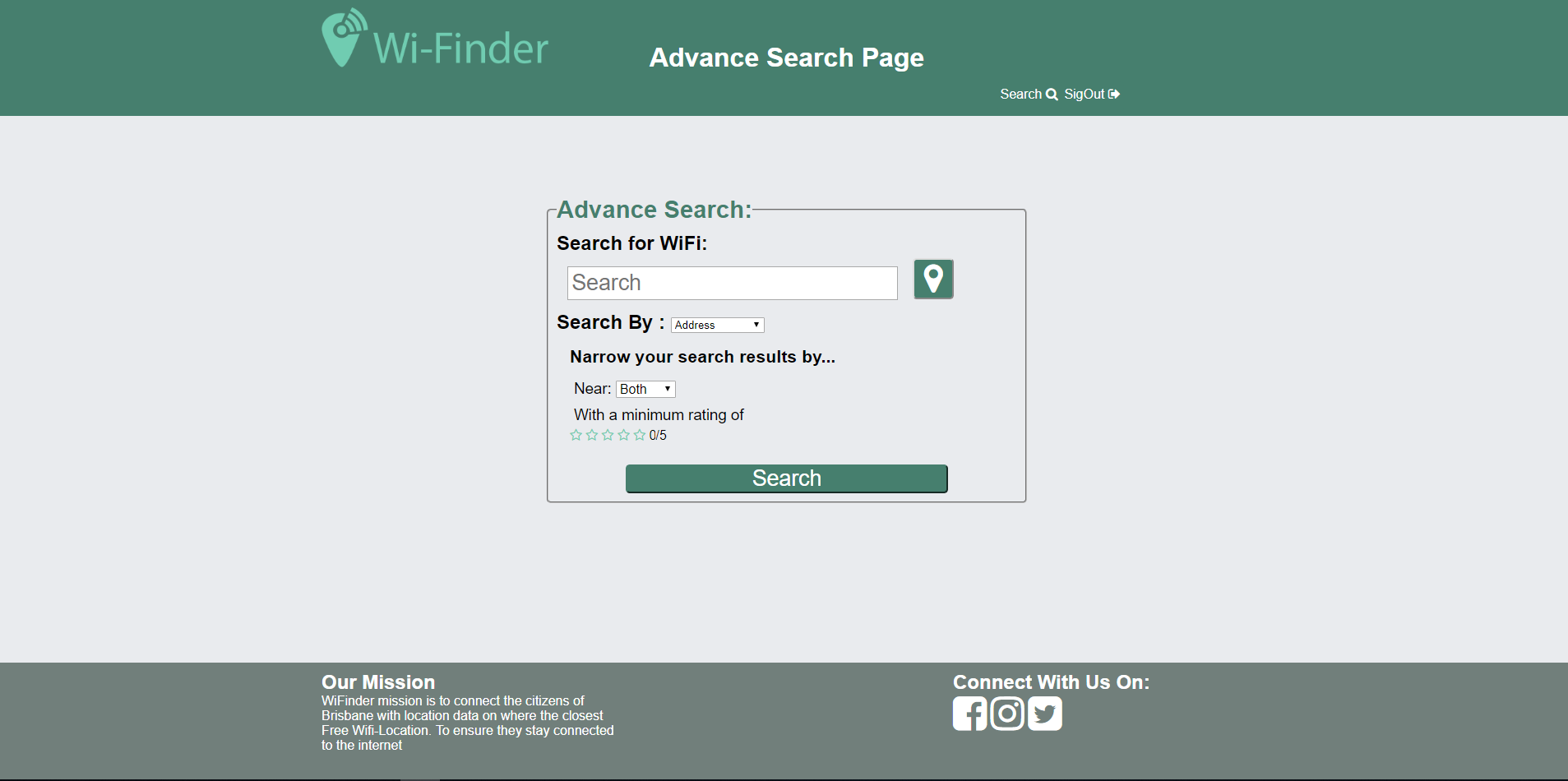
https://cab230.sef.qut.edu.au/Students/n9154566/search\_page.php

## Registering new user

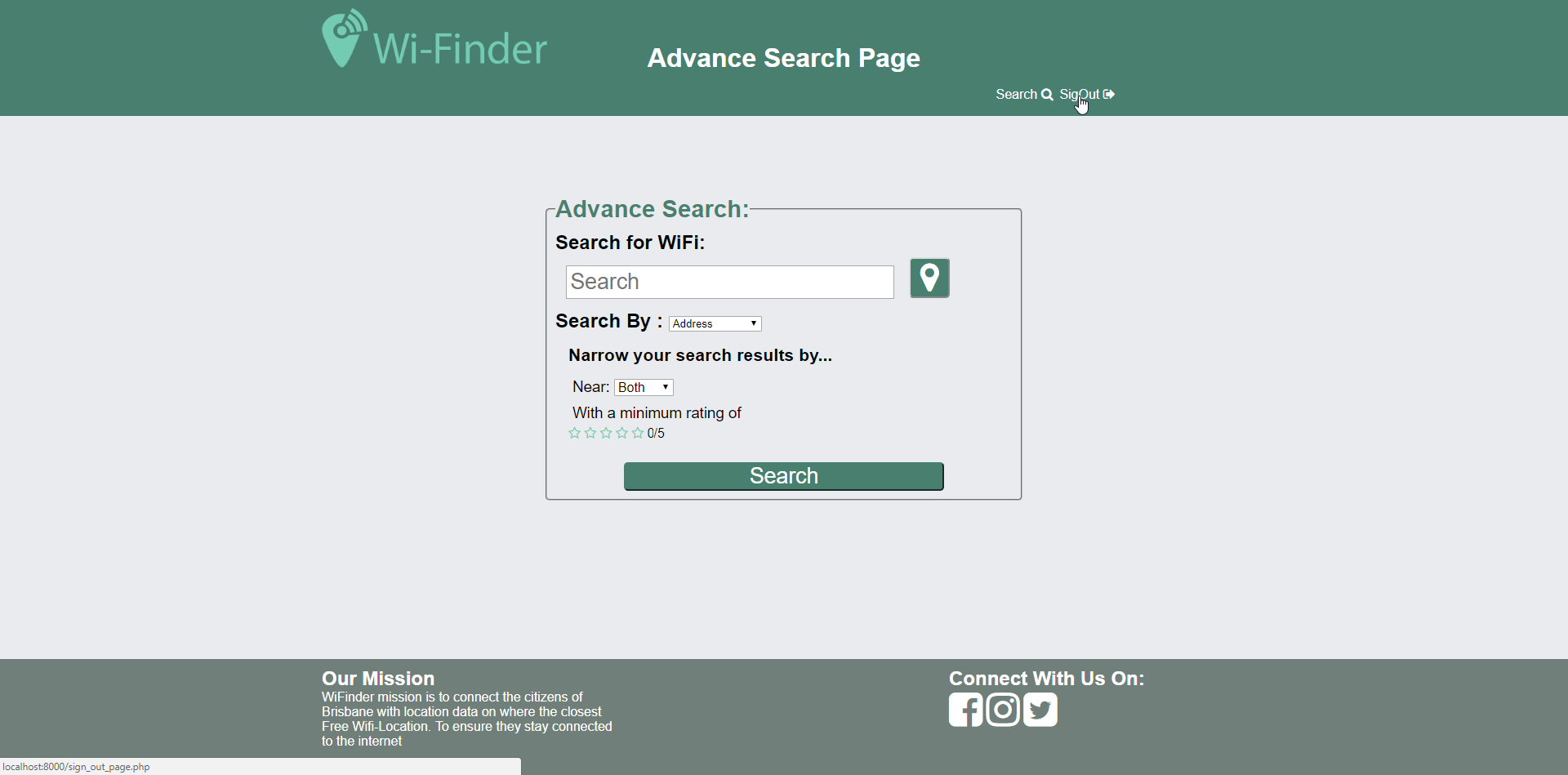


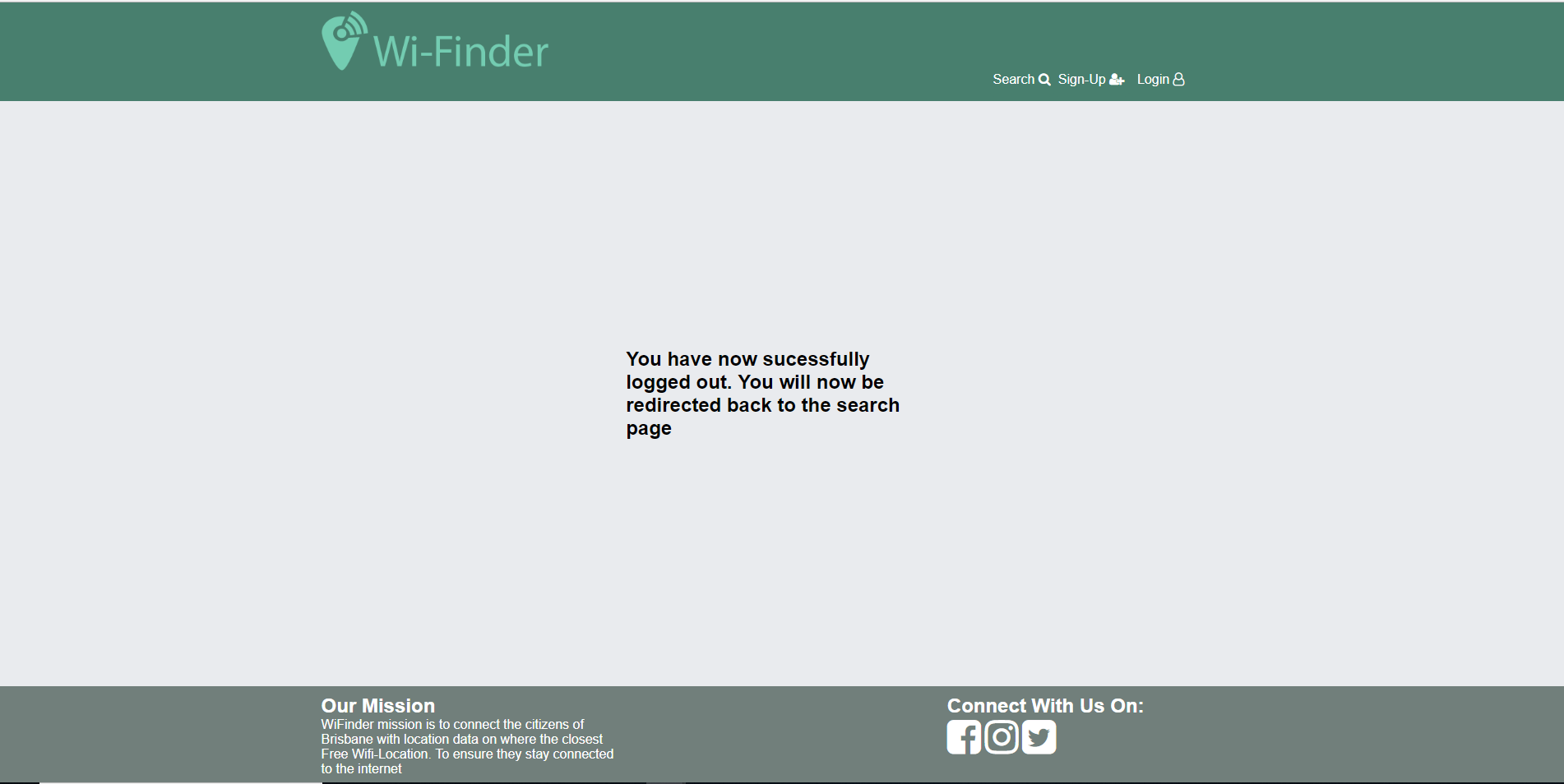


## Login in as existing user

After the user has entered their credentials and it is successfully verified by the server then the user is redirected to the search page and the signup and login button have been changed to a sign out button.  


## Logging out



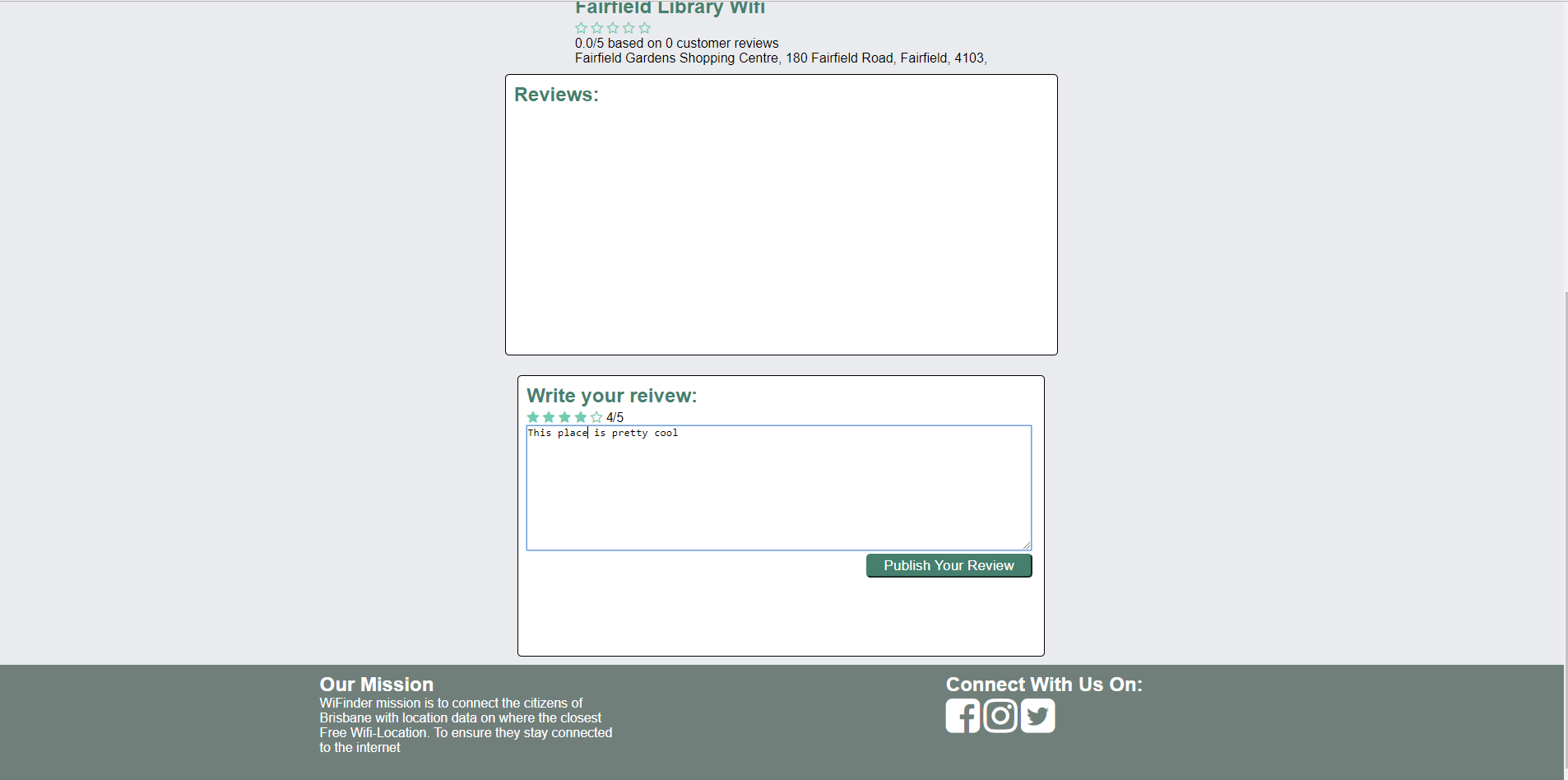


## Add a review

To add a review the user must be logged in if the user is not logged it in will show this screen.



