MINI PROJECT REPORT ON

WEATHER APP WEBPAGE

(CSE IV SEM MINI PROJECT)

2022-2023

Submitted by:

Name : Ayush chaudhary

Sec: J

Roll No : 12

University Roll No: 2118388

Student ID: 21011526

Submitted to:

Ms. Manika manwal

CONTENT

INTRODUCTION

The main objective of this project is to show a responsive weather info. Website.

as i just entered in the field of web development I wanted to makea mini project which look Simple,attractive, working. So I created a responsive website.

In previouss emester I used the same project but this time I make this project using react.

Called weather information application ,as of now I learned only frontend development and have gained small knowledge on backend development.

Language and extension used-

1. HTML – Hyper text markup language
2. CSS – cascading style sheets
3. JS – javascript
4. React.
5. EXTENSION: rapid api(navbar)
6. VSCODE
7. liveserver extension

SYSTEM REQUIREMENTS :-

On client side:

1. operating system(windows,mac-os,Linux,etc)

HARDWARE REQUIREMENTS:

1. i3 or above processor
2. 1 GB RAM(minimum)
3. 256 GB SSD
4. INTERNET connection.

FEATURES:

1. Responsive
2. Get weather updates in Real Time
3. user friendly
4. free of cost.
5. no need for deep dive in coding
6. consists api which makes your code more

easier7.Secure connection

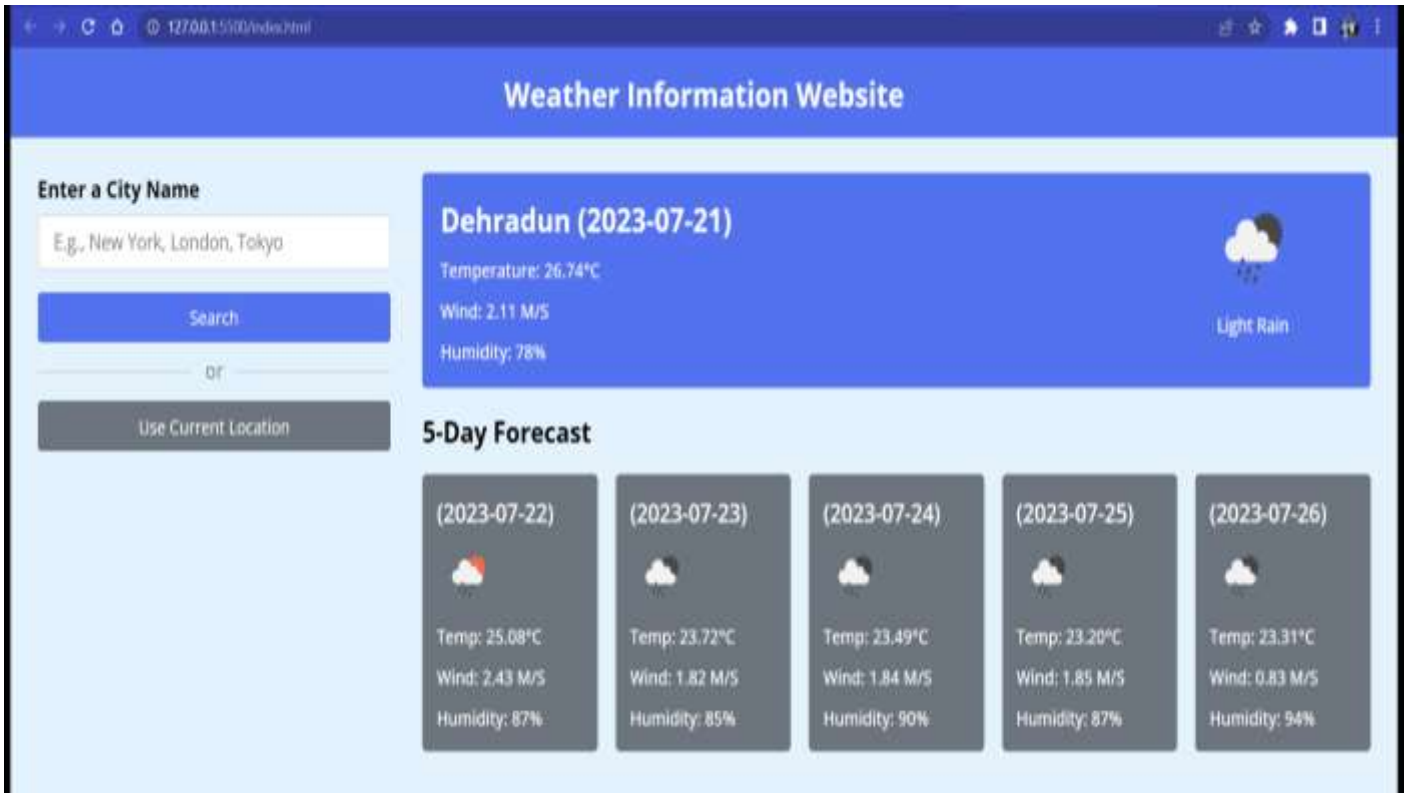
8.Simple UI.

