



Knight Foundation School of Computing & Information Sciences

Summer 2024 Senior Design Project (Capstone II)

Knight Foundation
School of Computing
and Information Sciences

Weather App (U.I. Development)

Student: Alejandro Pazos (U.I. Developer)

Instructor/Faculty: Seyedmasoud Sadjadi, Florida International University



PROBLEM STATEMENT

Would you like to know the present weather conditions anywhere?

Everyday can serve to be unpredictable, where the current weather conditions can massively impact daily plans and decisions. Cancelling a trip to the beach or plans in another country. To address these types of issues, our team has created a real-time, updating, weather forecast site.

This weather application includes:

- Javascript, HTML, and CSS as the programming languages used.
- React; a library that builds a responsive user interface (website).
- OpenWeather's API is used to fetch the current weather data of the city provided.
- Visual Studio Code is used as the IDE containing the application.

SUMMARY & IMPLEMENTATION

What is our app?

The app is built using React, which creates our website. Done within Visual Studio Code, Javascript, CSS, and HTML are the used due to React being a library of Javascript. The way data is fetched is through links acquired from the OpenWeather API. Providing the API key after subscription, to allow the API to fetch real-time data for the app [1].

Designing the UI:

Being in charge of the design and aesthetic of the application, I had to focus more on learning CSS and HTML [2]. Prioritizing the structure, content, and layout of the application with the built-in react "index.css" and "App.js" files. App.js maintained the display of the website, using HTML to organize the sections of the app. Placing the search mechanism and inputting the proper API data in each section specified ("Display"). After completion, "index.css" served to properly move the sections made in "App.js" into their proper place. Cleaning up their size, color, and orientation. Ultimately using CSS and HTML to design.

[1] Current weather data - openweathermap. (n.d.-a).

<https://openweathermap.org/current>

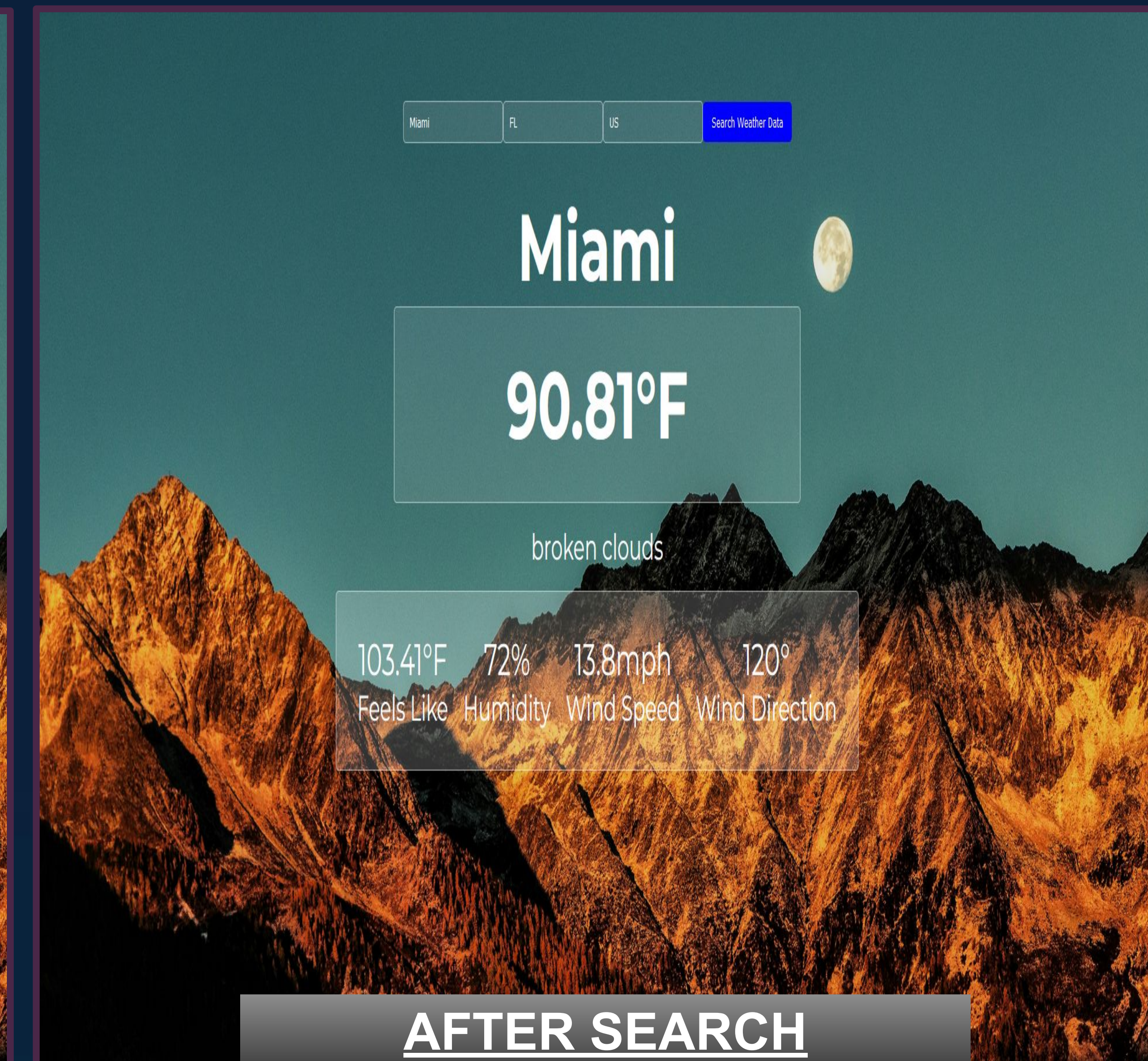
[2] MozDevNet. (n.d.). *Web technology for developers: MDN*. MDN Web