If they are logged in then they can type a review.

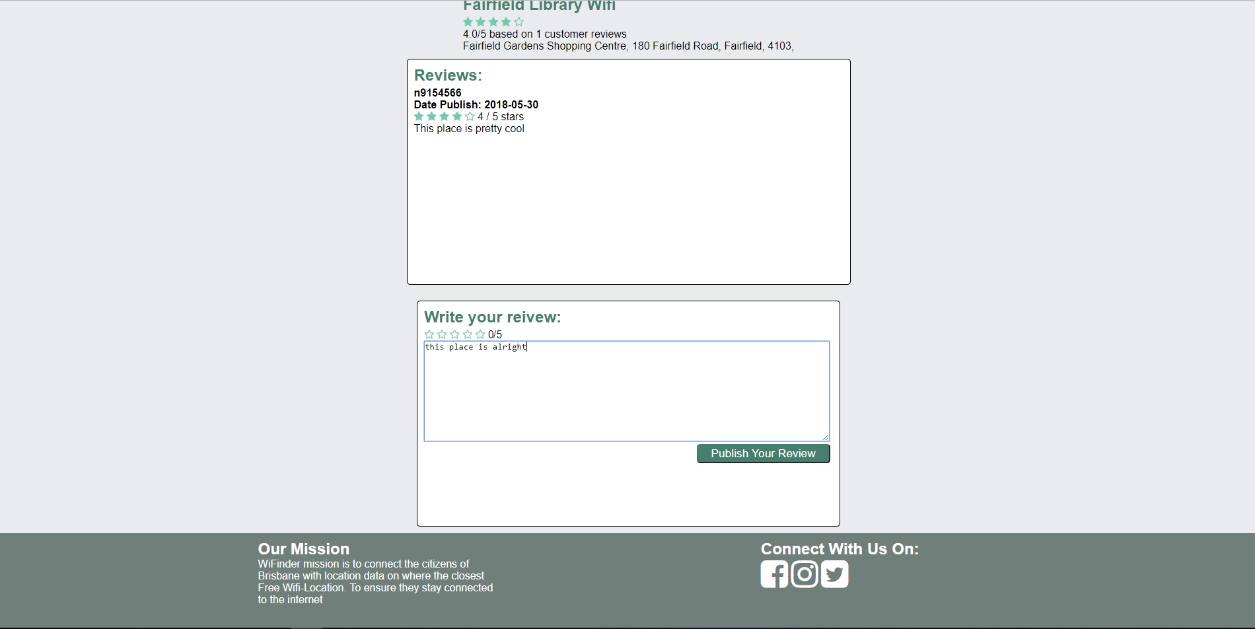


Here the output of the review being posted on the site.

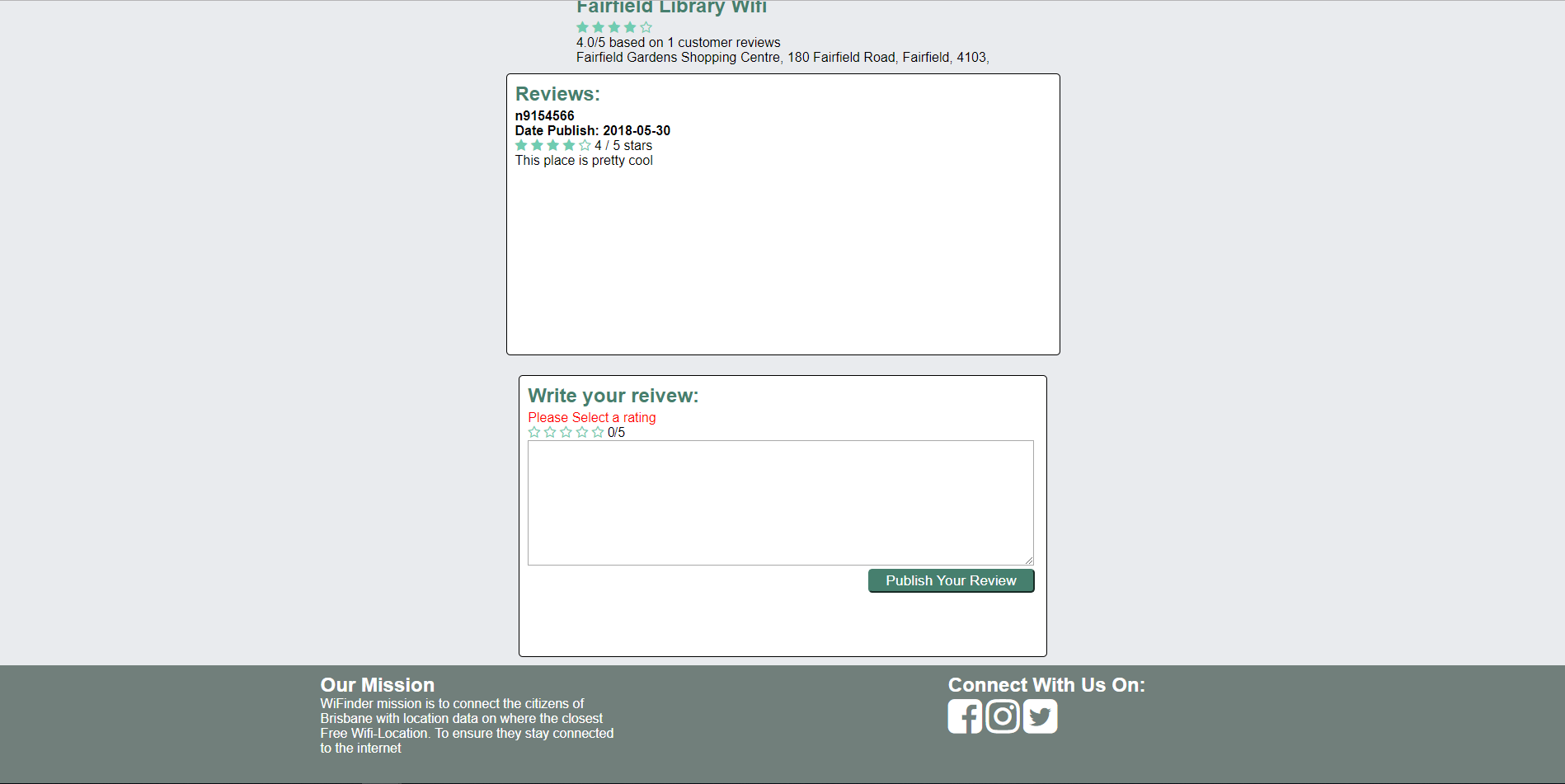
## 

## Valid input for writing a review

For a valid input for the review both a rating and a review description must be filled if one of them isn’t filled then it will shout an error as shown in the screenshots below.

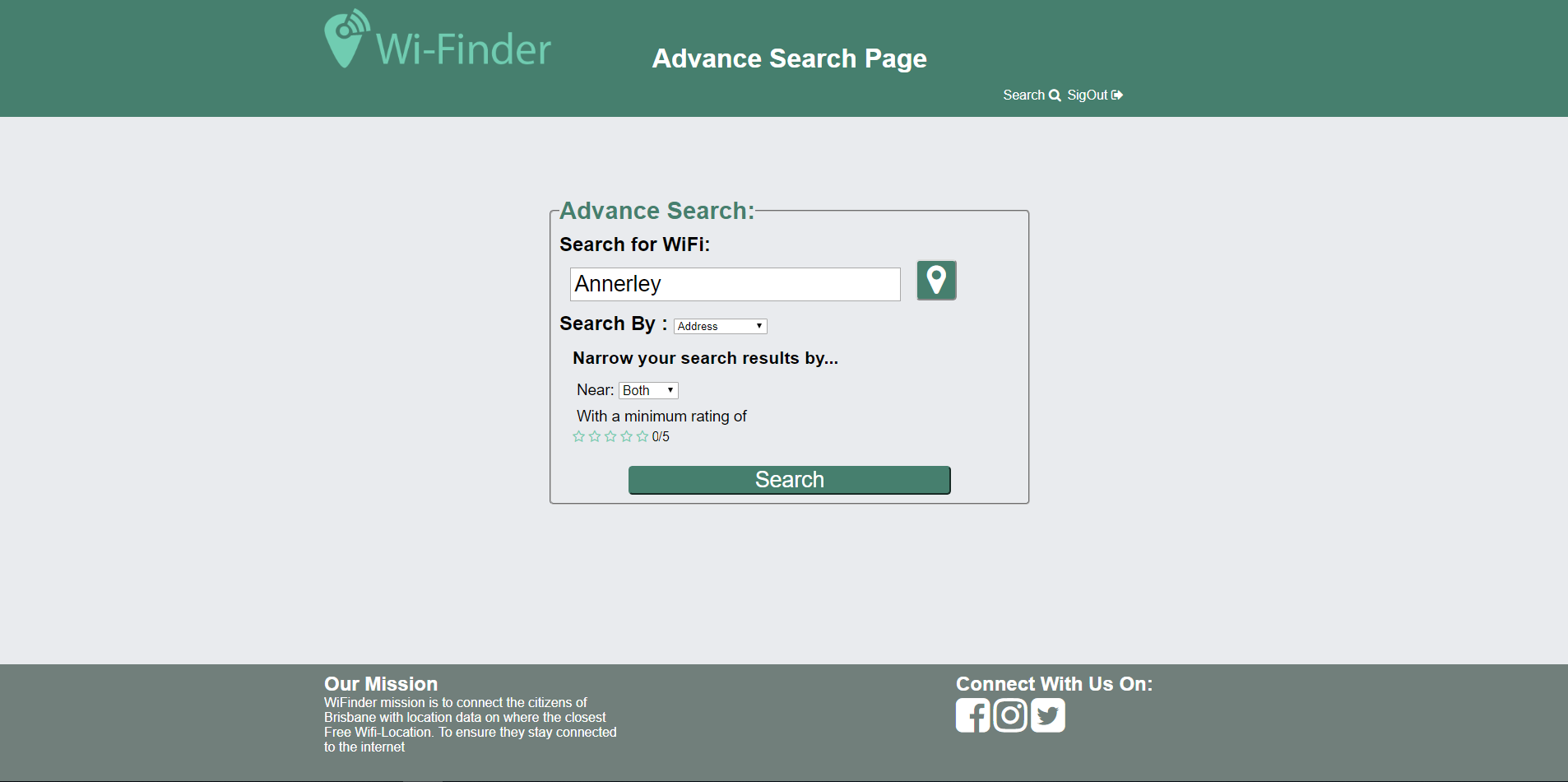


In this case the was no rating selected so when the user submitted it spitted out an error message and their review wasn’t added to the database.

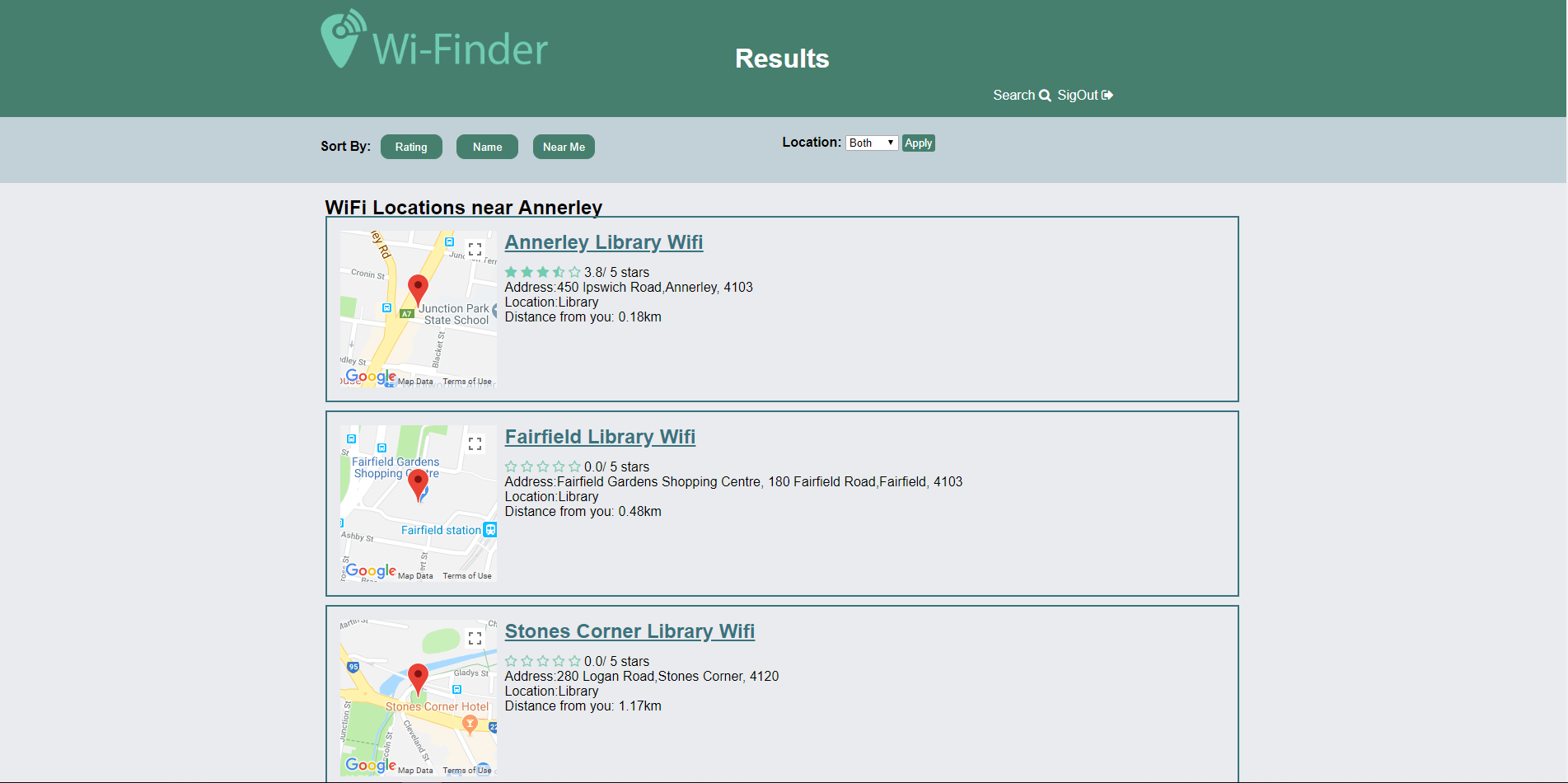


## Searching for an item that exists in the database

This test case we are searching for an item that exist in the database in this case it is annerley.

