

# **PROJECT REPORT**

**On**

## **“WEATHER JOURNAL APP”**

Department of Computer Engineering & Applications  
**GLA UNIVERSITY**



**GLA University**  
**Mathura- 281406, INDIA**  
**2022-2023**

### **SUBMITTED BY:-**

**Ayush Sharma (201500183)**  
**Param Varshney (201500460)**  
**Vaibhav Varshney (201500767)**

### **SUBMITTED TO:-**

**Mr. Bhanu Kapoor**  
**(Technical Trainer)**

## **Declaration**

We hereby declare that the work which is being presented in the Project of “**Weather Journal App**”, in partial fulfilment of the requirements for Project viva voce, is an authentic record of our own work carried by the team members under the supervision of our mentor Mr. Bhanu Kapoor. Group Members:

**Ayush Sharma**           **(201500183)**

**Param Varshney**       **(201500460)**

**Vaibhav Varshney**   **(201500767)**

Course: B.Tech (Computer Science and Engineering)

Year: 3<sup>rd</sup>

Semester: 5<sup>th</sup>

Supervised By

Mr. Bhanu Kapoor, Technical Trainer,

GLA University, Department of Computer Engineering & Application



**Department of computer Engineering and Applications**  
**GLA University, Mathura**

**17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,  
Mathura – 281406**

---

**Certificate**

This is to certify that the above statements made by the candidates are correct to the best of my/our knowledge and belief.

\_\_\_\_\_ Supervisor

Mr. Bhanu Kapoor

Technical Trainer

Dept of CEA, GLA University

\_\_\_\_\_  
\_\_\_\_\_

Project Mentor

(Mr. Bhanu Kapoor )

Program Coordinator

(Mr ShashShekhar)

# About the Project

Weather Report project application is a web based application through which you will be able to get all the reports related to weather forecasting of any locations. Its geographical locator which will be received through your browser setting and server configuration will automatically identify the location and be able to present its weather details such as temperature, direction of wind, rains, humidity etc. To change the location you will just have to select the options provided below to get its details.

## **Motivation**

The emergence of modern technologies has had profound impacts of the weather, with online journal now an integral part of the knowing weather. The main advantages of weather journal are flexibility and accessibility of the weather. User can access to website to assist themselves longer upcoming weather to the hours of operation of various forecasts and reports, and can be provided anytime and anywhere. It is a well-established, and effective, instructional method. However, there is a need for more empirical research to be directed toward investigating weather experiences with online services, their impact on nature and human well-being, and various places. The purpose of this project is to develop a frontend platform for weather journaling

# Requirements

## **a). Software Requirements:**

- Languages/Technologies Used: HTML, CSS, JavaScript
- IDE Used: Vs code
- Web Browser: Google Chrome
- GitHub: GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere. GitHub Repository: A GitHub repository can be used to store a development project. It can contain folders and any type of files (HTML, CSS, JavaScript, Documents, Data, Images). A GitHub repository should also include a license file and a README file about the project. A GitHub repository can also be used to store ideas, or any resources that you want to share.

## **b). Hardware Requirements:**

- Processor Required: Intel i3
- Operating System: Windows 10
- RAM: 4GB
- Hardware Devices: Computer System
- Hard Disk: 256GB

# **Acknowledgement**

We thank the almighty for giving us the courage and perseverance in completing the project. This project itself is an acknowledgement for all those people who have given us their heartfelt co-operation in making this project a grand success.

We extend our sincere thanks to Mr. Bhanu Kapoor, Technical Trainer at “GLA University, Mathura” for providing his valuable guidance at every stage of this project work. We are profoundly grateful towards the unmatched services rendered by him. And last but not least, we would like to express our deep sense of gratitude and earnest thanks giving to our dear parents for their moral support and heartfelt cooperation in doing the project.

# **Weather Journal App**

## **Abstract**

Weather is the state of the atmosphere at a given place and time in regards to heat, cloudiness, dryness, sunshine, wind, and rain. Of all the geophysical phenomena weather is the most significant one that influences us. Weather can vary greatly and largely depends on climate, seasons and various other factors. The chief goal of this work is to get the weather forecast of any city throughout the world through an application. This paper aims at creating a web application using javascript framework.



# Contents

**Acknowledgment.....**

**Abstract.....**

**1.Introduction -----**

**2.Technologies Used: -----12**

HTML -----**Error! Bookmark not defined.**

CSS -----**Error! Bookmark not defined.**

JavaScript -----

**List Of Figures.....**

**Conclusion.....**

**Bibliography.....**

## **Introduction**

The purpose of this project is to develop a front-end application for weather. It allows for flexible data format and deliver of its data so that each analysis application can receive only the information it needs and in the format required.

The project is divided into 2 modules – user's entries and report coming. The roles of the modules are as follows:

- **Student :**

The user enter various countries available. The user takes a report on a forecast. There might be countries and cities, which has various weather modules. Each entry has multiple choices with only one correct option. The entry will not be time bound. User can see the forecast schedule. New Users will not be required to register themselves in the system. All users will be able to modify their entries. User views previous reports. Users can go to the Generate and browse through city name and pin code and find desired report.

- **Course Expert :**

Creating report for the user, entry will reside in the generate area if either it is saved while creating/modifying or it has been rejected by admin. Modifying entries, deleting the entries browse through the generate link.

# Technologies Used

## HTML

**HTML** stands for **Hyper Text Markup Language**, which is the most widely used language on Web to develop web pages. **HTML** was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995. HTML 4.01 was a major version of HTML and it was published in late 1999. Though HTML 4.01 version is widely used but currently we are having HTML-5 version which is an extension to HTML 4.01, and this version was published in 2012.

It is used for:

- Web development (server-side),
- Software development.

