

WEATHER APP

Introduction

Overview:

- ❖ The Weather.io is a web application that provides real-time weather information for a specified location. It utilizes the Open Weather Map API to fetch weather data and displays it in a user-friendly interface. Users can search for a location by city name and receive detailed weather information, including temperature, humidity, wind speed, and weather conditions.
- ❖ Weather.io is the application of science and technology to predict the state of the atmosphere for a given location. Ancient weather application relied on observed patterns of events, also termed pattern recognition. For example, it might be observed that if the sunset was particularly red, the following day often brought fair weather.

Purpose:

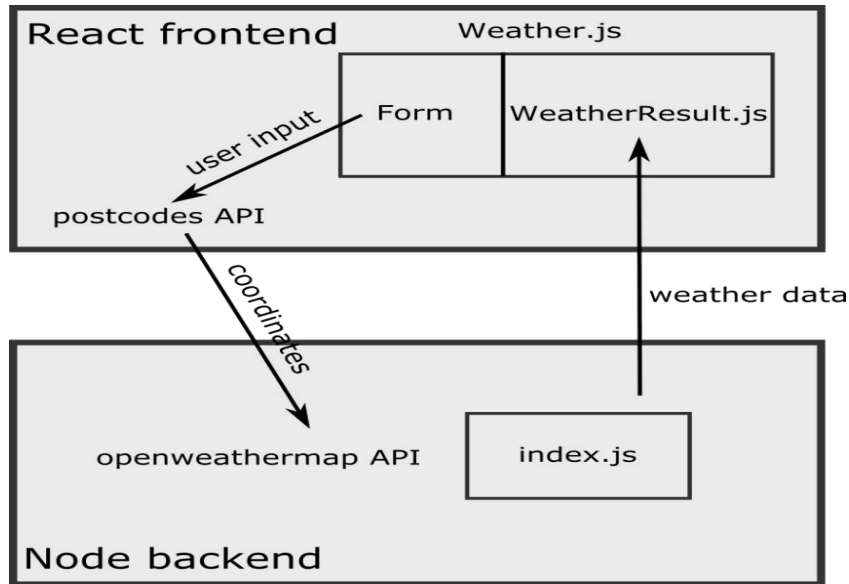
- ❖ In this weather app you enter the cities name or countries and its tell weather of the area and we add one function that tells there's Haze or cloud. So let's create this amazing weather app. Weather app used API to retrieve weather data for all cities and countries.
- ❖ Weather app is something everybody deals with, and accurate data of it like what is coming can help to make informed decisions.

1 Literature Survey

This specific project is not only the most easy topic but also rarely used. For instance there exist more than 15 Weather Applications and only 3 have managed to climb the cliff Downloads. With the market so low, entering into this was indeed difficult and the greatest risk. As Steve Jobs once said "There is no such thing as a simple project, make it complex", just what I did here. With every review, every guidance, every step I have reached one step closer to what is now a baby. The project, the Application in itself is enough explain and need no marketing at any point of time. Application stands for itself. It is indeed very important to take care of oneself, no one ever can ever predict rain, nor a storm. Climatic change is subject to risk. In interest of social service and user convenience that this app is perfect of a person who is climate conscious and doesn't use his /her phone well. This app will stand out from others for sure.

2 THEORATICAL ANALYSIS

Block diagram:



Hardware and software designing

Hardware and software requirements of the project:

Accessing a database:-

- ❖ The system should allow administrator to add historical weather data.
- ❖ The system should be able to recognize patterns in temperature ,humidity,and wind eith use of historical data .

Software constraints:-

- ❖ The development of the system will be constrained by the availability of required software such as webserver,dataset,and development tools.

Hardware Requirements:-

- ❖ The system requires a database in order to store persistent data.

HTML

```
!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta name="viewport" content="width=device-width", initial-scale="1.0">
```

```
  <title>Weather App </title>
```

```
  <link rel="stylesheet" href="style.css">
```

```
</head>
```

```
<style>
```

```
  body {
```

```
    background-image: url("images/background.png");
```

```
    background-repeat: no-repeat;
```

```
    background-attachment: fixed;
```

```
    background-size: cover;
```

```
  }
```

```
</style>
```

```
<body>
```

```
  <div class="card">
```

```
    <div class="search">
```

```
      <input type="text" placeholder="enter city name" spellcheck="false">
```

```
      <button> </button>
```

```
    </div>
```

```
<div class="weather">



  <div class="weather__datetime">

    <p>Monday 31 July 2023 11:00 AM </p>

  </div>

  <h1 class="temp">°c</h1>

  <h2 class="city">--</h2>

  <div class="details">

    <next>

    <div class="col">

      <div>

        <p class="humidity">50%</p>

        <p>Humidity</p>

      </div>

    </div>

    <div class="col">

      <div>

        <p class="Wind">15km/h</p>

        <p>Wind speed</p>

      </div>

    </div>

  </div>

</div>
```