Docs. <https://developer.mozilla.org/en-US/docs/Web>

PROJECT DEMOS / CODE



BEFORE SEARCH



AFTER SEARCH

```
src > # index.css > %p2
1 @import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@40,100..900;1,100..900&display=swap');
2
3 * {
4   box-sizing: border-box;
5   margin: 0;
6   padding: 0;
7 }
8
9 body {
10  margin: 0;
11  font-family: 'Montserrat', sans-serif;
12  -webkit-font-smoothing: antialiased;
13  text-orientation: upright;
14  max-overflow: scroll;
15 }
16
17 p {
18  display: flex;
19  font-size: 1.2em;
20  align-items: center;
21  justify-content: center;
22 }
23
24 p2 {
25  display: flex;
26  font-size: 1.2em;
27  align-items: center;
28  justify-content: center;
29 }
30
31 h1 {
32  font-size: 5.3em;
33 }
34
35 .app {
36  display: flex;
37  flex-direction: column;
38  align-items: center;
39  justify-content: center;
40  height: 100%;
41  background-color: #f0f0f0;
42  background-size: cover;
43  background-position: center;
44  color: #000;
45 }
46
47 .search {
48  margin-bottom: 20px;
49  margin-top: 20px;
50 }
51
52 .search button {
53  padding: 10px 15px;
54  background-color: #000;
55  color: #fff;
56  border: none;
57  border-radius: 5px;
58  cursor: pointer;
59  margin-top: 20px;
60 }
```

Formatting

```
src > # index.css > %p2
1 .search {
2   margin-bottom: 10px;
3   padding: 10px;
4   max-width: 300px;
5   color: #fff;
6   background-color: #000;
7   border: 1px solid #000;
8   border-radius: 5px;
9 }
10
11 .search input {
12  width: 100%;
13  margin: 0;
14  padding: 5px;
15  color: #fff;
16  text-align: center;
17 }
18
19 .location {
20  max-width: 300px;
21  width: 100%;
22  margin: 0 auto 10px auto;
23  padding: 5px;
24  color: #fff;
25  border: 1px solid #000;
26  display: flex;
27  justify-content: center;
28 }
29
30 .containment1 {
31  max-width: 300px;
32  width: 100%;
33  margin: 0 auto 10px auto;
34  padding: 5px;
35  color: #fff;
36  background-color: #000;
37  border: 1px solid #000;
38  display: flex;
39  justify-content: center;
40 }
41
42 .containment2 {
43  max-width: 300px;
44  width: 100%;
45  margin: 0 auto 10px auto;
46  padding: 5px;
47  color: #fff;
48  background-color: #000;
49  border: 1px solid #000;
50  display: flex;
51  justify-content: center;
52 }
53
54 .description {
55  margin-left: 0 20px;
56  padding: 10px;
57 }
58
59 .top, .bottom {
60  display: flex;
61  justify-content: space-between;
62  padding: 20px;
63 }
```

```
src > # App.js > ...
1 function App() {
2   <div className="app">
3     <div className="search">
4       <form onSubmit={getWeather}>
5         <input
6           type="text"
7           value={city}
8           onChange={(e) => setCity(e.target.value)}
9           placeholder="City Name"
10        />
11       <input
12         type="text"
13         value={state}
14         onChange={(e) => setState(e.target.value)}
15         placeholder="State Code (optional, e.g., CA)"
16       />
17       <input
18         type="text"
19         value={country}
20         onChange={(e) => setCountry(e.target.value)}
21         placeholder="Country Code (e.g., US)"
22       />
23       <button type="submit">Search Weather Data</button>
24     </form>
25   </div>
26   <div className="location">
27     <div>{weatherInfo ? weatherInfo.name : ""}</div>
28   </div>
29   <div className="containment1">
30     <div className="top">
31       <div className="temperature">
32         <div>{weatherInfo ? `${weatherInfo.main.temp}°F` : "Choose a City"}</div>
33       </div>
34     </div>
35     <div className="description">
36       <div>{weatherInfo ? weatherInfo.weather[0].description : ""}</div>
37     </div>
38     <div className="containment2">
39       <div className="bottom">
40         <div className="feels-like">
41           <p>{weatherInfo ? `${weatherInfo.main.feels_like}°F` : ""}</p>
42         </div>
43         <div className="humidity">
44           <p>{weatherInfo ? `${weatherInfo.main.humidity}%` : ""}</p>
45         </div>
46         <div className="wind">
47           <p>{weatherInfo ? `${weatherInfo.wind.speed}mph` : ""}</p>
48         </div>
49         <div className="wind-degree">
50           <p>{weatherInfo ? `${weatherInfo.wind.deg}°` : ""}</p>
51         </div>
52       </div>
53     </div>
54   </div>
55 }
```

Display



Engineering
& Computing

ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by XXXXX (name of the project sponsor: federal, state or local government, or corporate partners, where applicable). We/I thank XXXX for his/her/their assistance and mentorship that I/we received throughout the senior design project.

FLORIDA INTERNATIONAL UNIVERSITY