Here are some of the most common uses for HTML:

- **Web pages development** - HTML is used to create pages which are rendered over the web. Almost every page of web is having html tags in it to render its details in browser.
- **Internet Navigation** - HTML provides tags which are used to navigate from one page to another and is heavily used in internet navigation.
- **Responsive UI** - HTML pages now-a-days works well on all platform, mobile, tabs, desktop or laptops owing to responsive design strategy.

- **Offline support** HTML pages once loaded can be made available offline on the machine without any need of internet.
- **Game development**- HTML5 has native support for rich experience and is now useful in gaming development arena as well.

## CSS

CSS is used to control the style of a web document in a simple and easy way.

CSS is the acronym for "**Cascading Style Sheet**". This tutorial covers both the versions CSS1, CSS2 and CSS3, and gives a complete understanding of CSS, starting from its basics to advanced concepts.

## Applications of CSS

- **CSS saves time** - You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.
- **Pages load faster** - If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.
- **Easy maintenance** - To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.

- **Superior styles to HTML** - CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.
- **Multiple Device Compatibility** - Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.
- **Global web standards** - Now HTML attributes are being deprecated and it is being recommended to use CSS. So its a good idea to start using CSS in all the HTML pages to make them compatible to future browsers.

## JavaScript:-

**JavaScript** is a lightweight, interpreted **programming** language. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. **JavaScript** is very easy to implement because it is integrated with HTML. It is open and cross-platform.

JavaScript is the most popular **programming language** in the world and that makes it a programmer's great choice. Once you learnt Javascript, it helps you developing great front-end as

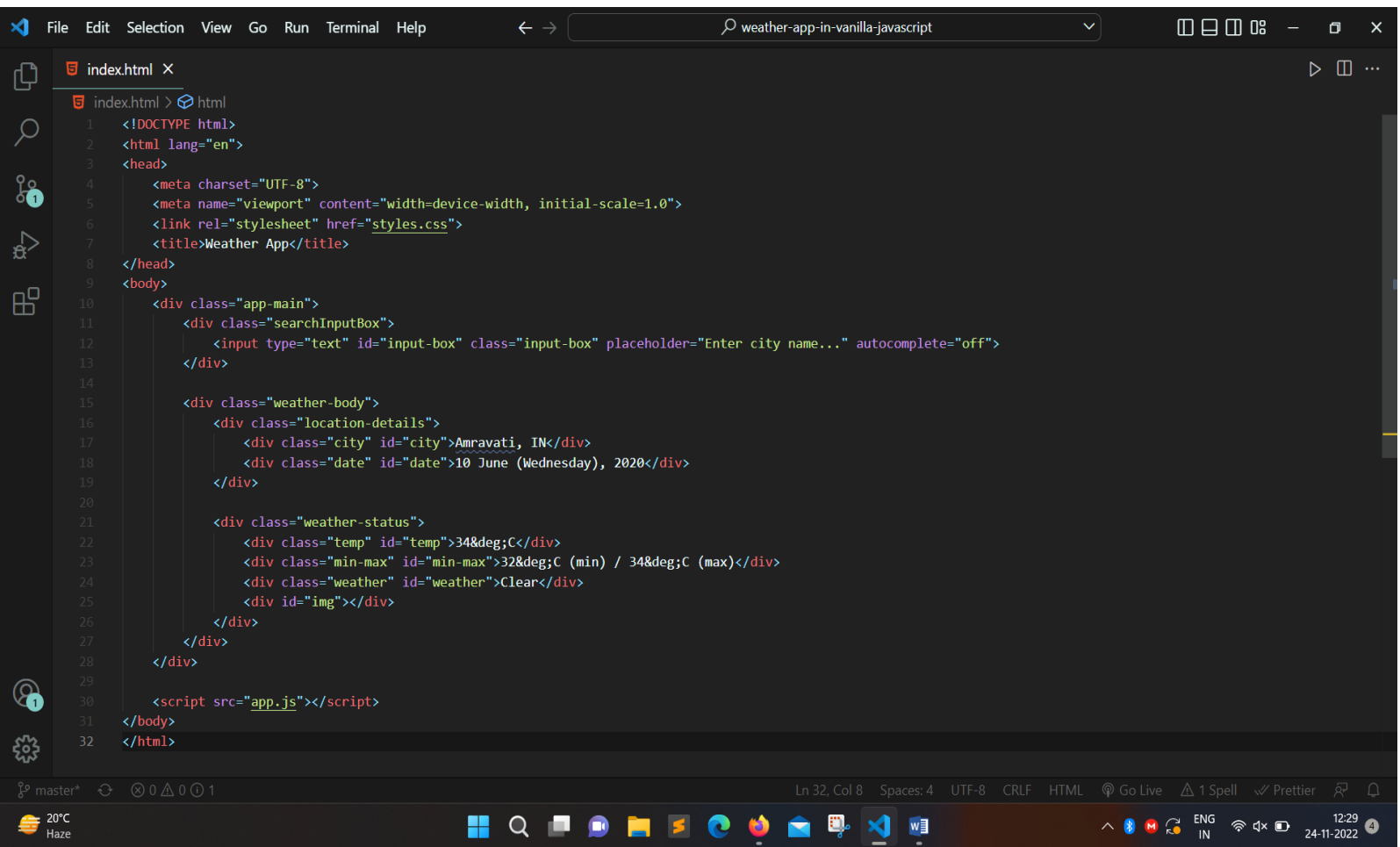
well as back-end softwares using different Javascript based frameworks like jQuery, Node.JS etc

## **Applications of Javascript Programming:-**

- **Client side validation** - This is really important to verify any user input before submitting it to the server and Javascript plays an important role in validating those inputs at front-end itself.
- **Manipulating HTML Pages** - Javascript helps in manipulating HTML page on the fly. This helps in adding and deleting any HTML tag very easily using javascript and modify your HTML to change its look and feel based on different devices and requirements.
- **User Notifications** - You can use Javascript to raise dynamic pop-ups on the webpages to give different types of notifications to your website visitors.
- **Back-end Data Loading** - Javascript provides Ajax library which helps in loading back-end data while you are doing some other processing. This really gives an amazing experience to your website visitors.
- **Presentations** - JavaScript also provides the facility of creating presentations which gives website look and feel. JavaScript provides RevealJS and BespokeJS libraries to build a web-based slide presentations.
- **Server Applications** - Node JS is built on Chrome's Javascript runtime for building fast and scalable network applications. This is an event based library which helps in developing very sophisticated server applications including Web Servers.

