

```
1 <!-- Import Bootstrap Library -->
2 <link rel="stylesheet"
  href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.cs
  s"
3   integrity="sha384-
  gg0yR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQU0hcWr7x9JvoRxt2MZw1T"
  crossorigin="anonymous">
4 <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
5   integrity="sha384-
  q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
  crossorigin="anonymous"></script>
7 <script
  src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.j
  s"
8   integrity="sha384-
  U02eT0CpHqdSJQ6hJty5KVphtPhzWj9W01cLHTMga3JJDZwrnQq4sF86dIHNDz0W1"
  crossorigin="anonymous"></script>
9
10 <script
  src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
11   integrity="sha384-
  JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
12   crossorigin="anonymous"></script>
13
14 <!-- Import Vue.js distribution -->
15 <script src="https://npmcdn.com/vue/dist/vue.js"></script>
16
17 <!-- HTML code using Vue.js components -->
18 <html>
19   <head>
20     <title>CSU44000 - Weather Predictor</title>
21   </head>
22   <div id="app" class="ml-2">
23     <h1>Welcome to the Magic Weather Predictor</h1>
24
25     Please enter the town you are seeking the forecast of:
26
27     <span v-if="isError">
28       <br />
29       <br />
30       <font color="red">{{errorMessage}}</font>
31     </span>
32
33     <br />
34     <br />
35
36     <!-- Town Text Input -->
37     Town: <input v-model="town">
38
39     <br />
40     <br />
41
42     <button v-on:click="getWeatherForTown">Show me the weather!</button>
43
44     <br />
45     <br />
46
47     <!-- Rain Packing Tips -->
48     <div v-if="!isError && isRain != null">
49       <p>
50         <b>Rain:</b>
```

```

51         <span v-if="isRain">There is rain forecast during these
dates. You should pack an umberella.</span>
52         <span v-if="!isRain">There is no rain forecast during these
dates. No need to pack an umberella!.</span>
53     </p>
54 </div>
55
56 <!-- Weather Sentiment Packing Tips -->
57 <div v-if="!isError && temperatureSummary != null">
58     <p>
59         <b>Packing:</b>
60         <span v-if="temperatureSummary.sentiment ==
'hot'">Temperature range is >20°C. You should pack light clothes
61             as to keep cool.</span>
62         <span v-if="temperatureSummary.sentiment ==
'warm'">Temperature range is between
63             {{temperatureSummary.max}}°C and
{{temperatureSummary.min}}°C. You should pack both light clothes with
64             some extra layers for when it gets colder.</span>
65         <span v-if="temperatureSummary.sentiment ==
'cold'">Temperature range is between
66             {{temperatureSummary.max}}°C and
{{temperatureSummary.min}}°C. You should pack warm clothes with plenty
67             of insulation to keep warm.</span>
68     </p>
69 </div>
70
71 <table v-if="forecastJson" class="table table-dark w-75">
72     <thead>
73         <tr>
74             <th scope="col">Date</th>
75             <th scope="col">Avg Temperature (°C)</th>
76             <th scope="col">Avg Wind Speed (m/s)</th>
77             <th scope="col">Rainfall Level (mm)</th>
78             <th scope="col">Min Temperature (°C)</th>
79             <th scope="col">Max Temperature (°C)</th>
80         </tr>
81     </thead>
82     <tbody>
83         <template v-for="(dateEntry, date) in forecastJson">
84             <tr>
85                 <th scope="row">{{ date }}</th>
86                 <td>{{ dateEntry.averageTemp }}</td>
87                 <td>{{ dateEntry.averageWind }}</td>
88                 <td>{{ dateEntry.rainfallLevels }}</td>
89                 <td>{{ dateEntry.temperatureRange.min }}</td>
90                 <td>{{ dateEntry.temperatureRange.max }}</td>
91             </tr>
92         </template>
93     </tbody>
94 </table>
95 </div>
96 </html>
97
98 <!-- Vue.js Components -->
99 <script>
100     var app = new Vue({
101         el: '#app',
102         data: {
103             town: '',

```

```
104         isError: false,
105         errorMessage: '',
106         forecastJson: null,
107         isRain: null,
108         temperatureSummary: null,
109         townSearchHistory: []
110     },
111     methods: {
112         getWeatherForTown: getWeatherForTown
113     }
114 });
115
116 function getWeatherForTown() {
117     console.log(`Getting weather for town ${this.town}...`);
118
119     fetch(`http://localhost:3000/forecast/${this.town}`)
120         .then(response => {
121             // First check that there are no errors
122             if (!response.ok)
123                 throw Error(response.statusText);
124
125             // If all good, return the json
126             return response.json();
127         })
128         .then(responseJSON => {
129             console.log(responseJSON);
130             this.forecastJson = responseJSON.forecastSummary;
131             this.temperatureSummary = responseJSON.temperatureSummary;
132             this.isRain = responseJSON.isRain;
133
134             // Reset any errors
135             this.isError = false;
136             this.errorMessage = '';
137         })
138         .catch(error => {
139             console.error(error);
140
141             // Set error messages
142             this.isError = true;
143             this.errorMessage = `Unable to fetch weather data for
144             ${this.town}`;
145         });
146     }
147 }
148 </script>
```