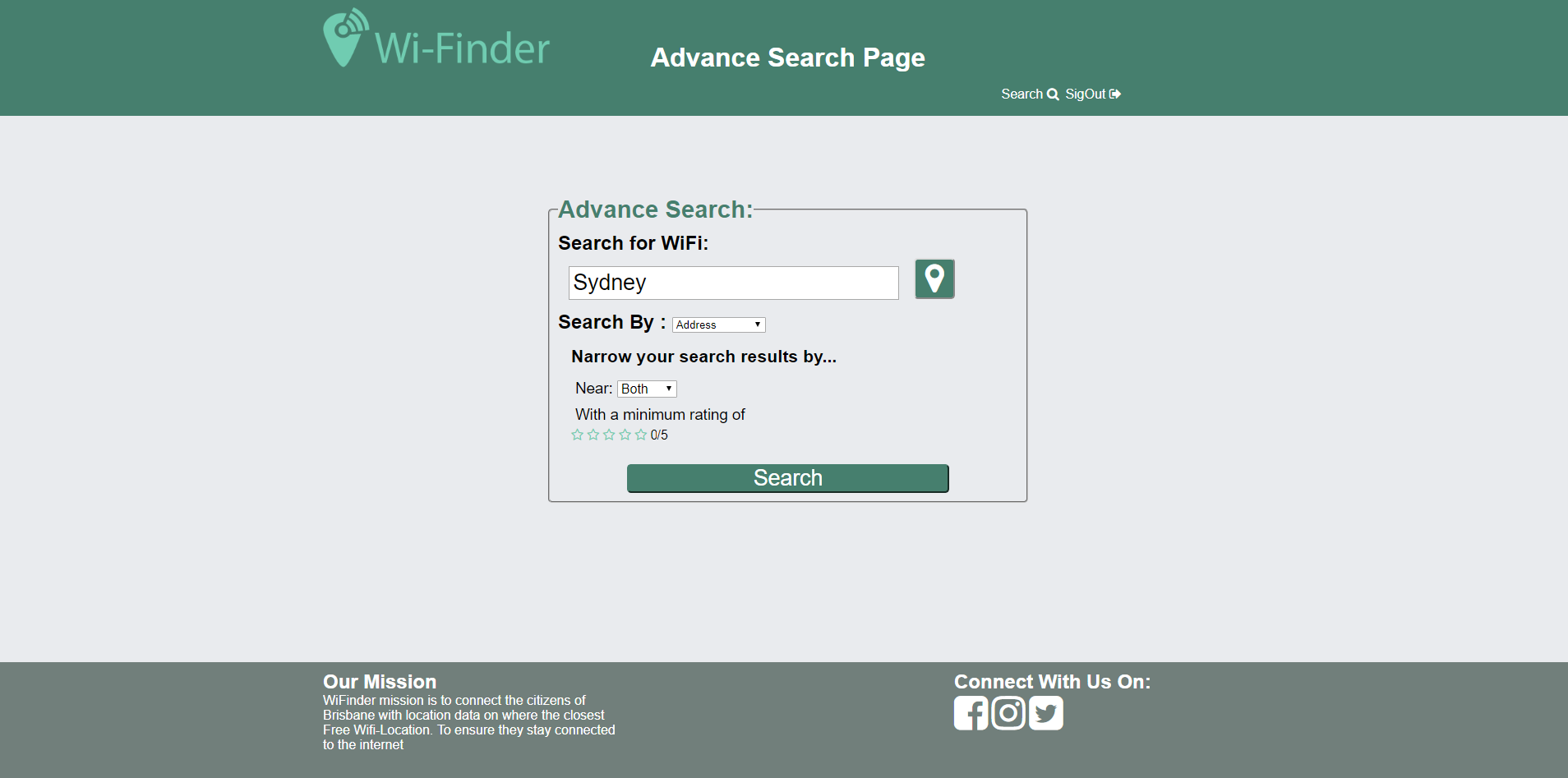


The results it shows are then displayed so the user can see.

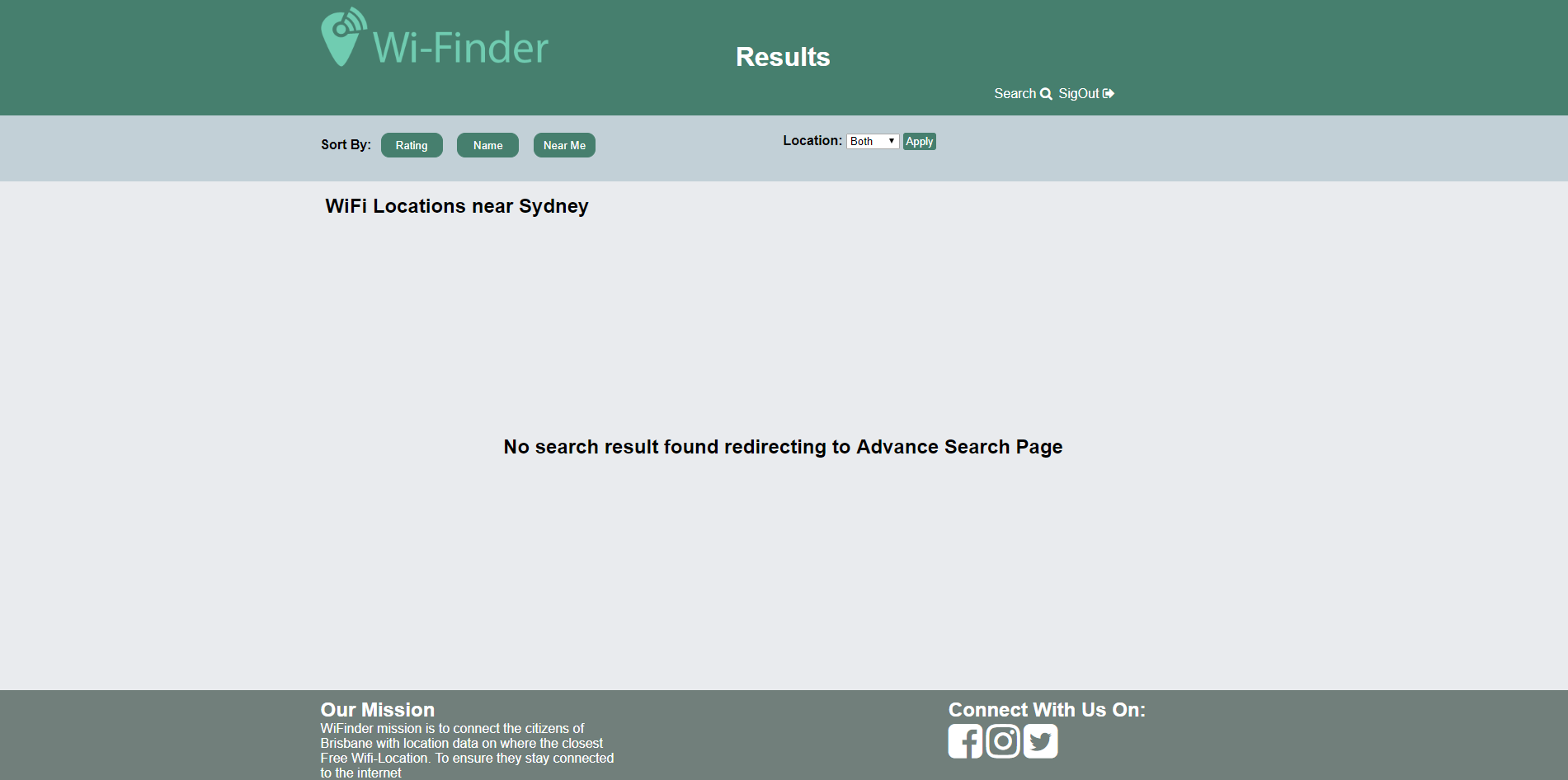


## Searching for an item that doesn’t not exist in the database

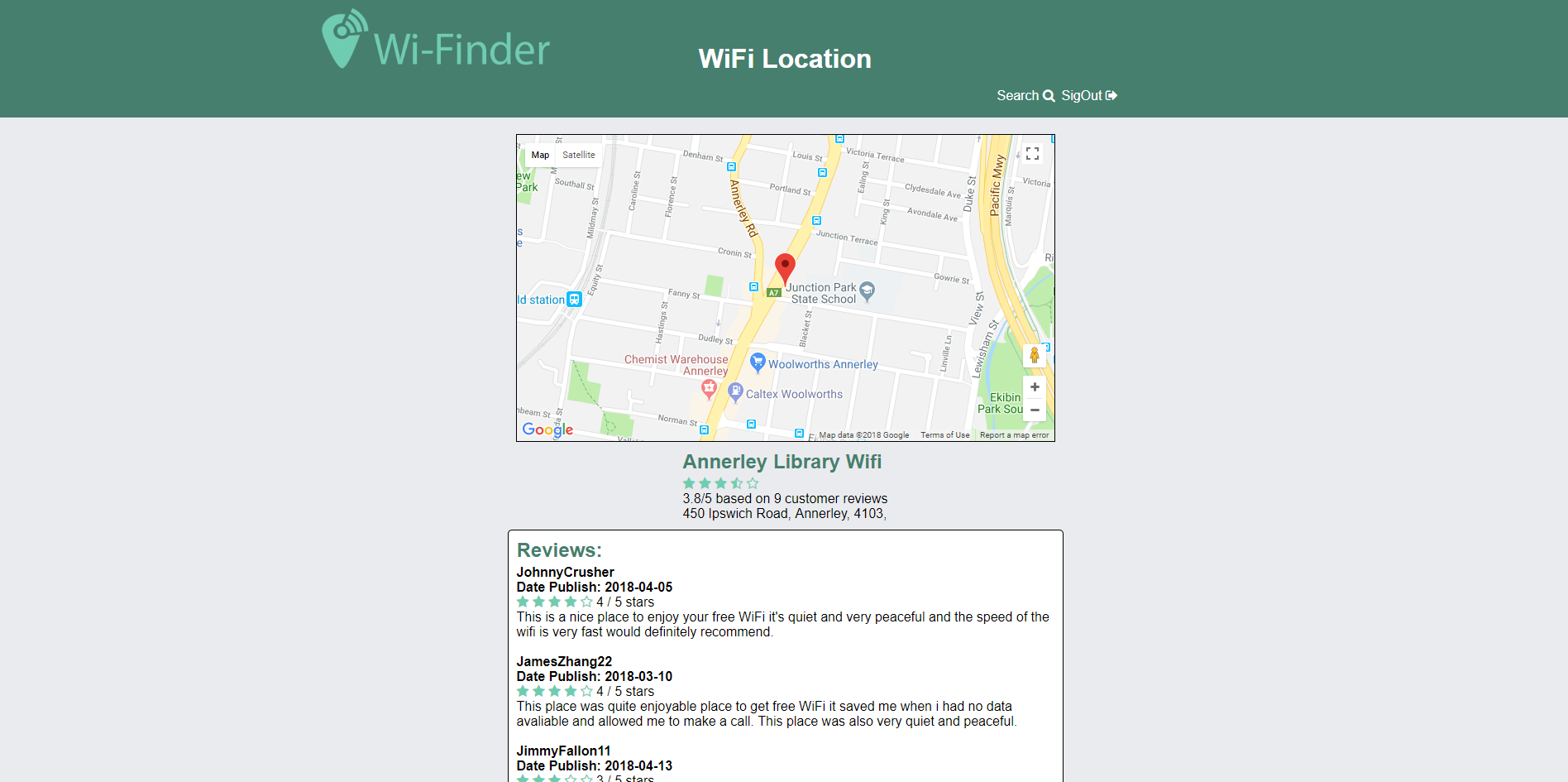
In this test case we are searching for an item that doesn’t exist in the database which is sydney



Then shows that there are no results in the search results page which will then redirect to the search page so the user can perform another search.



