CSCI 3230: Web Application Development
Final Project - Additional Topics

In order to implement the ideas we had for our project, we needed to use a few tools that required further research. In order to keep our application dynamic we chose to use React.JS. React uses a component-based architecture, which allows us to break our project down into smaller components. This was beneficial for our team, so everyone could work on sections of the project without disrupting others while still keeping everything organized. Another benefit to using React is that it utilizes a virtual DOM. This allows for faster and more efficient rendering of elements to the actual DOM.

When building the web application we decided to use the Open Meteo weather api, as it allowed us to get a variety of different weather variables of a specific location using just latitude and longitude coordinates. To obtain the latitude and longitude coordinates we decided to use the MapBox's Temporary Geocoding API. MapBox is a platform that provides mapping and location service. This was very useful for our project as it allowed us to do geolocation and geocoding.

We then created a drawTemperatureChart function to display a temperature chart for a given location using D3.js. We extracted the hourly temperature data for the next 24 hours and converted it to the chosen unit. We defined the chart's dimensions and margins and created linear scales for the x-axis (time in hours) and the y-axis (temperature), with their domains set to the range of hours and temperatures. We defined the line generator function to create the line path for the chart using input data, then removed existing elements in the temperature chart container. We then appended an SVG element with the appropriate dimensions and margins to the container, and drew the line using the line generator. An animation effect was applied to the line, gradually revealing it using a linear transition. We appended axis labels and a chart title to the SVG, and created a tooltip element to display the temperature value and corresponding time when hovering over the line. We added a circle and a rectangle element to handle hover events, revealing the circle and tooltip when the pointer was near the line and hiding them otherwise. Lastly, we added a 'resize' event listener for the window, ensuring the chart would be redrawn to maintain responsiveness when the window was resized.

SQLite is a lightweight, embedded relational database management system that is widely used in various types of software applications due to its ease of integration and low memory usage. In general, SQLite is often used in small-scale applications where a full database system might be overkill. SQLite was used in our weather app to store city names recently searched for by the user. This allowed users to quickly retrieve weather data for cities they have already searched for, without having to search for them again. SQLite was an ideal choice for this project because it is best suited for smaller applications and can be easily integrated into web-based applications.

Axios is a promise-based HTTP client for the browser and node.js. Axios encapsulates Ajax and provides a simple interface to make and handle http requests. Axios was very helpful as it allowed us to make http requests from node.js. This was mainly used for asynchronous calls from our backend server.