# List of Figures

## 1. HTML Code:-

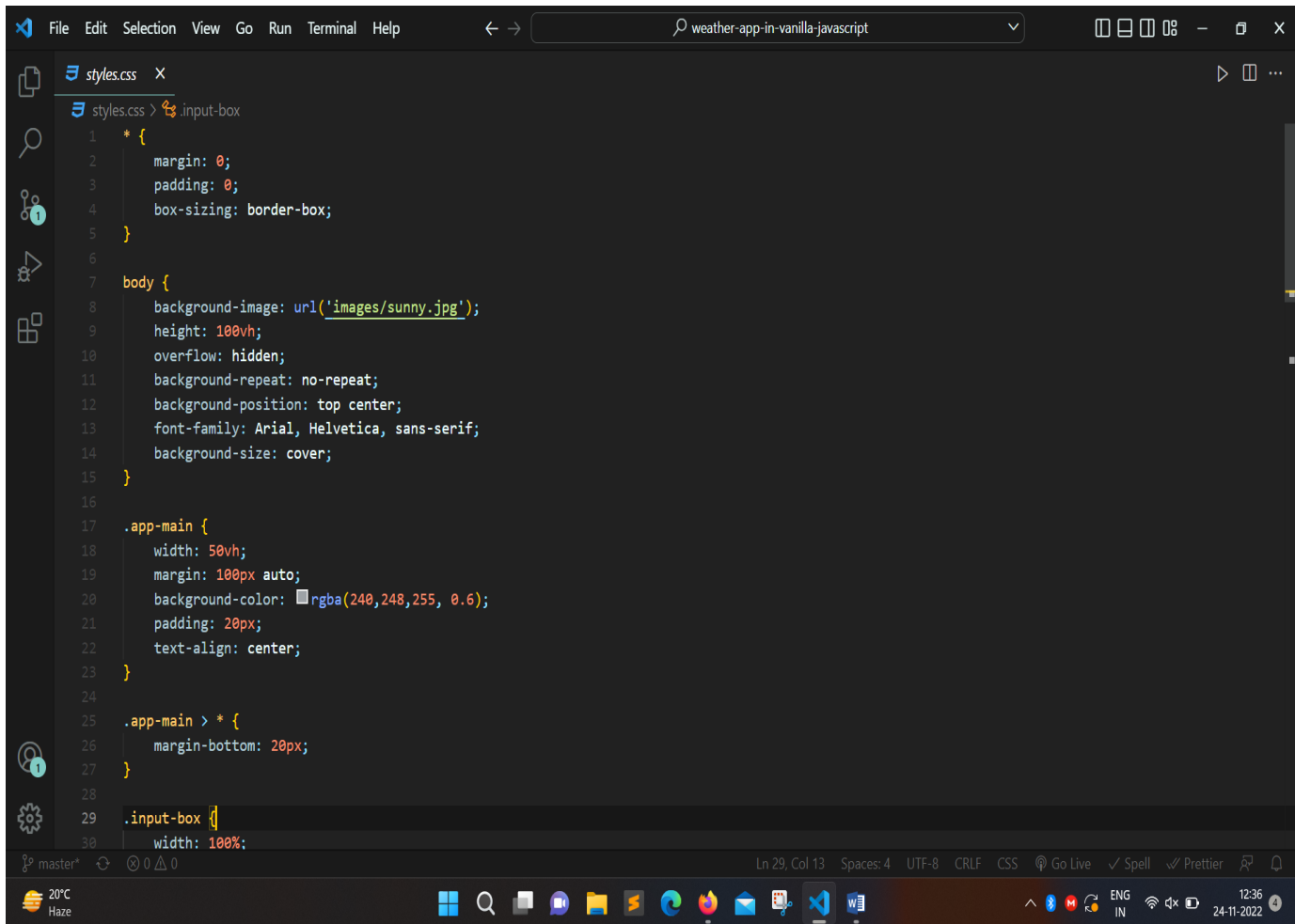


The screenshot shows a code editor with the following HTML code:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <link rel="stylesheet" href="styles.css">
7   <title>Weather App</title>
8 </head>
9 <body>
10  <div class="app-main">
11    <div class="searchInputBox">
12      <input type="text" id="input-box" class="input-box" placeholder="Enter city name..." autocomplete="off">
13    </div>
14
15    <div class="weather-body">
16      <div class="location-details">
17        <div class="city" id="city">Amravati, IN</div>
18        <div class="date" id="date">10 June (Wednesday), 2020</div>
19      </div>
20
21      <div class="weather-status">
22        <div class="temp" id="temp">34&deg;C</div>
23        <div class="min-max" id="min-max">32&deg;C (min) / 34&deg;C (max)</div>
24        <div class="weather" id="weather">Clear</div>
25        <div id="img"></div>
26      </div>
27    </div>
28  </div>
29
30  <script src="app.js"></script>
31 </body>
32 </html>
```

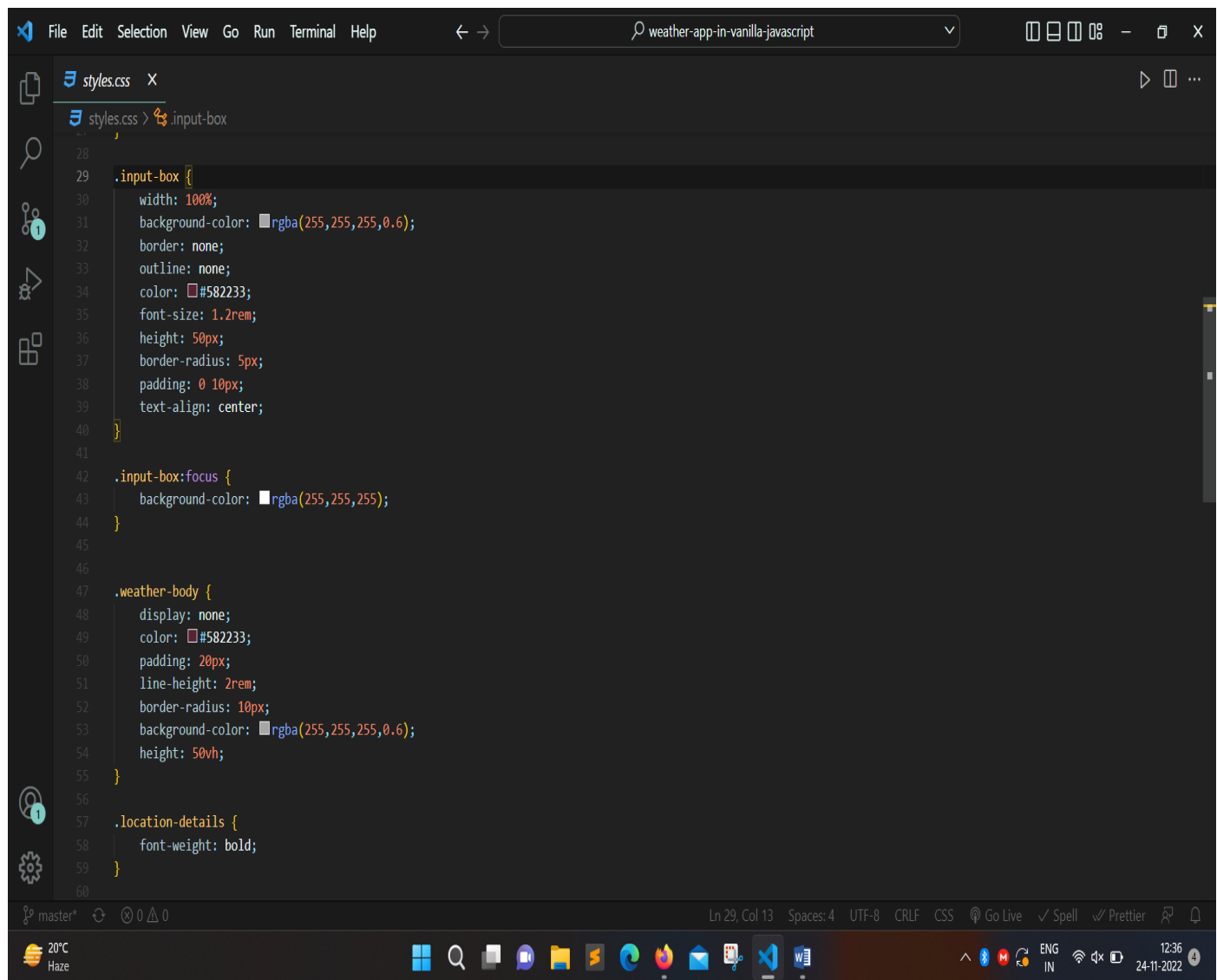