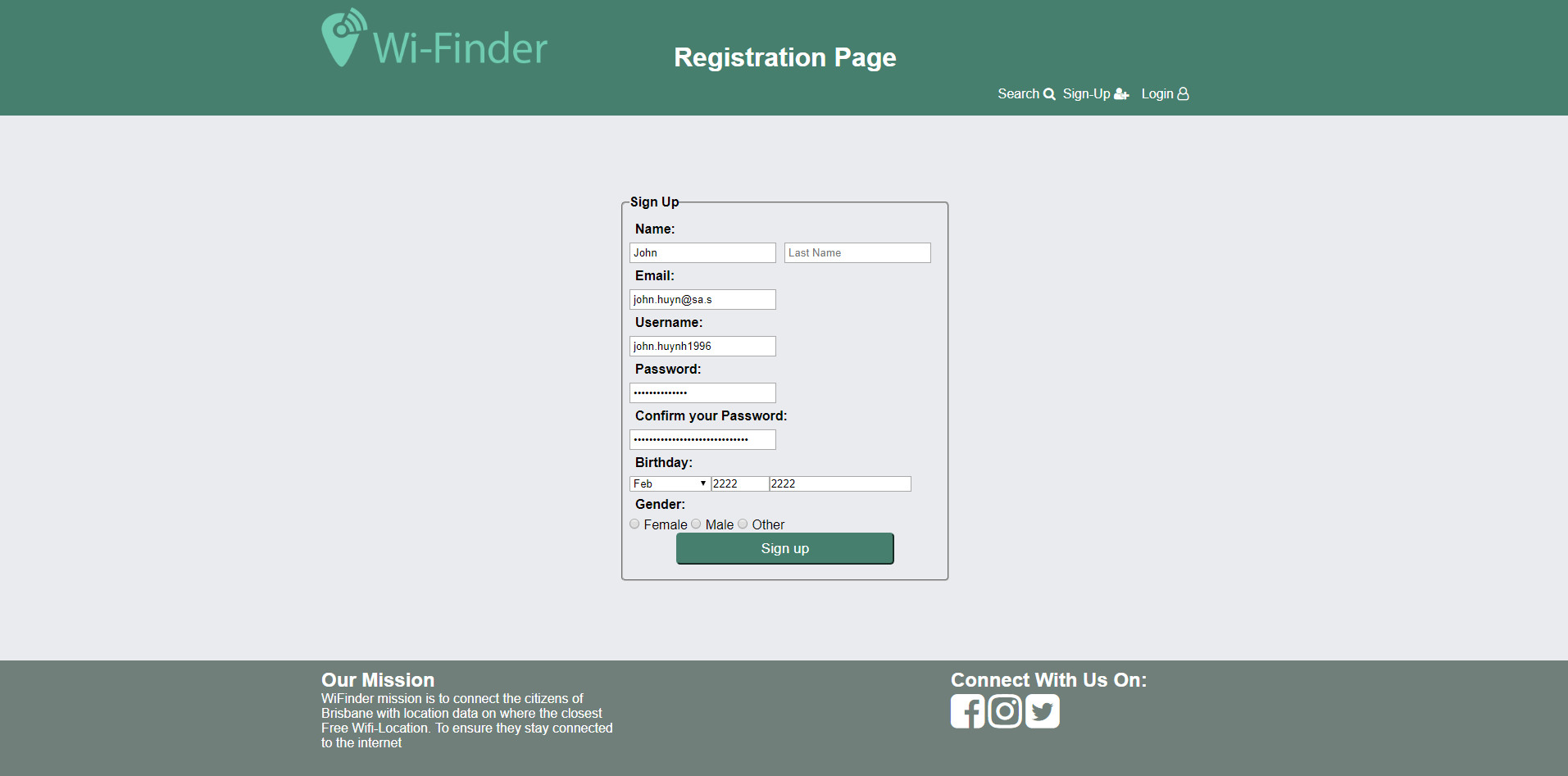
## Individual item page



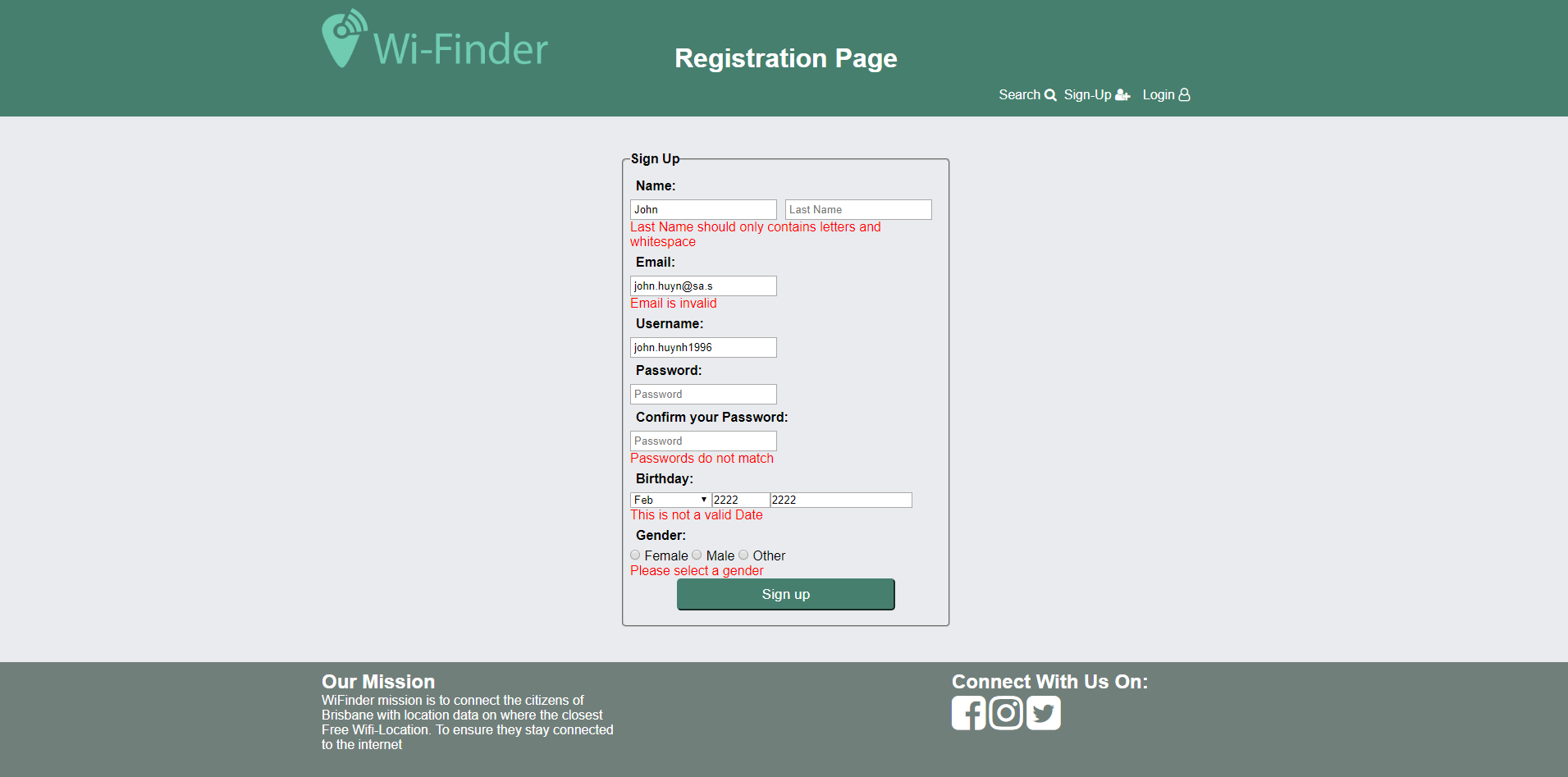
This page shows the individual item page

## A user not being able to register with invalid data including examples of invalid numeric, alphabetic, and email

This test cases we are testing if the registration form has validation by entering values that aren’t allowed in these boxes or invalid dates or mismatched passwords.

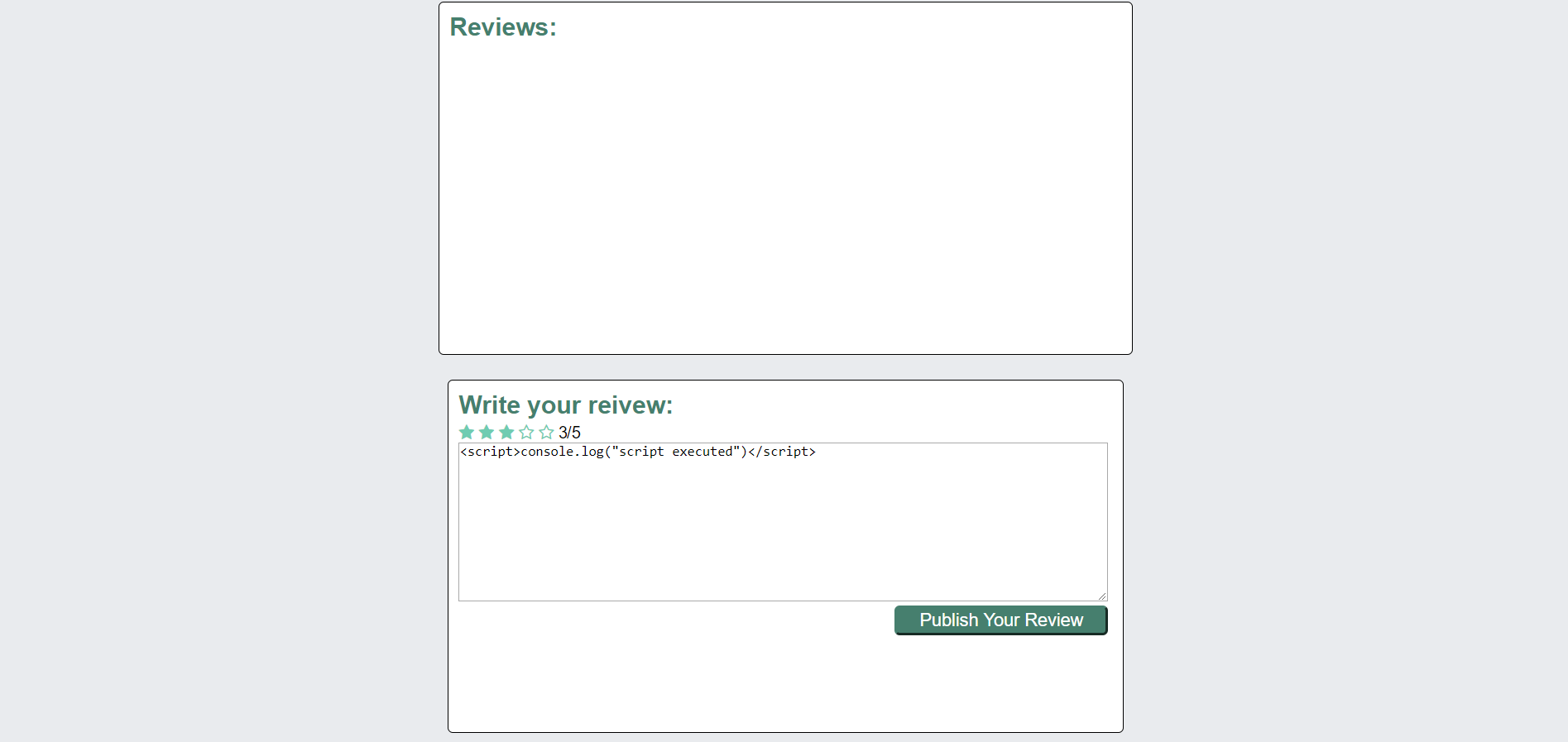


After submitted the form it then display these errors that the user needs to correct before they can become a registered user.

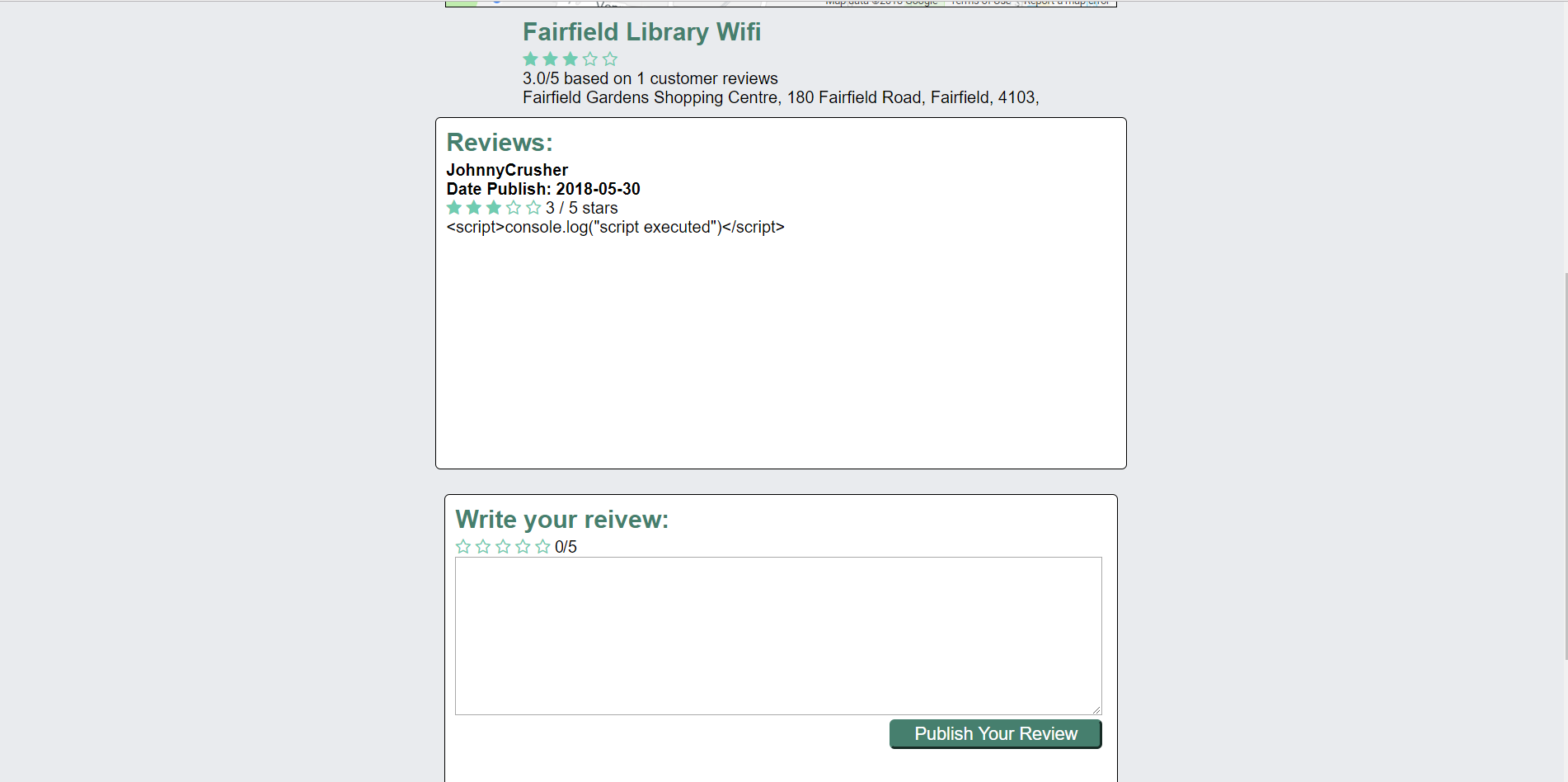


## Attempts to use a cross site scripting attack that are unsuccessful

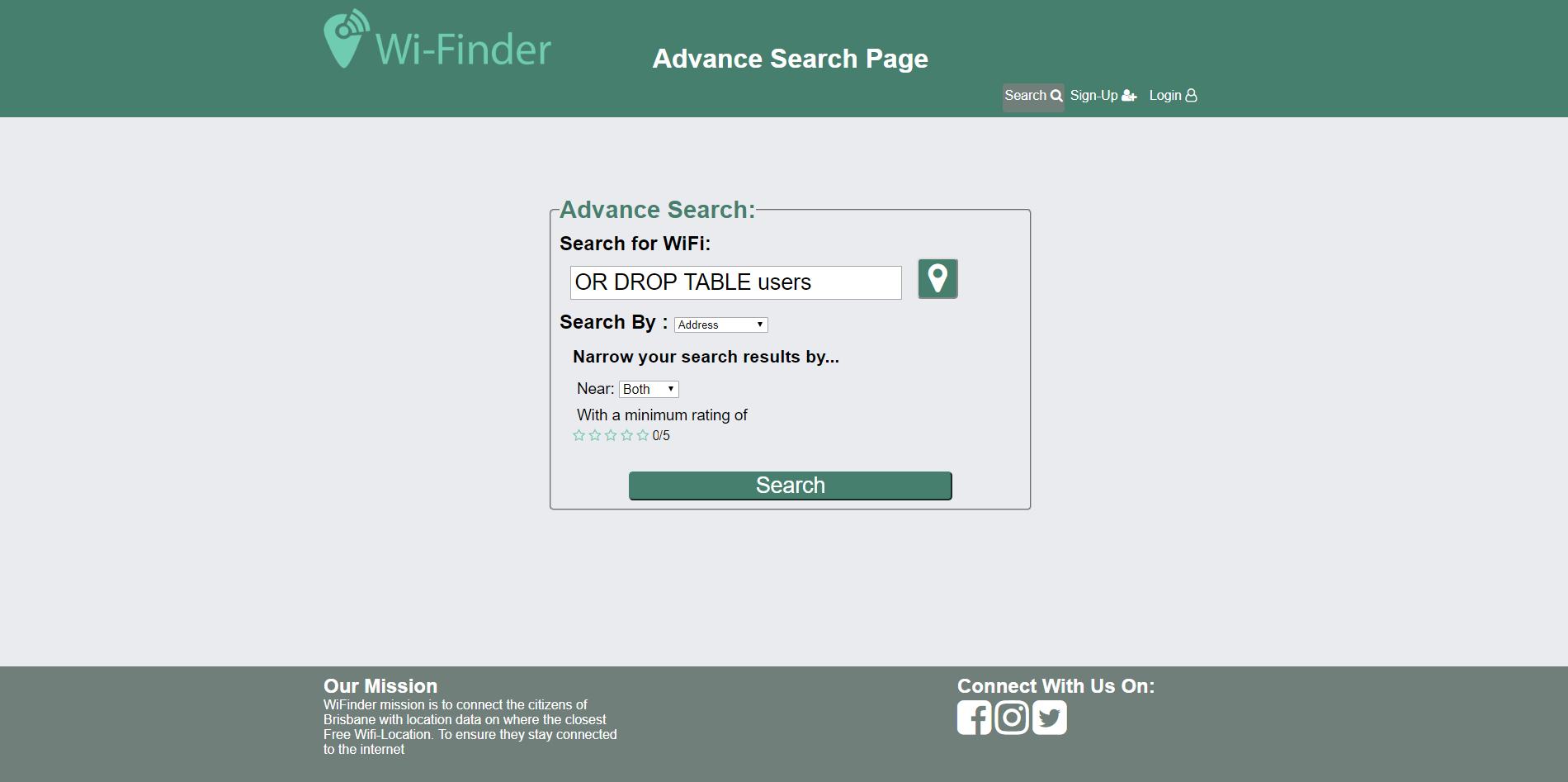
In this test cases we are demonstrating XSS attacks a common entry point for our web application is writing scripts in the review page. As if successful then when the user clicks on the page a script will be executed.