</next>

<next>

<div class="col">

<div>

<p class="high">25°C</p>

<p>feelslike</p>

</div>

</div>

<div class="col">

<div>

<p class="pressure">1005 pa</p>

<p>pressure</p>

</div>

</div>

</next>

</div>

</div>

</div>

<script src="script.js"></script>

</[..\Desktop\weather](#)body>

</html>

CSS

```
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

body {
  font-family: 'montserrat', sans-serif;
  background-image: url('bg.jpg');
  background-size: cover;
  background-position: top center;
}

.app-wrap {
  display: flex;
  flex-direction: column;
  min-height: 100vh;
  background-image: linear-gradient(to bottom, rgba(0, 0, 0, 0.3), rgba(0, 0, 0, 0.3));
}

header {
  display: flex;
  justify-content: center;
  align-items: center;
  padding: 50px 15px 15px;
}

header input {
  width: 100%;
  max-width: 280px;
  padding: 10px 15px;
  border: none;
  outline: none;
  background-color: rgba(255, 255, 255, 0.3);
  border-radius: 0px 16px 0px 16px;
  border-bottom: 3px solid gray;

  color: #313131;
  font-size: 20px;
  font-weight: 300;
  transition: 0.2s ease-out;
}

header input:focus {
  background-color: rgba(255, 255, 255, 0.6);
}
```

```
main {
  flex: 1 1 100%;
  padding: 25px 25px 50px;
  display: flex;
  flex-direction: column;
  align-items: center;
  text-align: center;
}

.location .city {
  color: #fff;
  font-size: 32px;
  font-weight: 500;
  margin-bottom: 5px;
}

.location .date {
  color: #fff;
  font-size: 16px;
}

.current .temp {
  color: #fff;
  font-size: 102px;
  font-weight: 900;
  margin: 30px 0px;
  text-shadow: 2px 10px rgba(0, 0, 0, 0.6);
}

.current .temp span {
  font-weight: 500;
}

.current .weather {
  color: #fff;
  font-size: 32px;
  font-weight: 700;
  font-style: italic;
  margin-bottom: 15px;
  text-shadow: 0px 3px rgba(0, 0, 0, 0.4);
}

.current .hi-low {
  color: #fff;
  font-size: 24px;
  font-weight: 500;
  text-shadow: 0px 4px rgba(0, 0, 0, 0.4);
}
```

Javascript

```
const apiKey ="d5555408c08e29140c46c781883f2e1c";
const apiUrl ="https://api.openweathermap.org/data/2.5/weather?units=metric&q=";

const searchBox = document.querySelector(".search input");
const searchBtn = document.querySelector(".search button");
const weatherIcon = document.querySelector(".weather-icon");

async function checkWeather(city){
const response = await fetch(apiurl + city + `&appid=${apiKey}`);
var data= await response.json();
    console.log(data);

    document.querySelector(".city").innerHTML = data.name;
    document.querySelector(".temp").innerHTML = Math.round(data.main.temp) + "°c";
    document.querySelector(".humidity").innerHTML = data.main.humidity + "%";
    document.querySelector(".wind").innerHTML = data.wind.speed + " km/";

    if(data.weather[0].main == "Cloud"){
        weatherIcon.src = "images/Cloud.png";
    }
    else if(data.weather[0].main == "Thunderstroms"){
        weatherIcon.src = "images/Thunderstorms.png";
    }
    else if(data.weather[0].main == "Rain"){
        weatherIcon.src = "images/Rain.png";
    }
    else if(data.weather[0].main== "Sunny"){
        weatherIcon.src = "images/Sunny.png";
    }
    else if(data.weather[0].main == "Error"){
        weatherIcon.src = "images/Error.png";
    }

}

searchBtn.addEventListener("click", ()=>{
    checkWeather(searchBox.value);
})
```