## 2. CSS Code



```
1  * {
2      margin: 0;
3      padding: 0;
4      box-sizing: border-box;
5  }
6
7  body {
8      background-image: url('images/sunny.jpg');
9      height: 100vh;
10     overflow: hidden;
11     background-repeat: no-repeat;
12     background-position: top center;
13     font-family: Arial, Helvetica, sans-serif;
14     background-size: cover;
15 }
16
17 .app-main {
18     width: 50vh;
19     margin: 100px auto;
20     background-color: rgba(240,248,255, 0.6);
21     padding: 20px;
22     text-align: center;
23 }
24
25 .app-main > * {
26     margin-bottom: 20px;
27 }
28
29 .input-box {
30     width: 100%;
```

## CSS Code

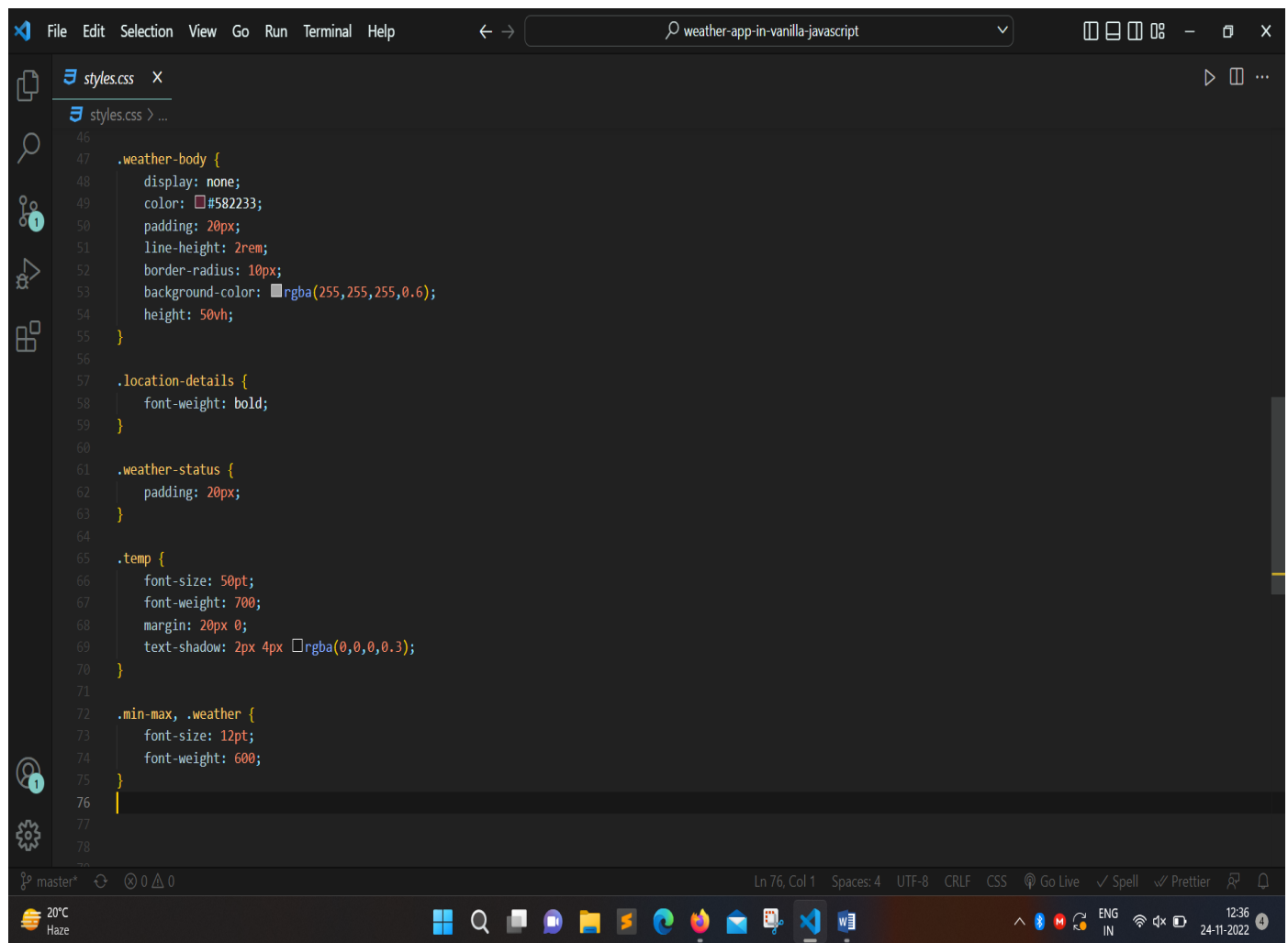


The image shows a screenshot of the Visual Studio Code (VS Code) editor interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The address bar shows the file path 'weather-app-in-vanilla-javascript'. The left sidebar contains icons for Explorer, Search, Source Control, and Run and Debug. The main editor area displays a CSS file named 'styles.css' with the following code:

```
28
29 .input-box {
30     width: 100%;
31     background-color: rgba(255,255,255,0.6);
32     border: none;
33     outline: none;
34     color: #582233;
35     font-size: 1.2rem;
36     height: 50px;
37     border-radius: 5px;
38     padding: 0 10px;
39     text-align: center;
40 }
41
42 .input-box:focus {
43     background-color: rgba(255,255,255);
44 }
45
46
47 .weather-body {
48     display: none;
49     color: #582233;
50     padding: 20px;
51     line-height: 2rem;
52     border-radius: 10px;
53     background-color: rgba(255,255,255,0.6);
54     height: 50vh;
55 }
56
57 .location-details {
58     font-weight: bold;
59 }
60
```

The bottom status bar shows the current file is 'master', the cursor is at line 29, column 13, and the file is encoded in UTF-8 with CRLF line endings. The bottom right corner of the status bar shows the time '12:36' and the date '24-11-2022'.

## CSS code

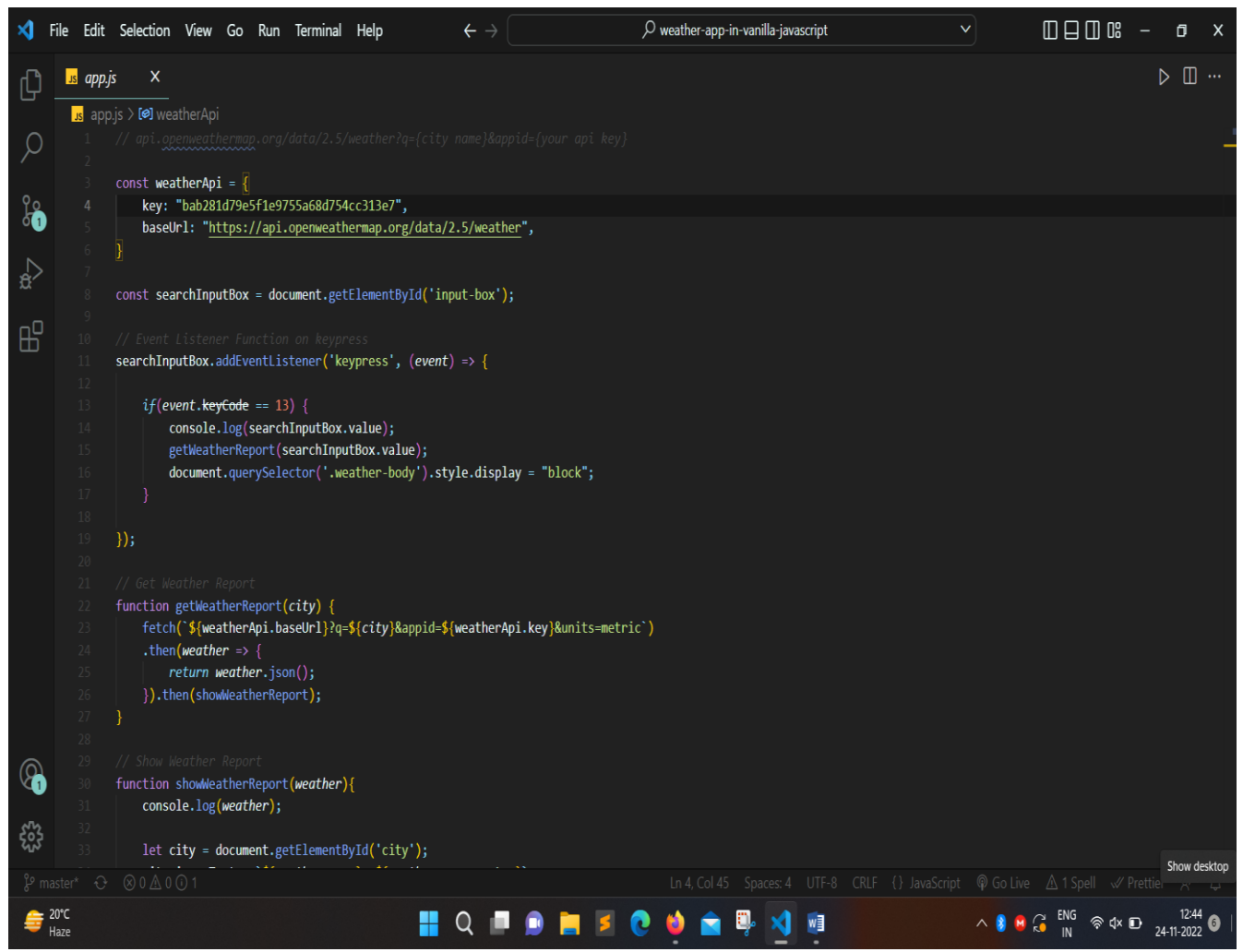


The image shows a screenshot of a Visual Studio Code editor window. The title bar at the top reads "File Edit Selection View Go Run Terminal Help" and "weather-app-in-vanilla-javascript". The editor is open to a file named "styles.css". The code is as follows:

```
46
47 .weather-body {
48     display: none;
49     color: #582233;
50     padding: 20px;
51     line-height: 2rem;
52     border-radius: 10px;
53     background-color: rgba(255,255,255,0.6);
54     height: 50vh;
55 }
56
57 .location-details {
58     font-weight: bold;
59 }
60
61 .weather-status {
62     padding: 20px;
63 }
64
65 .temp {
66     font-size: 50pt;
67     font-weight: 700;
68     margin: 20px 0;
69     text-shadow: 2px 4px rgba(0,0,0,0.3);
70 }
71
72 .min-max, .weather {
73     font-size: 12pt;
74     font-weight: 600;
75 }
76
77
78
```

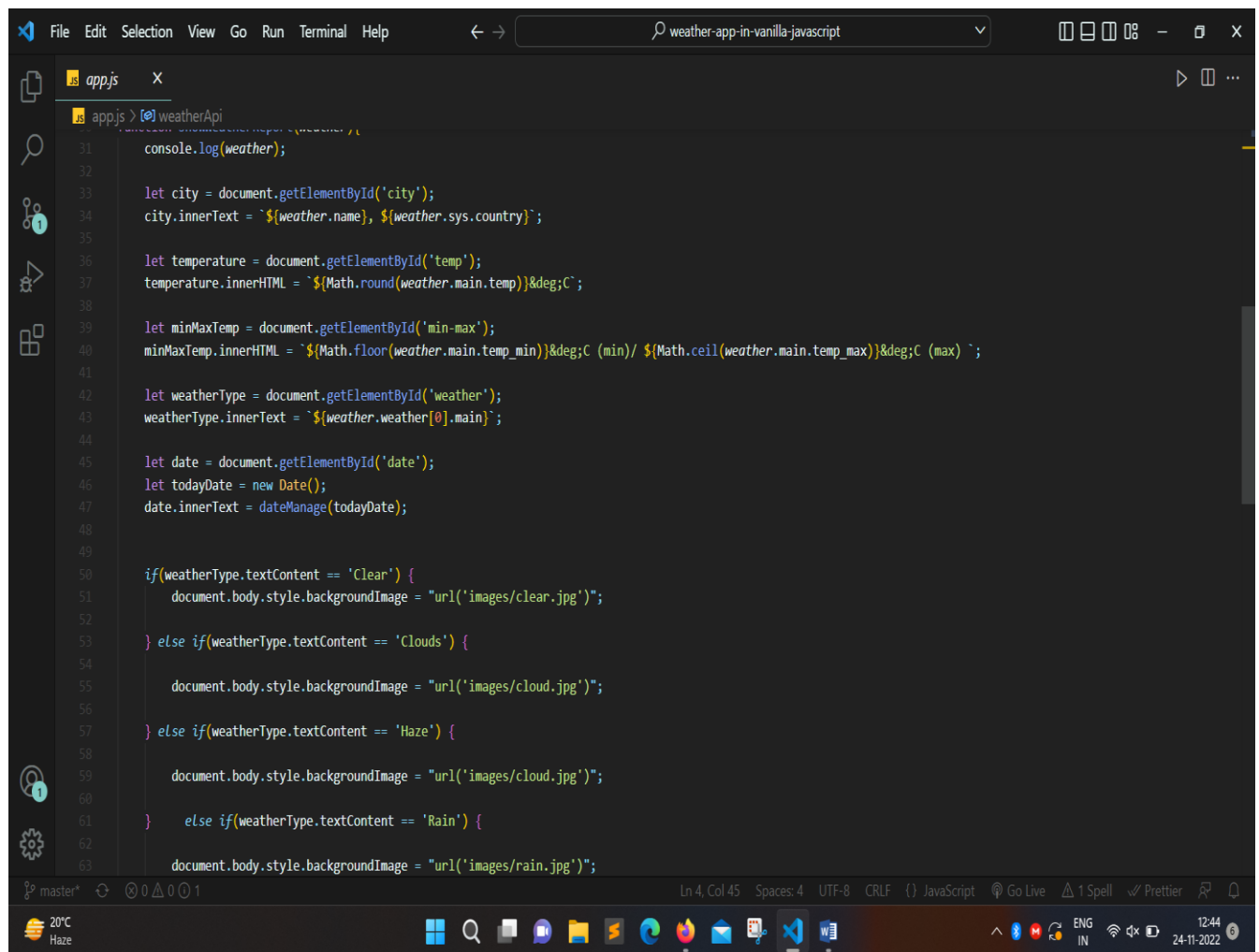
The status bar at the bottom shows "master\*", "Ln 76, Col 1", "Spaces: 4", "UTF-8", "CRLF", "CSS", "Go Live", "Spell", "Prettier", and a system tray with weather information (20°C, Haze) and the date/time (12:36, 24-11-2022).

### 3- JavaScript Code



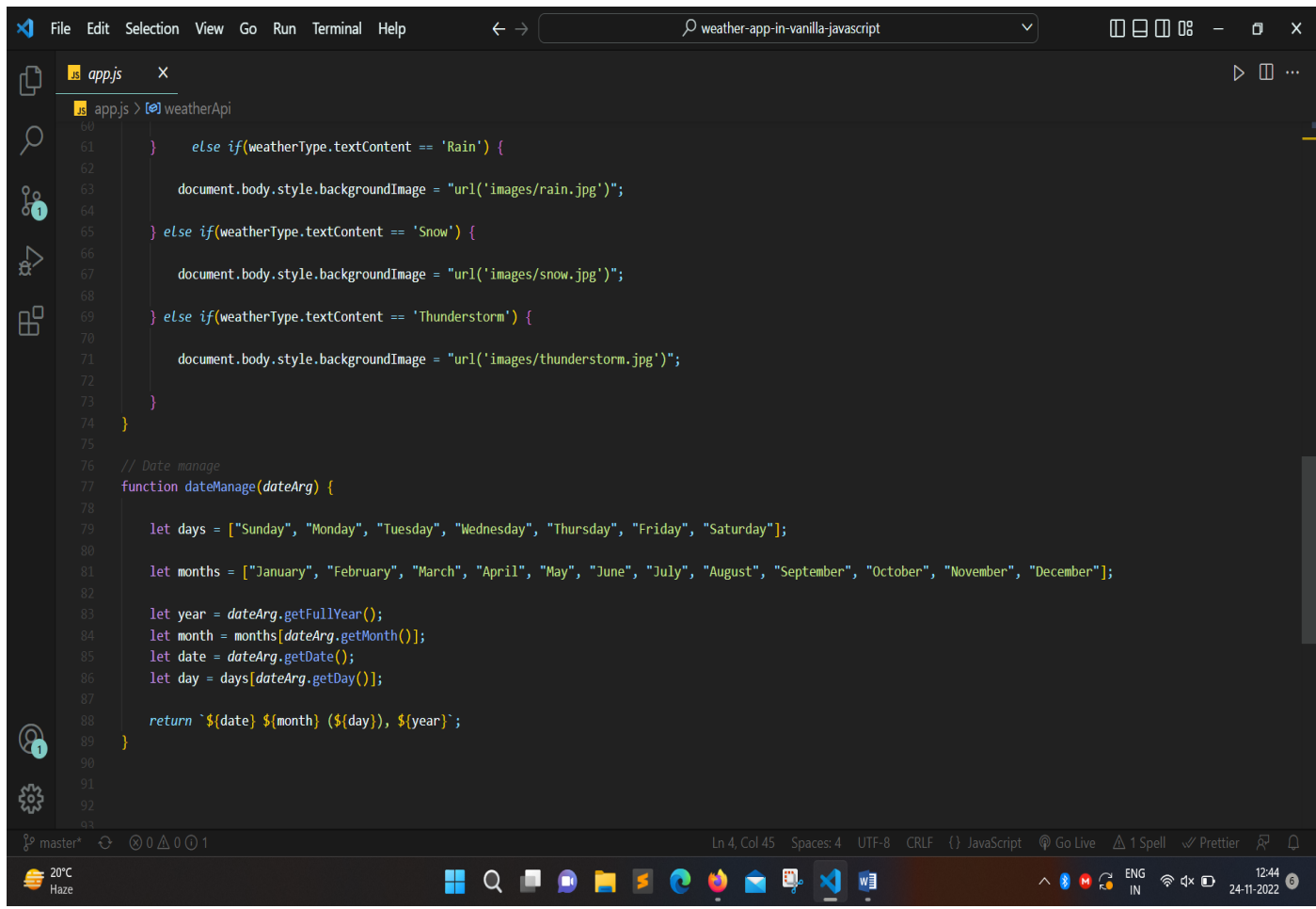
```
1 // api.openweathermap.org/data/2.5/weather?q={city name}&appid={your api key}
2
3 const weatherApi = {
4   key: "bab281d79e5f1e9755a68d754cc313e7",
5   baseUrl: "https://api.openweathermap.org/data/2.5/weather",
6 }
7
8 const searchInputBox = document.getElementById('input-box');
9
10 // Event Listener Function on keypress
11 searchInputBox.addEventListener('keypress', (event) => {
12
13   if(event.keyCode == 13) {
14     console.log(searchInputBox.value);
15     getWeatherReport(searchInputBox.value);
16     document.querySelector('.weather-body').style.display = "block";