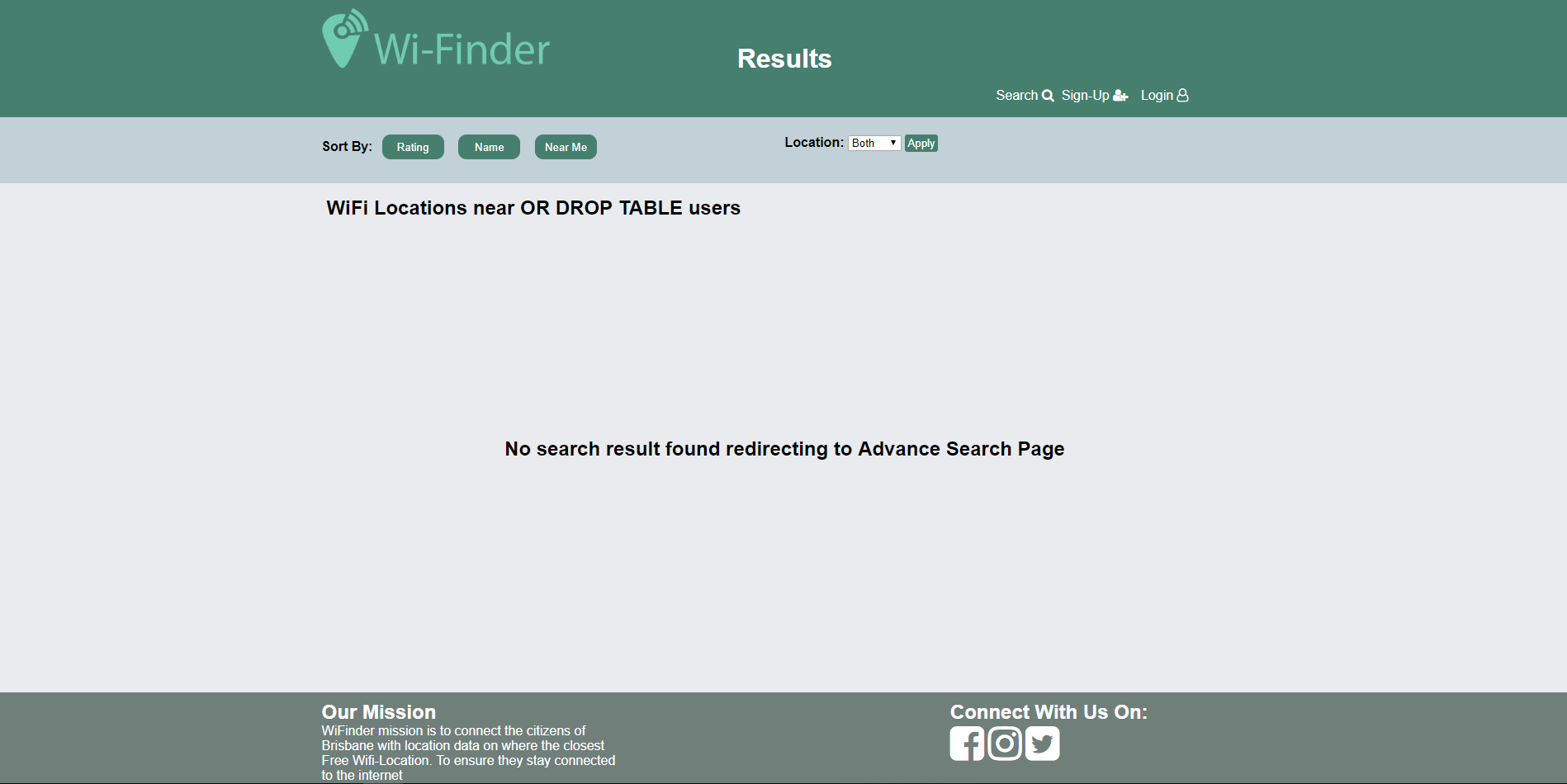
Upon performing the XSS attack it is unsuccessful as instead of executing a script in printing the script in the html document hence not executing it.

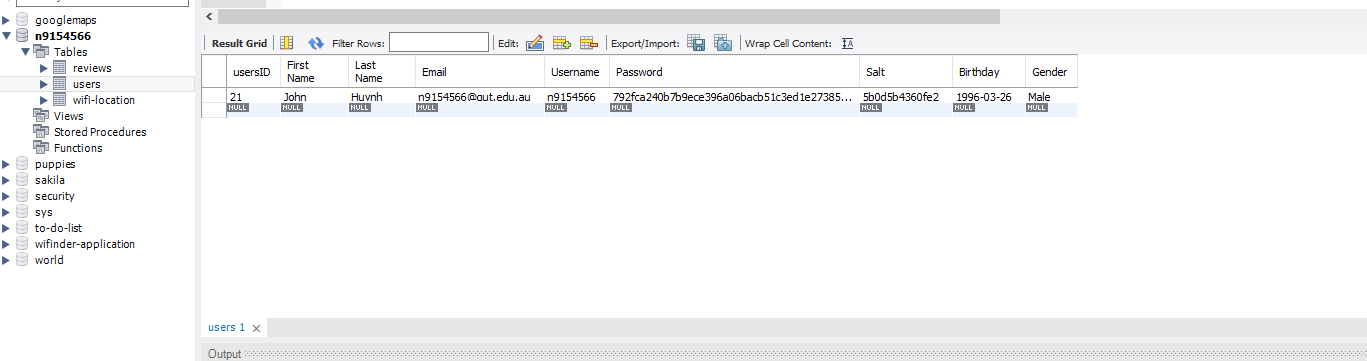


## Attempts to use a SQL attack that are unsuccessful



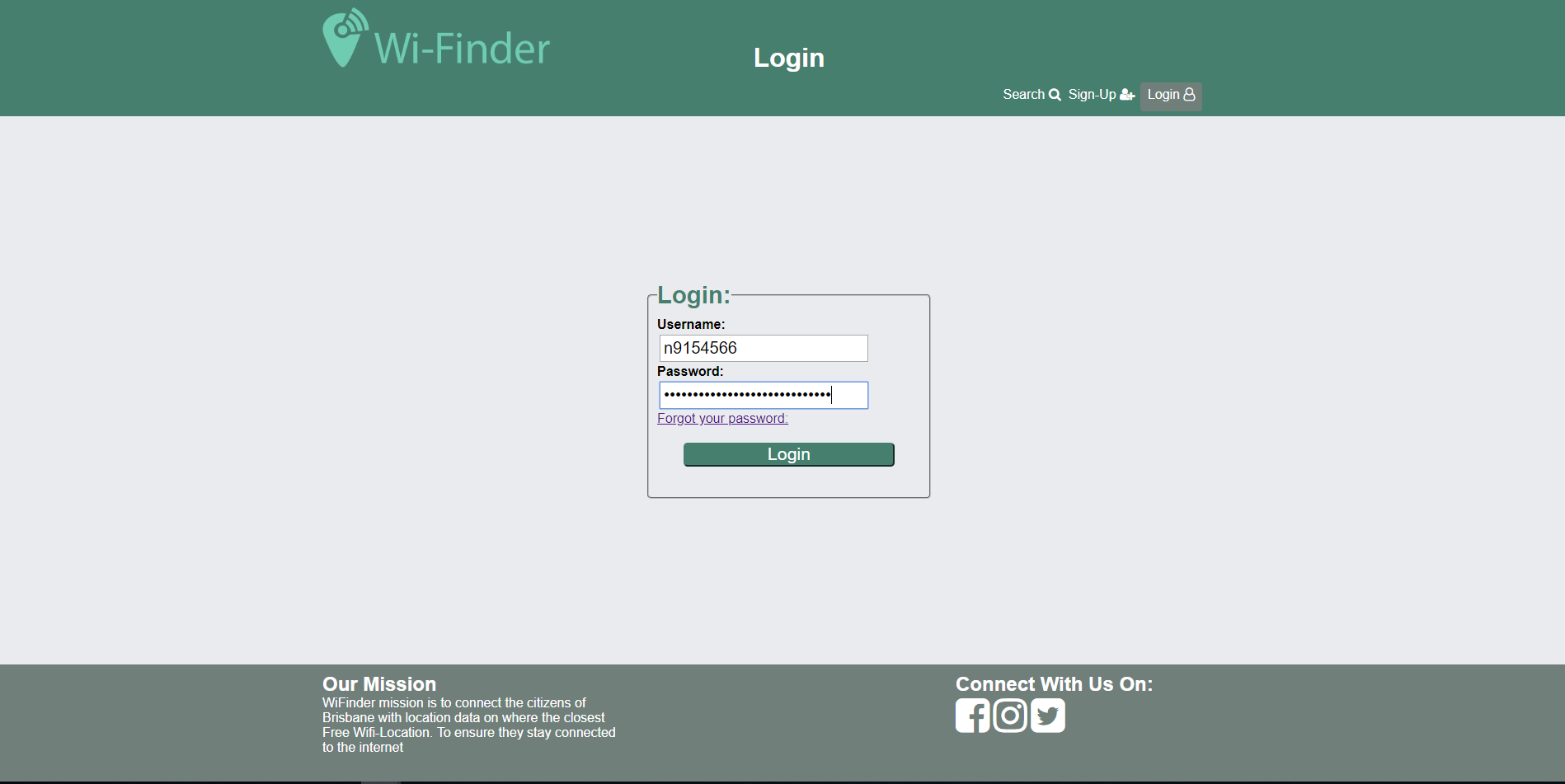
After performing that SQL injection command it had no affect to the database as upon observing the schema the user table is still there



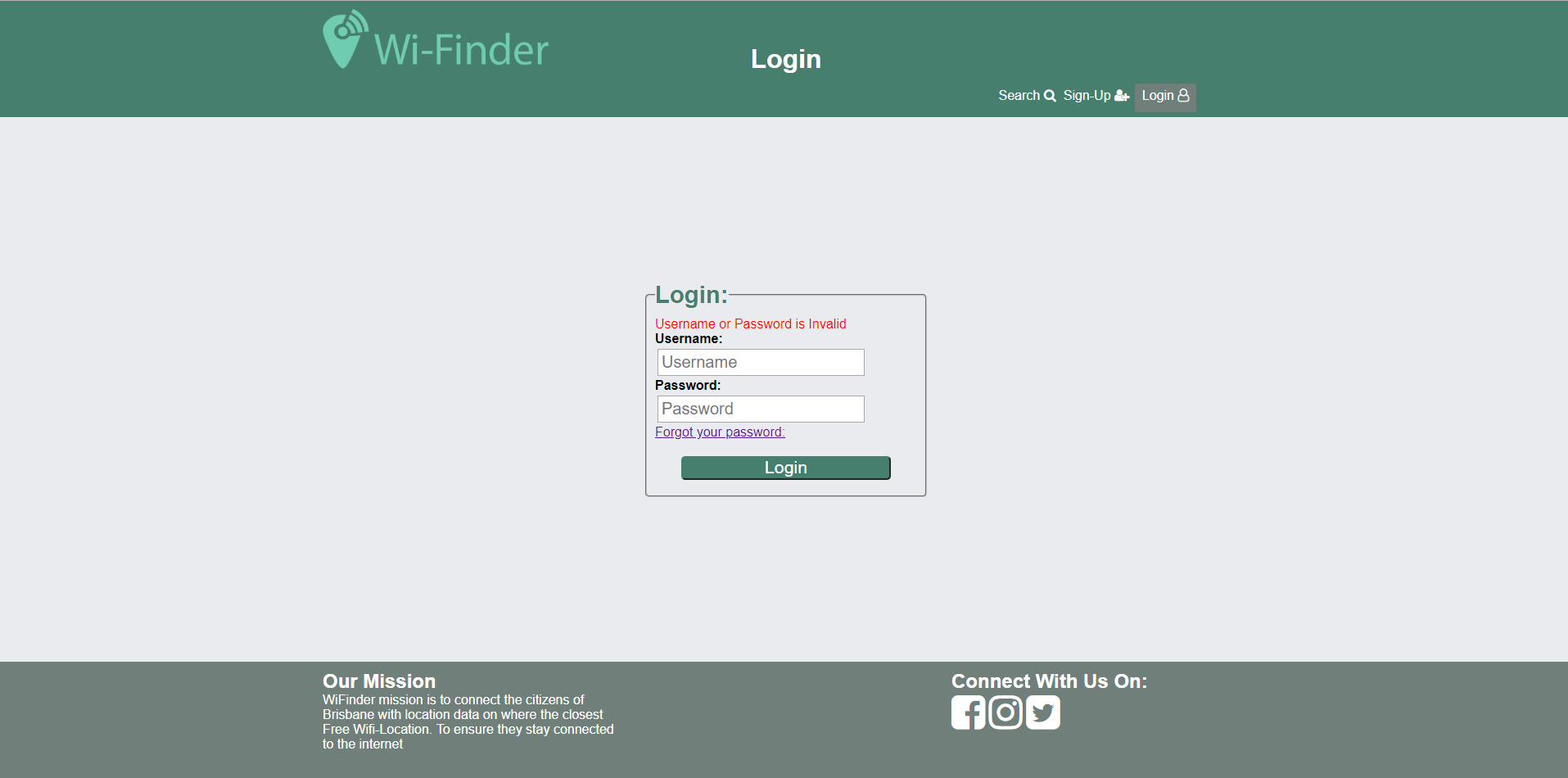


## A user not being able to log in if they are not registered

In this case a user is trying to log in when they are not registered.