API(APPLICATION PROGRAMMING INTERFACE)

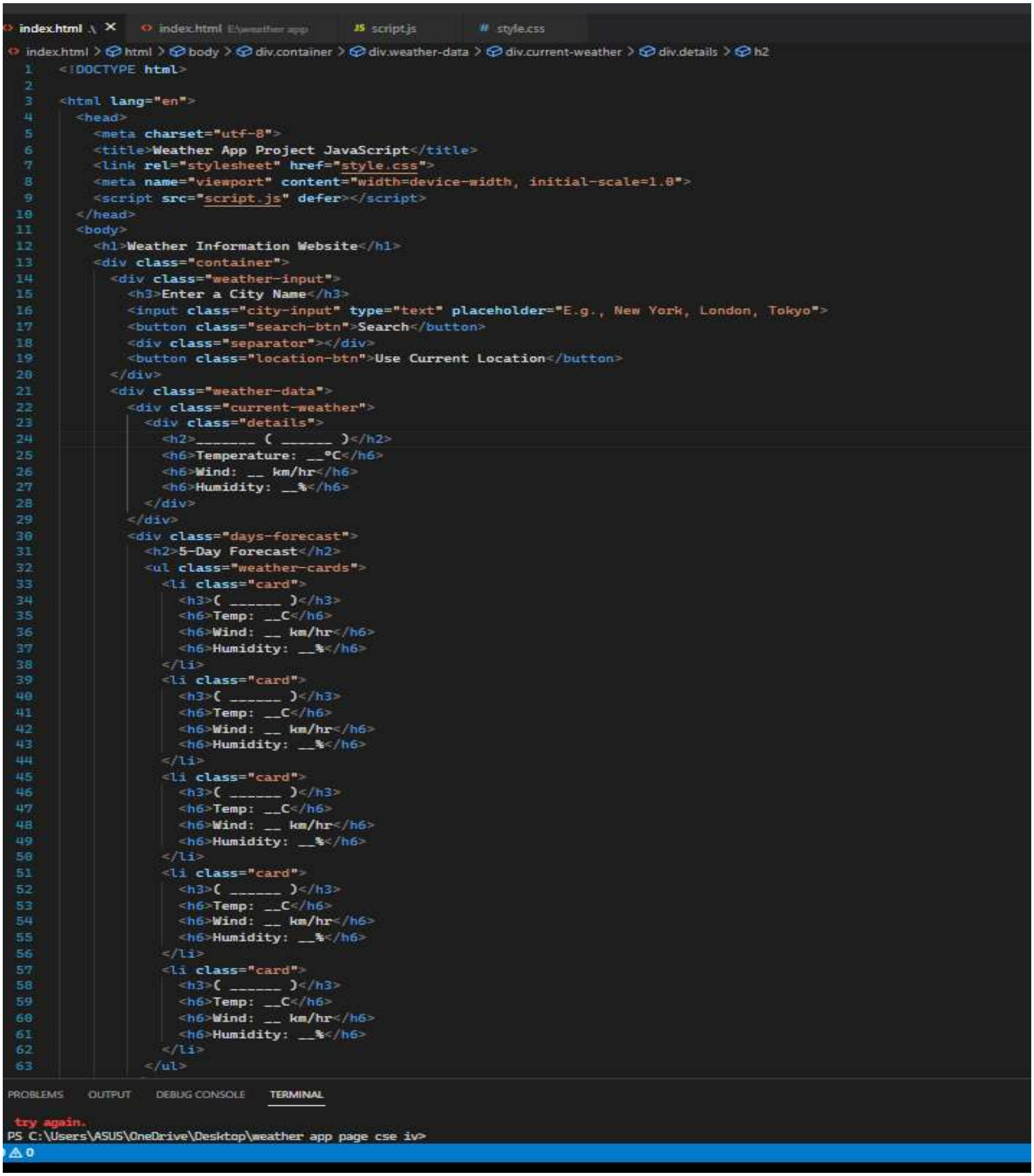
An API is an interface that software developers use to programmatically interact with software components or resources outside of their own code. An even simpler definition is that an API is the part of a software component that is accessible to other components.