17   }
18
19 });
20
21 // Get Weather Report
22 function getWeatherReport(city) {
23   fetch(`${weatherApi.baseUrl}?q=${city}&appid=${weatherApi.key}&units=metric`)
24     .then(weather => {
25       return weather.json();
26     }).then(showWeatherReport);
27 }
28
29 // Show Weather Report
30 function showWeatherReport(weather){
31   console.log(weather);
32
33   let city = document.getElementById('city');
```

# JavaScript Code



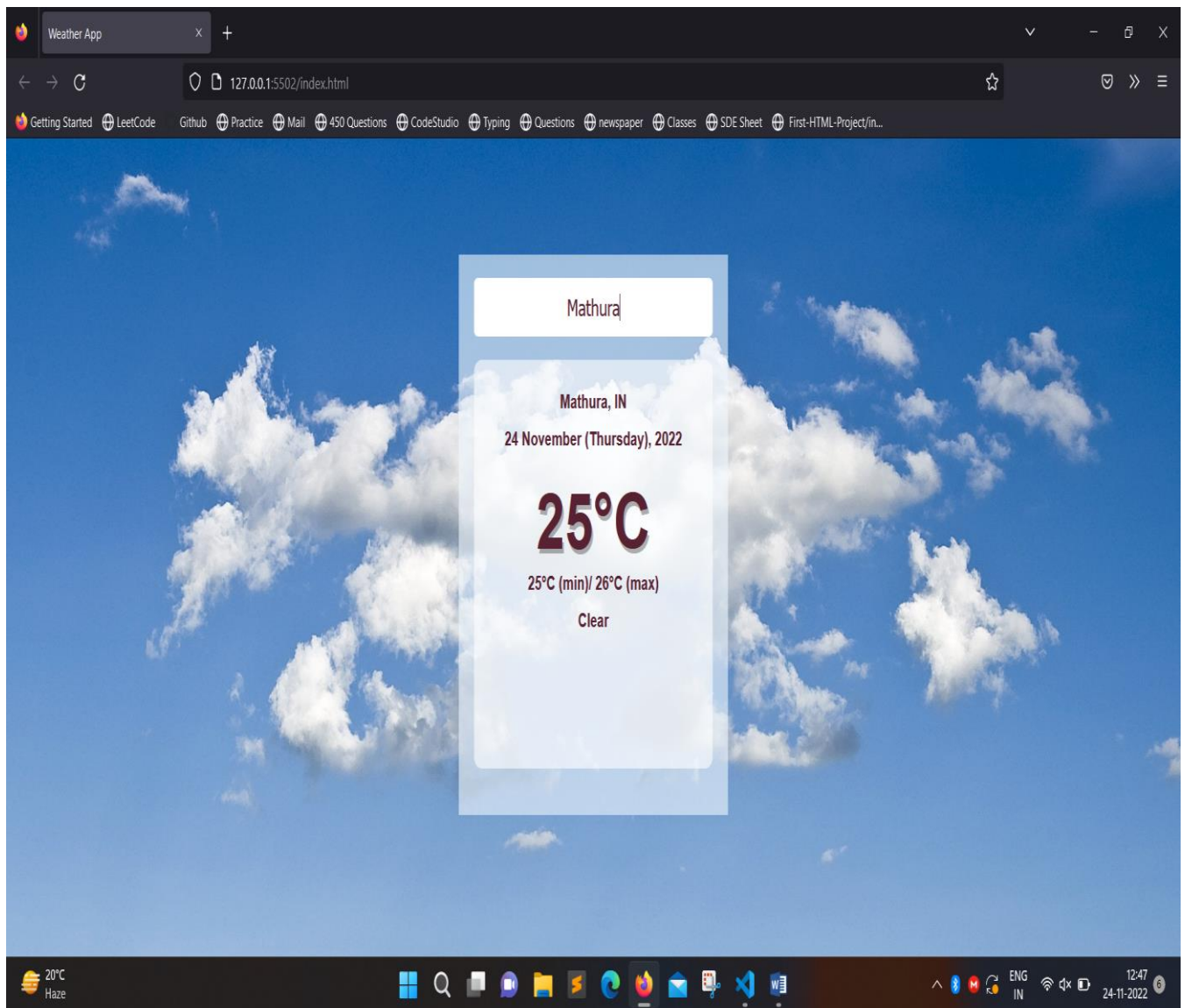
```
31 console.log(weather);
32
33 let city = document.getElementById('city');
34 city.innerHTML = `${weather.name}, ${weather.sys.country}`;
35
36 let temperature = document.getElementById('temp');
37 temperature.innerHTML = `${Math.round(weather.main.temp)}&deg;C`;
38
39 let minMaxTemp = document.getElementById('min-max');
40 minMaxTemp.innerHTML = `${Math.floor(weather.main.temp_min)}&deg;C (min)/ ${Math.ceil(weather.main.temp_max)}&deg;C (max)`;
41
42 let weatherType = document.getElementById('weather');
43 weatherType.innerHTML = `${weather.weather[0].main}`;
44
45 let date = document.getElementById('date');
46 let todayDate = new Date();
47 date.innerHTML = dateManage(todayDate);
48
49
50 if(weatherType.textContent == 'Clear') {
51     document.body.style.backgroundImage = "url('images/clear.jpg')";
52 } else if(weatherType.textContent == 'Clouds') {
53     document.body.style.backgroundImage = "url('images/cloud.jpg')";
54 } else if(weatherType.textContent == 'Haze') {