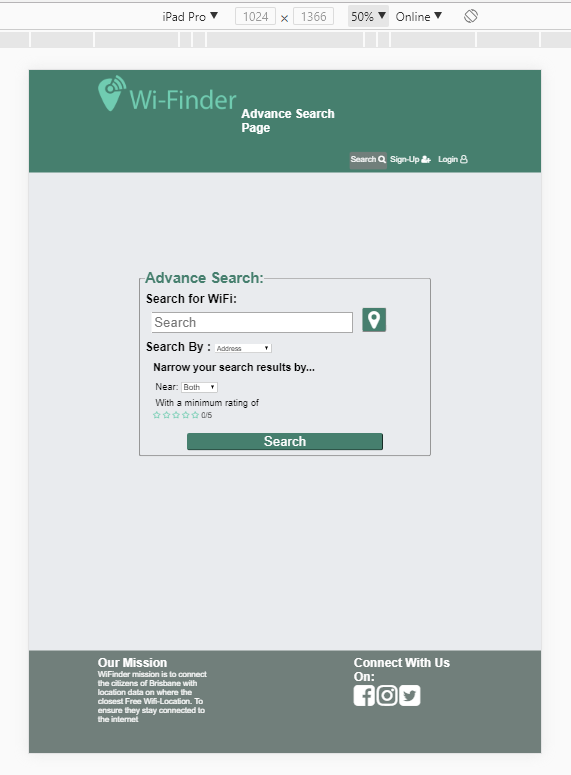
Upon submitted their details for their non existing account they are greeted with an error message prompting them either the username or password is wrong.



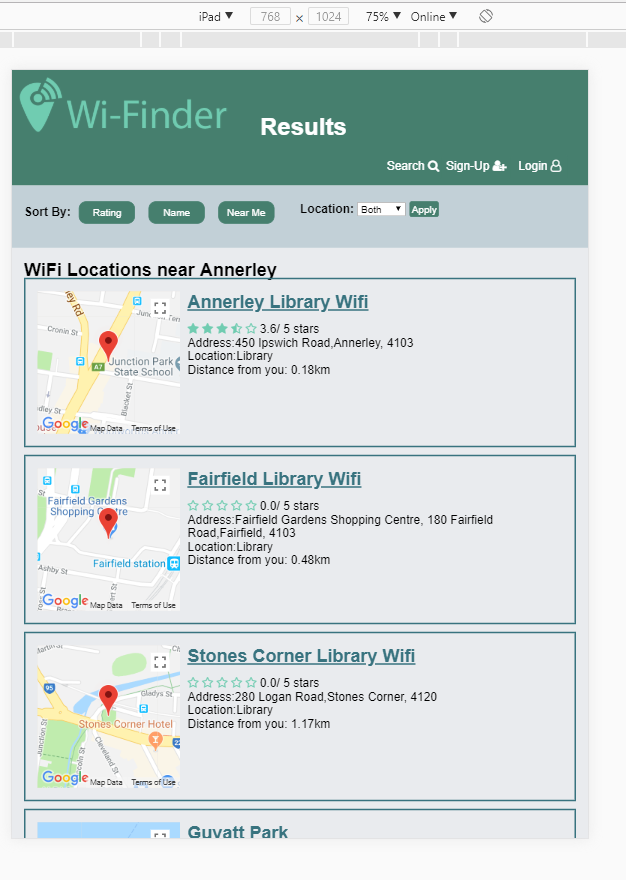
## Add on #3 Mobile Ready Design

### Evidence of site operating gracefully in multiple resolutions

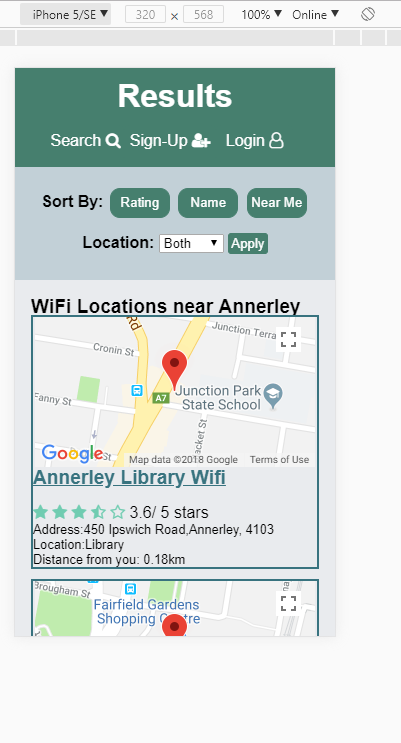
In this test case we are testing if our page works on different resolution. The first resolution is of an iPad Pro with a resolution of 1024x1366 the page seems to display correctly.



The second resolution is a resolution of a typical iPad which is 768 x 1024.



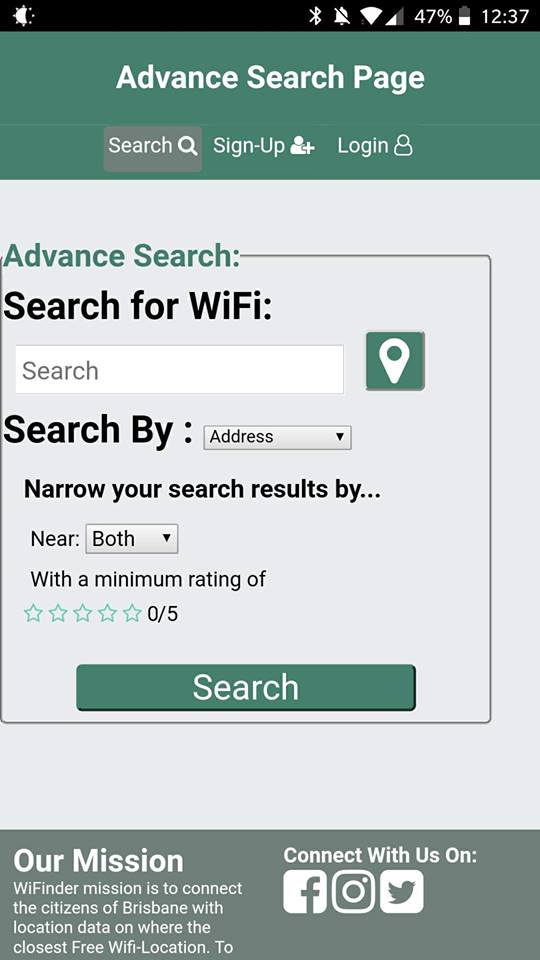
This final resolution is one of an iPhone 5 which was the smallest resolution our website had to be designed for and it displays perfectly.

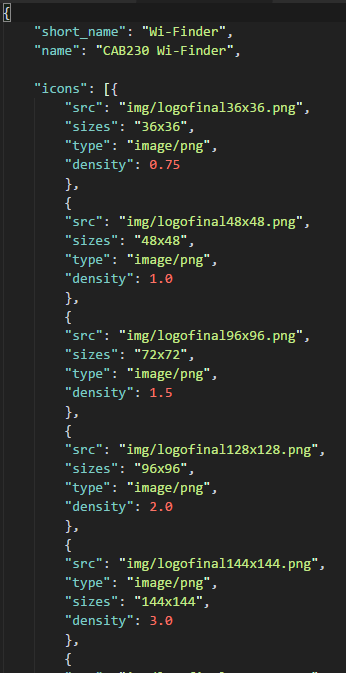
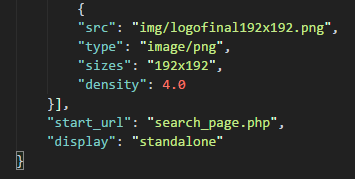


### Evidence of Mobile Ready (site accessible from a button on mobile phone)

#### Android

To show that our site was mobile ready we had to add our website to the home screen which in our case did and it showed our site logo. Upon launch the application it showed it as a full fledge web app with no chrome interfaces.

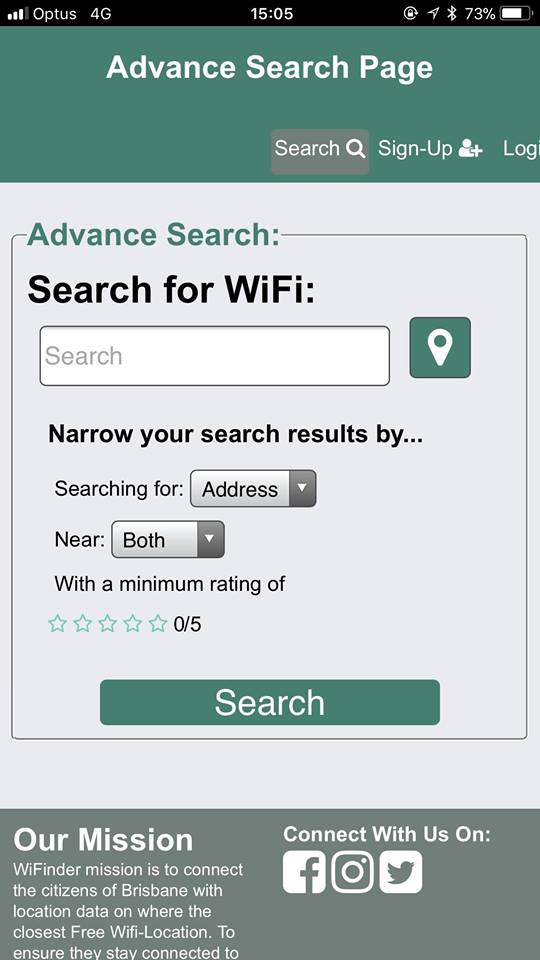


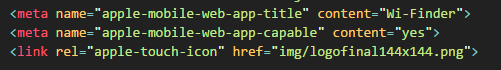
To create an android mobile app a manifest.json file had to be created an in that contains name of the app different image resolutions and then a starting url and display type. Shown in the code below.

^ Here is the line that is used to load the manifest json file so the web browser knows it’s a web app.

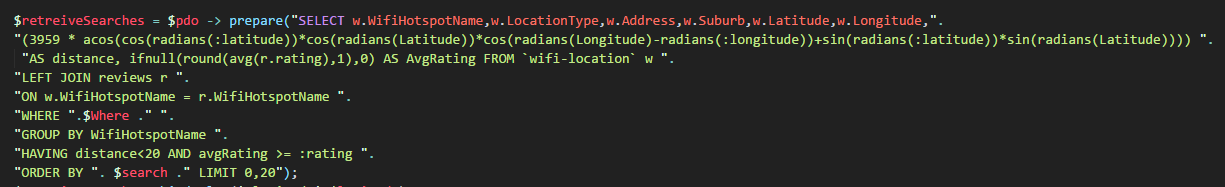
##### IOS

(Had to censor because it contained personal information on home screen as its not my phone but this screenshots shows the application on the home screen and can be identified as an IOS device)



For IOS it required 2 meta tag and 1 link tag to generate the mobile app 1 first meta tag defined the title second tag defined if it web app capable and the link is the icon of the webapp.  


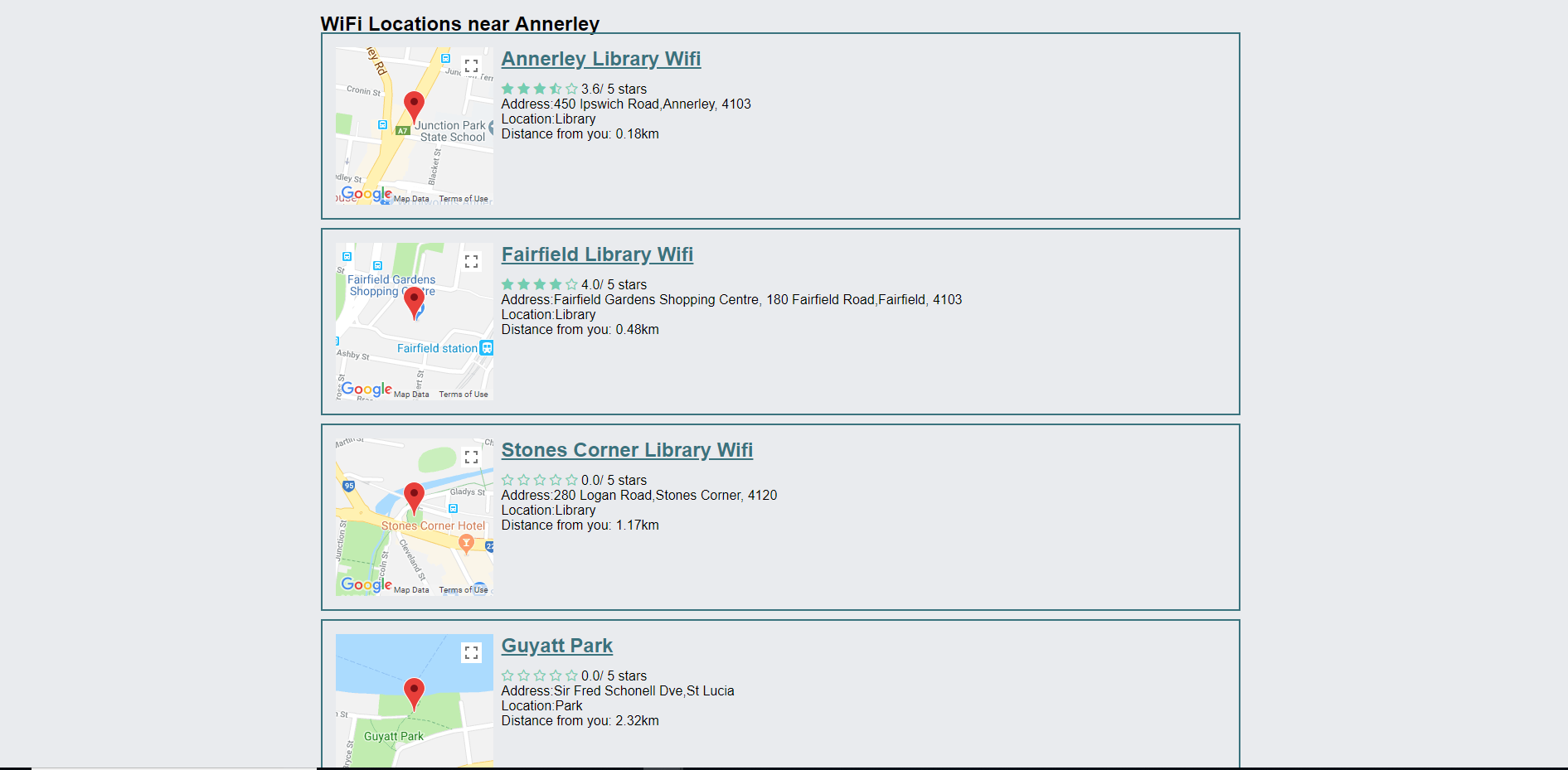
## An example of a SQL Query that is implemented in code and a description of where this Query is located.