Unless you write every line of code from scratch, you will interact with external software components, and each of these will have its own API. Even if you do write all of your code from scratch, a well-designed application should have internal APIs to help organize the code and make its components more reusable.

SCREENSHOTS OF WEATHER INFO WEBSITE



SOURCE CODE(HTML)



```html
<!DOCTYPE html>

<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Weather App Project JavaScript</title>
    <link rel="stylesheet" href="style.css">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <script src="script.js" defer></script>
  </head>
  <body>
    <h1>Weather Information Website</h1>
    <div class="container">
      <div class="weather-input">
        <h3>Enter a City Name</h3>
        <input class="city-input" type="text" placeholder="E.g., New York, London, Tokyo">
        <button class="search-btn">Search</button>
        <div class="separator"></div>
        <button class="location-btn">Use Current Location</button>
      </div>
      <div class="weather-data">
        <div class="current-weather">
          <div class="details">
            <h2>_______ ( ______ )</h2>
            <h6>Temperature: __°C</h6>
            <h6>Wind: __ km/hr</h6>
            <h6>Humidity: __%</h6>
          </div>
        </div>
        <div class="days-forecast">
          <h2>5-Day Forecast</h2>
          <ul class="weather-cards">
            <li class="card">
              <h3>( ______ )</h3>
              <h6>Temp: __C</h6>
              <h6>Wind: __ km/hr</h6>
              <h6>Humidity: __%</h6>
            </li>
            <li class="card">
              <h3>( ______ )</h3>
              <h6>Temp: __C</h6>
              <h6>Wind: __ km/hr</h6>
              <h6>Humidity: __%</h6>
            </li>
            <li class="card">
              <h3>( ______ )</h3>
              <h6>Temp: __C</h6>
              <h6>Wind: __ km/hr</h6>
              <h6>Humidity: __%</h6>
            </li>
            <li class="card">
              <h3>( ______ )</h3>
              <h6>Temp: __C</h6>
              <h6>Wind: __ km/hr</h6>
              <h6>Humidity: __%</h6>
            </li>
            <li class="card">
              <h3>( ______ )</h3>
              <h6>Temp: __C</h6>
              <h6>Wind: __ km/hr</h6>
              <h6>Humidity: __%</h6>
            </li>
          </ul>
```

```
 try again.
PS C:\Users\ASUS\OneDrive\Desktop\weather app page cse iv>
```

SOURCE CODE(CSS)

style.css > .weather-input input

```
/* Import Google font - Open Sans */
@import url('https://fonts.googleapis.com/css2?family=Open+Sans:wght@400;500;600;700&display=swap');
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: 'Open Sans', sans-serif;
}
body {
  background: #E3F2FD;
}
h1 {
  background: #5372F0;
  font-size: 1.75rem;
  text-align: center;
  padding: 18px 0;
  color: #fff;
}
.container {
  display: flex;
  gap: 35px;
  padding: 30px;
}
.weather-input {
  width: 550px;
}
.weather-input input {
  height: 46px;
  width: 100%;
  outline: none;
  font-size: 1.07rem;
  padding: 0 17px;
  margin: 10px 0 20px 0;
  border-radius: 4px;
  border: 1px solid #ccc;
}
.weather-input input:focus {
  padding: 0 16px;
  border: 2px solid #5372F0;
}
.weather-input .separator {
  height: 1px;
  width: 100%;
  margin: 25px 0;
  background: #BBBBBB;
  display: flex;
  align-items: center;
  justify-content: center;
}
.weather-input .separator::before{
  content: "or";
  color: #6C757D;
  font-size: 1.18rem;
  padding: 0 15px;
  margin-top: -4px;
  background: #E3F2FD;
}
.weather-input button {
  width: 100%;
  padding: 10px 0;
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
            <th scope="row" class="text-start">Extra security</th>
             <td></td>
            <td></td>
            <td><svg class="bi" width="24" height="24"><use xlink:href="#check"></use></svg></td>
           </tr>
          </tbody>
         </table>
        </div>
      </main>
    </div>
```