55     document.body.style.backgroundImage = "url('images/cloud.jpg')";
56 } else if(weatherType.textContent == 'Rain') {
57     document.body.style.backgroundImage = "url('images/rain.jpg')";
58 }
59
60
61
62
63
```

# JavaScript Code



```
60
61     } else if(weatherType.textContent == 'Rain') {
62
63         document.body.style.backgroundImage = "url('images/rain.jpg')";
64
65     } else if(weatherType.textContent == 'Snow') {
66
67         document.body.style.backgroundImage = "url('images/snow.jpg')";
68
69     } else if(weatherType.textContent == 'Thunderstorm') {
70
71         document.body.style.backgroundImage = "url('images/thunderstorm.jpg')";
72
73     }
74 }
75
76 // Date manage
77 function dateManage(dateArg) {
78
79     let days = ["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"];
80
81     let months = ["January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"];
82
83     let year = dateArg.getFullYear();
84     let month = months[dateArg.getMonth()];
85     let date = dateArg.getDate();
86     let day = days[dateArg.getDay()];
87
88     return `${date} ${month} (${day}), ${year}`;
89 }
90
91
92
93
```

## Result



## **Conclusion**

We have completed our project within time limit with the coordination of our team members under the supervision of our mentor Mr. Bhanu Kapoor.



## **Bibliography**

[www.google.com](http://www.google.com)

[\*\*www.geeksforgeeks.org\*\*](http://www.geeksforgeeks.org)

[www.youtube.com](http://www.youtube.com)