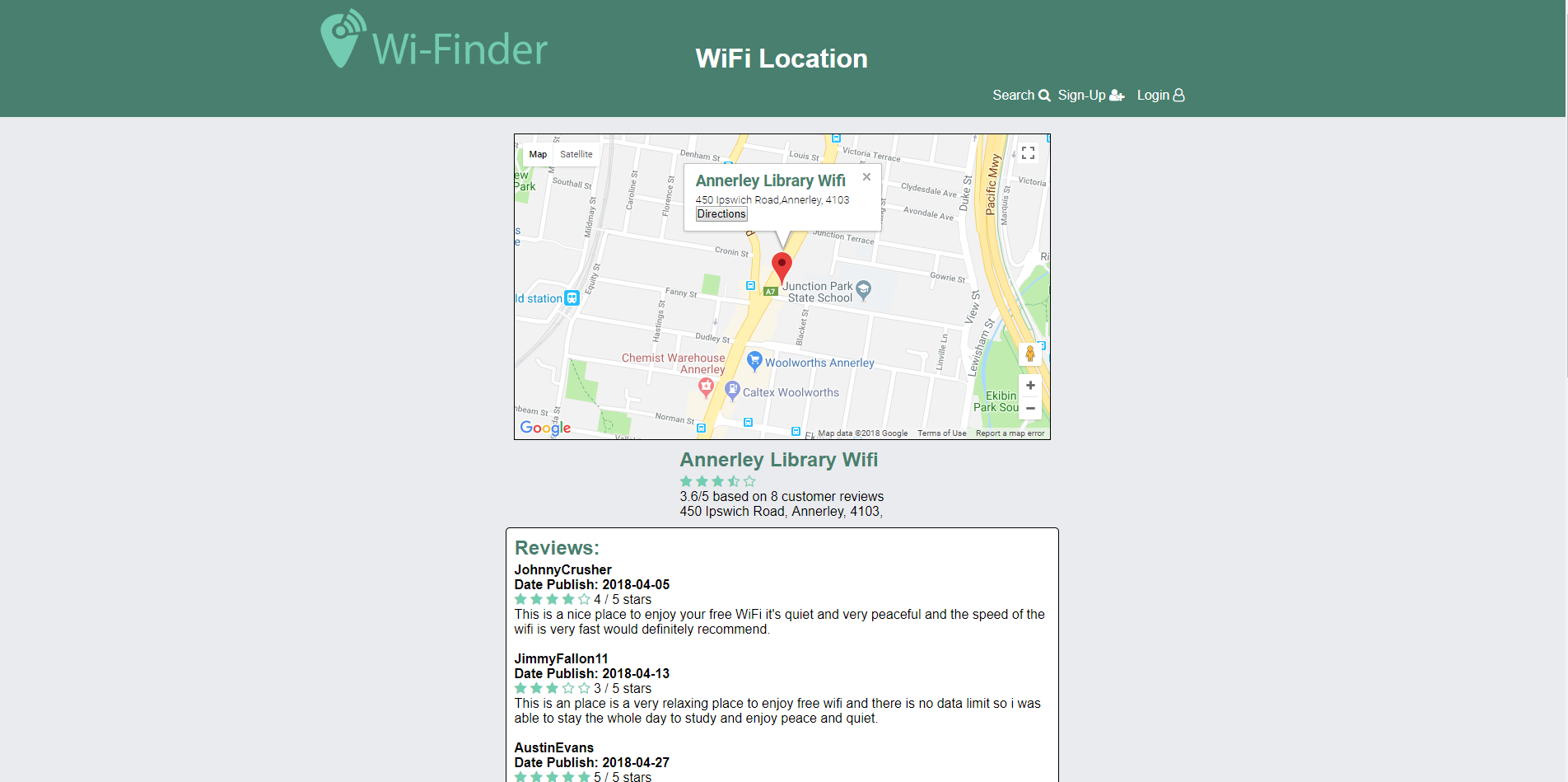
This SQL Query is the most complicated SQL query used for the website in consisted of grabbing data from two tables and merging them. From one table it grabbed the WifiHotspotName, LocationType, Address, Suburb, Latitude and Longitude and also calculated the distance using the Haversine formula. Then grabbed all entries from reviews table and merged the avgReview with the matched location and filter out if it had a distance over 20km and the avgRating is below the specified rating. This query is in the retreieveFromDatabase.php file.

## Add On #1 Maps

The first addon tasked consisted having a map for both the search page and the individual item page. Since we decided to do a tile based design it was a much better option to have all the maps of all the location next to the item name. As opposed to one big map which all the locations this helped make the sites look nicer and not incorporate the static tabular design.

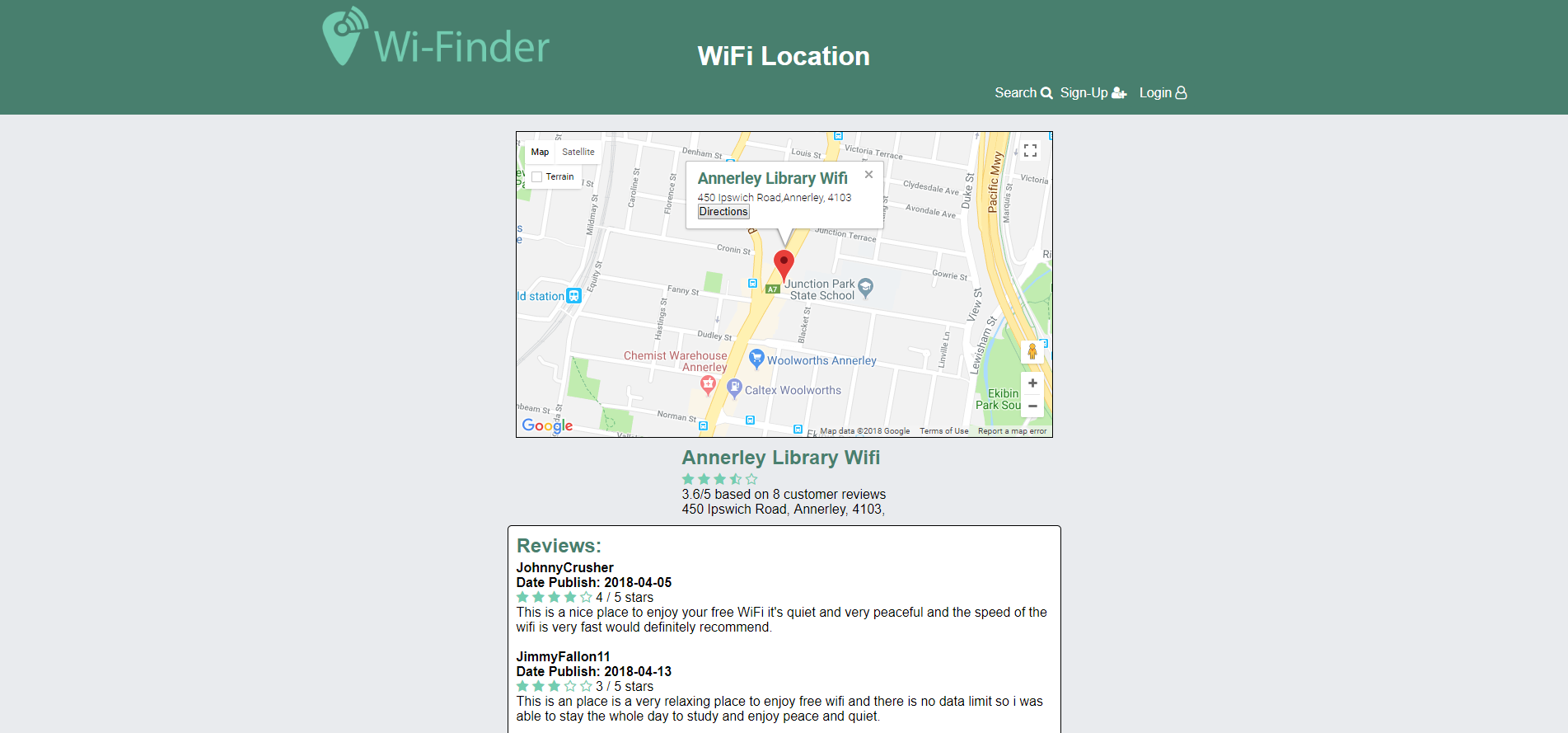


The individual result page has a bigger picture of the map and contains a marker which displays information of the wifi location and as an added feature if the user clicks on the direction button then they will be forward to a google map location marker where they can get directions to.

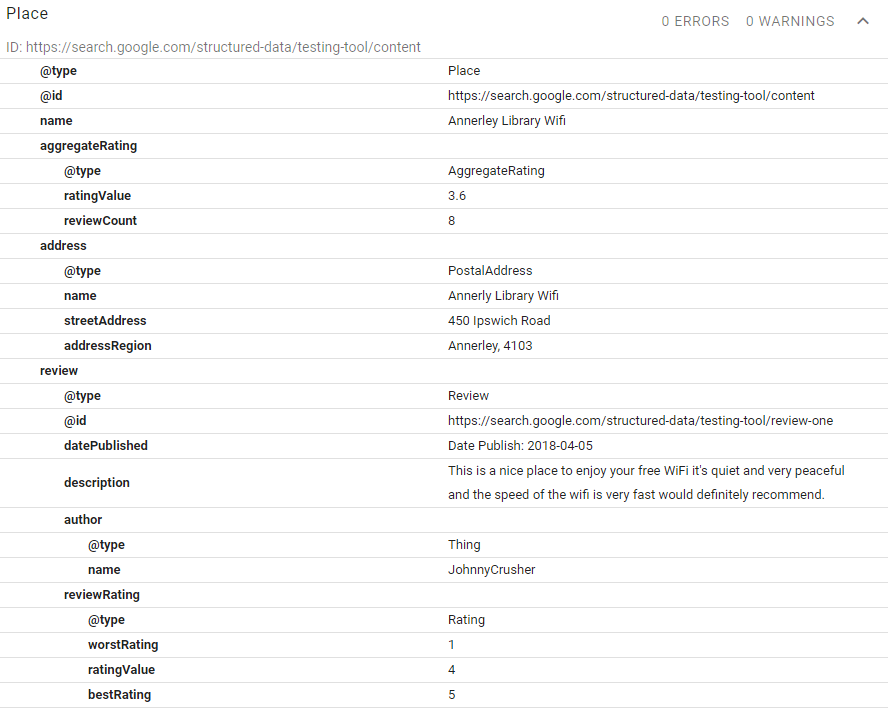


## Add on #2 Metadata and Microdata

This is a individual results page that contains the microdata of the place and review description.



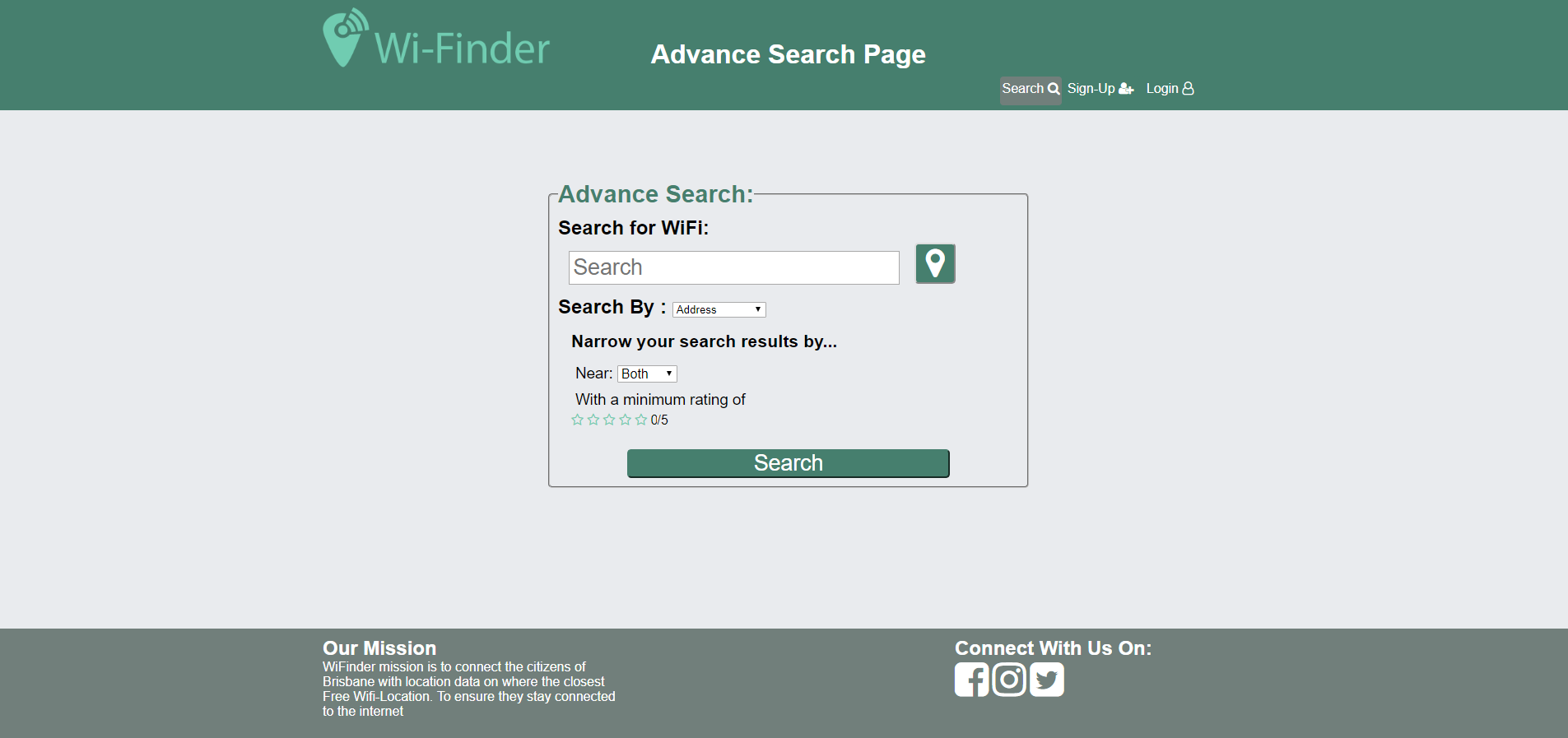
Here is the output of the microdata using the Google Data Structuring tool it shows that this is a place and has its required fields. It also incorporates the review schema as it shows the reviews schema information below.