SOURCE CODE(JS)

index.html | index.html E:\weather app | script.js | style.css

script.js > createWeatherCard

```javascript
const cityInput = document.querySelector(".city-input");
const searchButton = document.querySelector(".search-btn");
const locationButton = document.querySelector(".location-btn");
const currentWeatherDiv = document.querySelector(".current-weather");
const weatherCardsDiv = document.querySelector(".weather-cards");

const API_KEY = "2f26c2b38834338306396180 10b468e6"; // API key for OpenWeatherMap API

const createWeatherCard = (cityName, weatherItem, index) => {
    if(index === 0) { // HTML for the main weather card
        return '<div class="details">
                    <h2>${cityName} (${weatherItem.dt_txt.split(" ")[0]})</h2>
                    <h6>Temperature: ${(weatherItem.main.temp - 273.15).toFixed(2)}°C</h6>
                    <h6>Wind: ${weatherItem.wind.speed} M/S</h6>
                    <h6>Humidity: ${weatherItem.main.humidity}%</h6>
                </div>
                <div class="icon">
                    <img src="https://openweathermap.org/img/wn/${weatherItem.weather[0].icon}@4x.png" alt="weather-icon">
                    <h6>${weatherItem.weather[0].description}</h6>
                </div>';
    } else { // HTML for the other five day forecast card
        return '<li class="card">
                    <h3>(${weatherItem.dt_txt.split(" ")[0]})</h3>
                    <img src="https://openweathermap.org/img/wn/${weatherItem.weather[0].icon}@4x.png" alt="weather-icon">
                    <h6>Temp: ${(weatherItem.main.temp - 273.15).toFixed(2)}°C</h6>
                    <h6>Wind: ${weatherItem.wind.speed} M/S</h6>
                    <h6>Humidity: ${weatherItem.main.humidity}%</h6>
                </li>';
    }
}

const getWeatherDetails = (cityName, latitude, longitude) => {
    const WEATHER_API_URL = `https://api.openweathermap.org/data/2.5/forecast?lat=${latitude}&lon=${longitude}&appid=${API_KEY}`;

    fetch(WEATHER_API_URL).then(response => response.json()).then(data => {
        // Filter the forecasts to get only one forecast per day
        const uniqueForecastDays = [];
        const fiveDaysForecast = data.list.filter(forecast => {
            const forecastDate = new Date(forecast.dt_txt).getDate();
            if (!uniqueForecastDays.includes(forecastDate)) {
                return uniqueForecastDays.push(forecastDate);
            }
        });

        // Clearing previous weather data
        cityInput.value = "";
        currentWeatherDiv.innerHTML = "";
        weatherCardsDiv.innerHTML = "";

        // Creating weather cards and adding them to the DOM
        fiveDaysForecast.forEach((weatherItem, index) => {
            const html = createWeatherCard(cityName, weatherItem, index);
            if (index === 0) {
                currentWeatherDiv.insertAdjacentHTML("beforeend", html);
            } else {
                weatherCardsDiv.insertAdjacentHTML("beforeend", html);
            }
        });
    }).catch(() => {
        alert("An error occurred while fetching the weather forecast!");
```

FUTURE SCOPE

See future scope in this field is that we can have enormous opportunities as we can go towards full stack development field.
And through this I can improve this project of weather ̀website into backend responsive website..

REFERENCES

www.codewithharry.com

www.w3schools.com

www.geeksforgeeks.com