# Web Design Principles

## User Experience

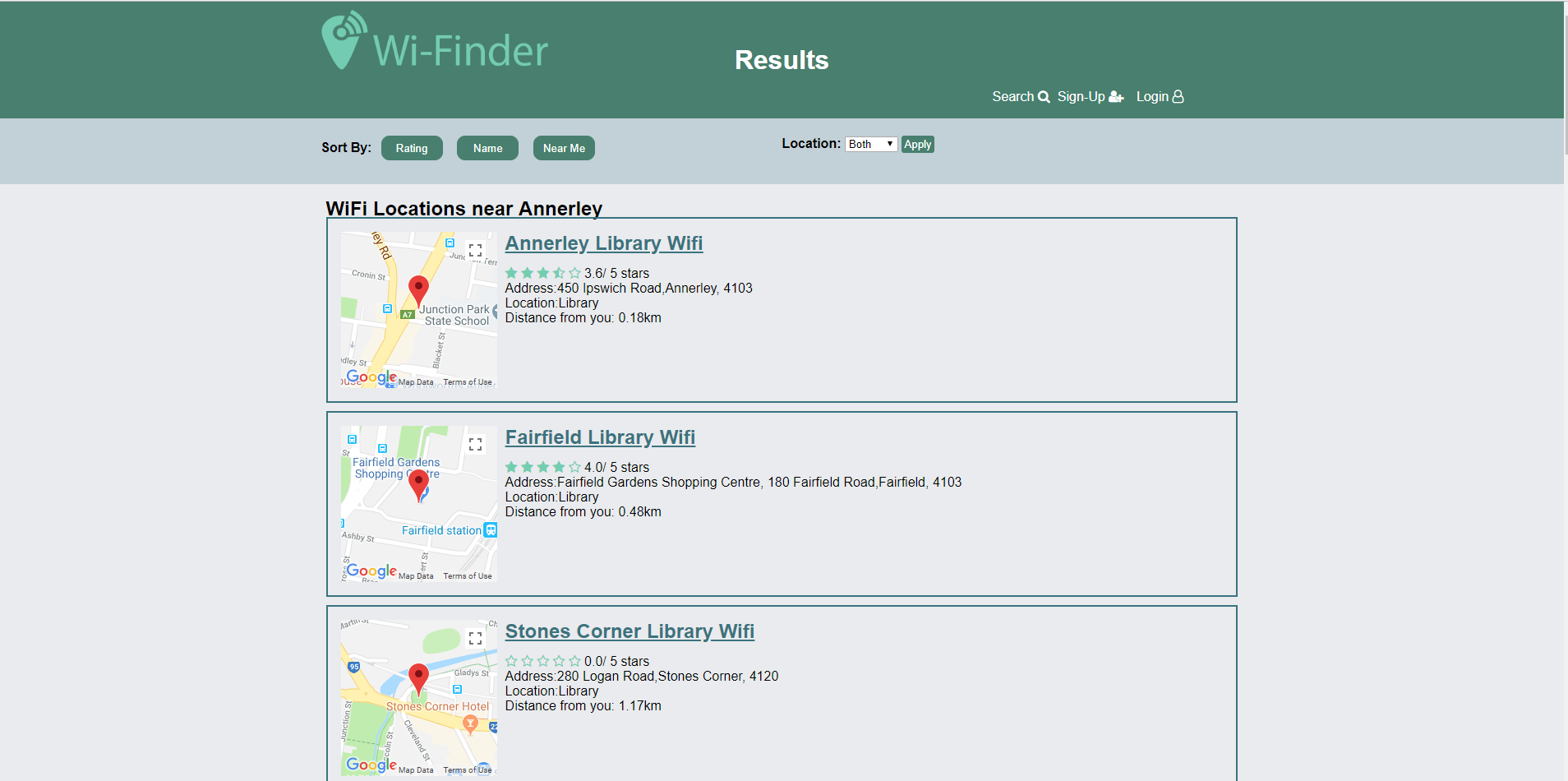
User Experience was an important aspect in our website design as it incorporated a simple design where every function was either very obvious on what its purpose was or had text next to it to describing its purpose. Allowing the user to quickly identify what that button/input box was designed for which incorporated the “don’t make me think!” principle.



For example our search page is quite easy to understand it has a input box where the user can type either the address or name they want to search for. Then below it there a label that allows the user to select the searching mode they want. Then next each drop down box there is label next to it to describe its purpose. Finally, there is 2 buttons one of them is for the geolocation which can be identified by looking at the button with the universally recognised GPS logo and the other one is wide button labelled search. These elements contain a different colour to other elements so it’s easier to differentiate.

## Visual Design

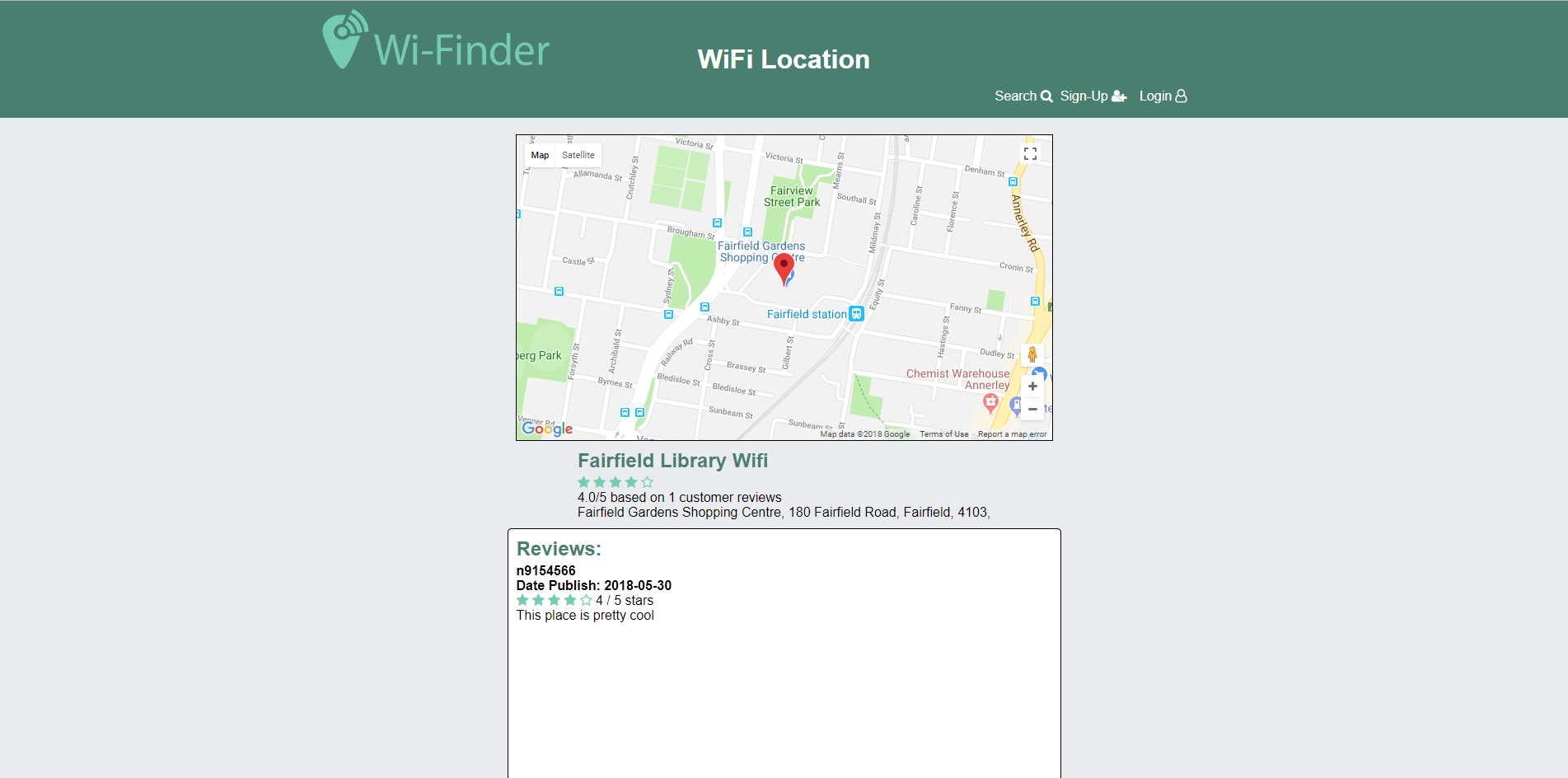
For the visual design aspect, it was decided to keep it simple by having a consistent colouring scheme through the whole website. This consistent colour scheme consisted of a dark green header bar, grey content bar and a greyish green footer and all other elements followed a similar colour scheme. For the contrasting aspect of the visual design it was decided to use dark backgrounds with lighter text and lighter backgrounds with darker text. This made it easier to read the text and contributed to the overall readability of the website and improved the overall look and feel.



This page uses the aforementioned design patterns which including having dark text on light background and vice versa. It also has a very simplistic interface where it easier to read and understand each elements purpose.

## Page Layout

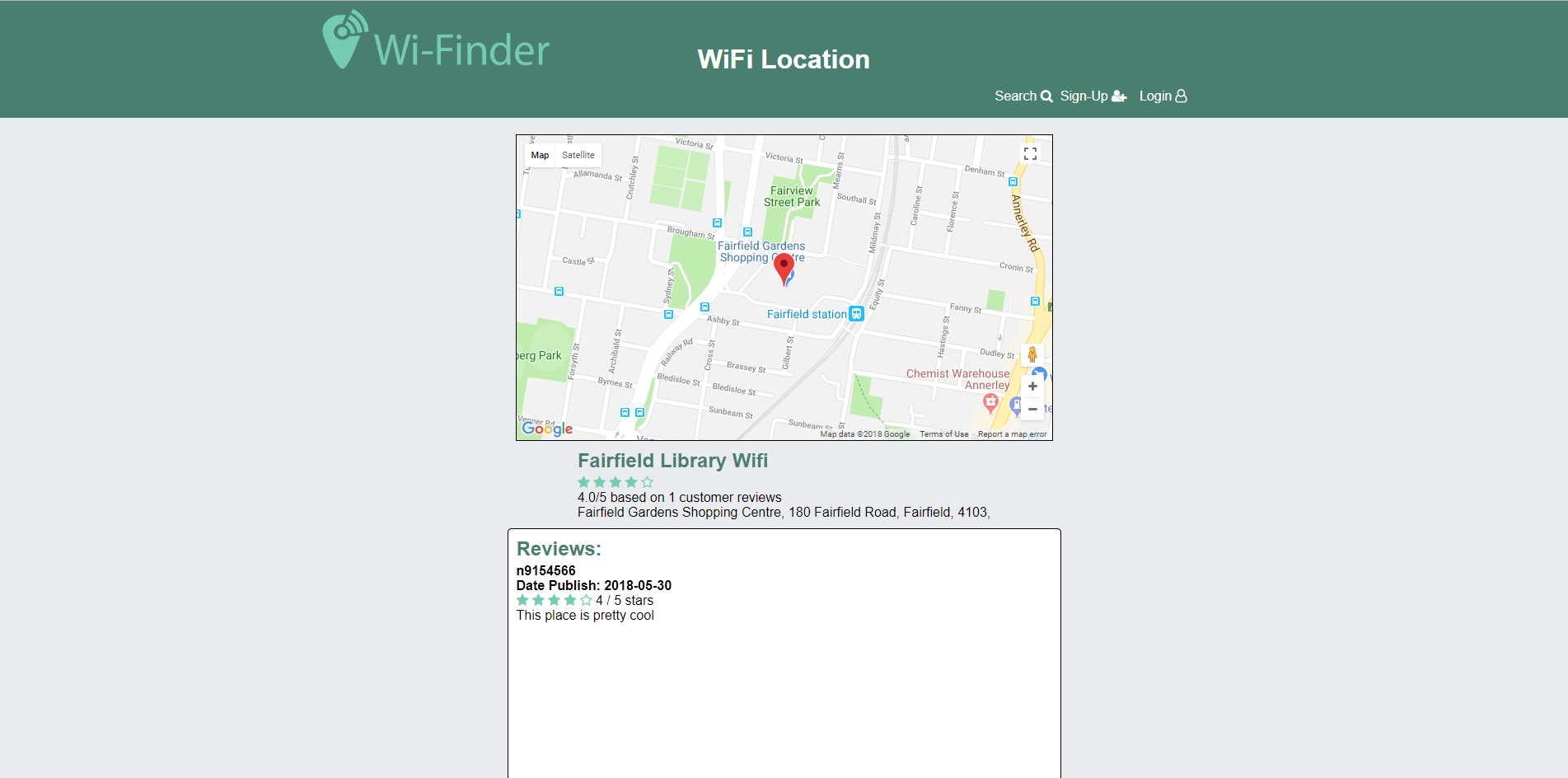
For the page layout aspect, it was decided to use some sort of grid layout which was implemented using CSS Grid Layout. Which made the page look very well structured and professional thus making the page more presentable. Another design aspect was separating the different sections from each other this was achieved by having different background for each section of the page section the header, navbar, footer and content. Also, each subsection had their own colour scheme but not to the point it overpowers the user with many different colours. Another design consideration was using a centred page design where most of the content would reside in the centre and the sides would contain active whitespace and will dynamically adjust as the screen width gets smaller.



The page above incorporates the aforementioned principles where it is using a grid-based layout, there is an obvious transition between the different sections and incorporates page centred design where the content is centred in the page.

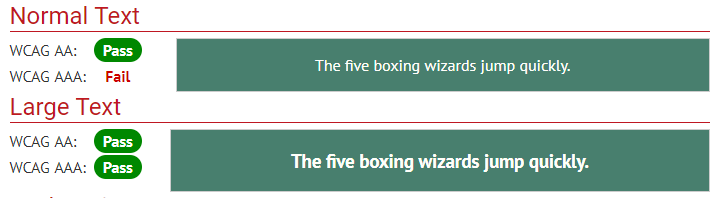
# Standards

For standards consideration to cater for people who require the use for accessibility features this requires having sufficient contrast to make things easier to see for people who are colour blind. The colour isn’t overwhelming as well which is a huge benefits as it won’t causes seizures. All images have alternate text as well which helps users who are colour-blind identify the purpose of the picture. This website uses a lot of structured elements such as h1,h2,h3 etc to emphases important contents such as titles, sub heading and location names.



Take the individual results page for example the contrast is great where it has dark colour on light text for the heading and for the content it has light background on dark text. It also has different headings to show the emphasis of these titles making screen reader job even easier.

Header with dark green background and white text using WebAIM Color Contrast Checker.



Content with grey background and black text using WebAIM Color Contrast Checker.

## 