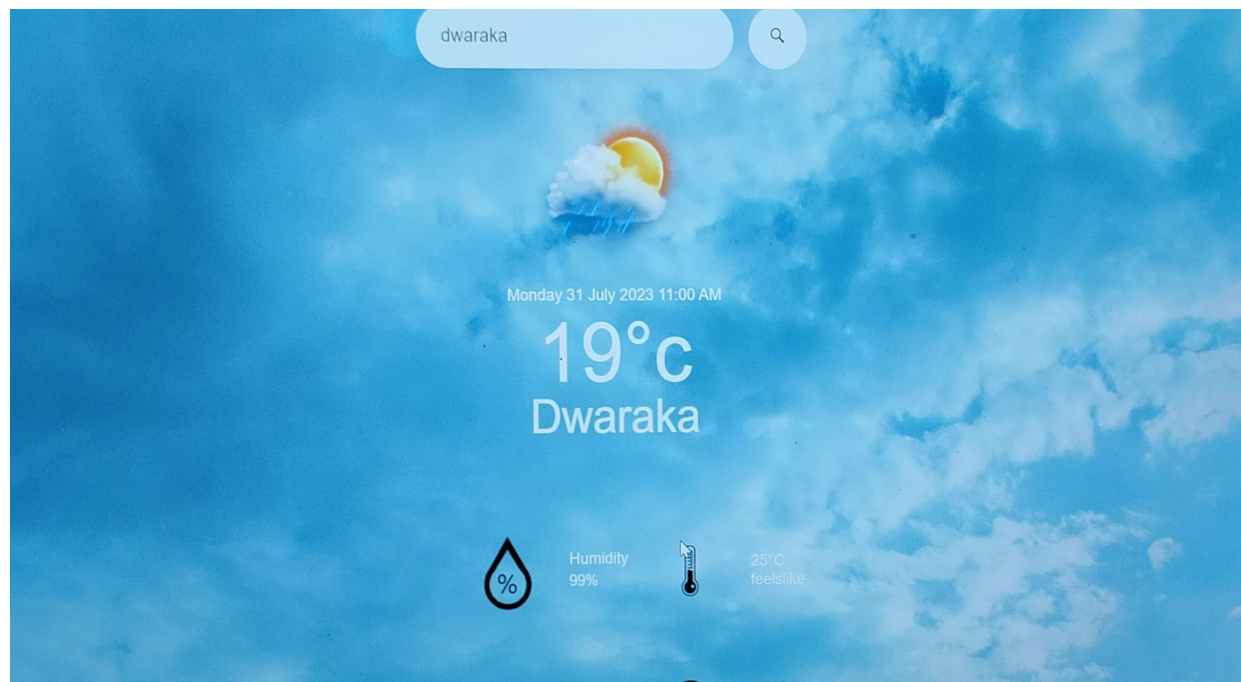
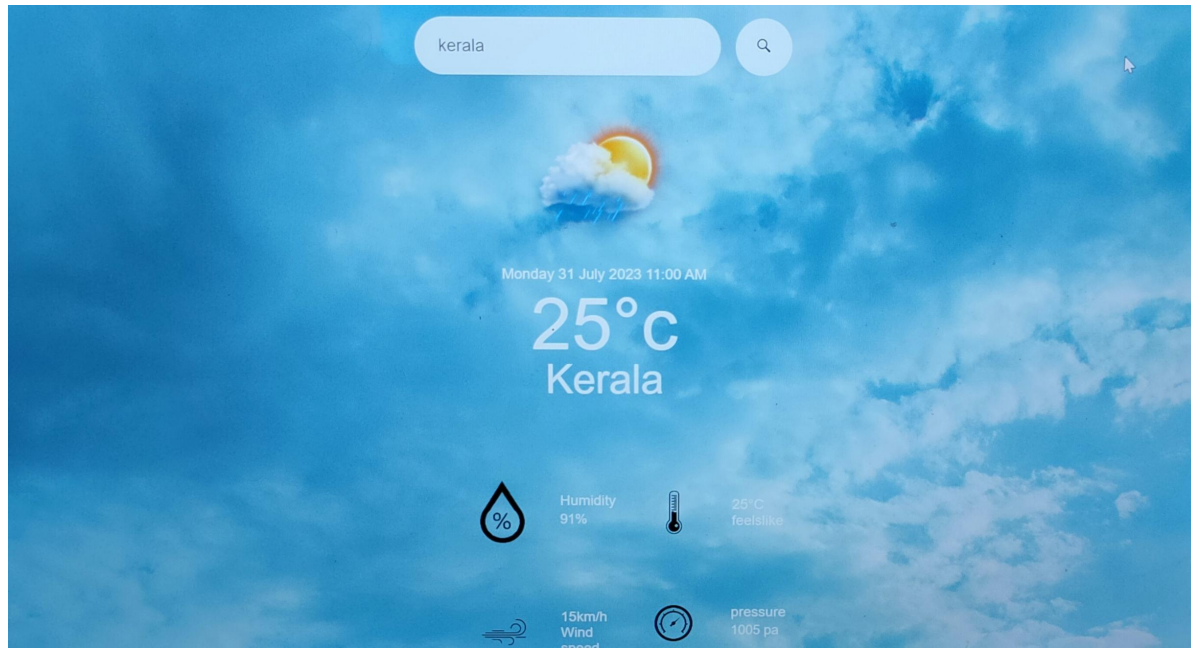


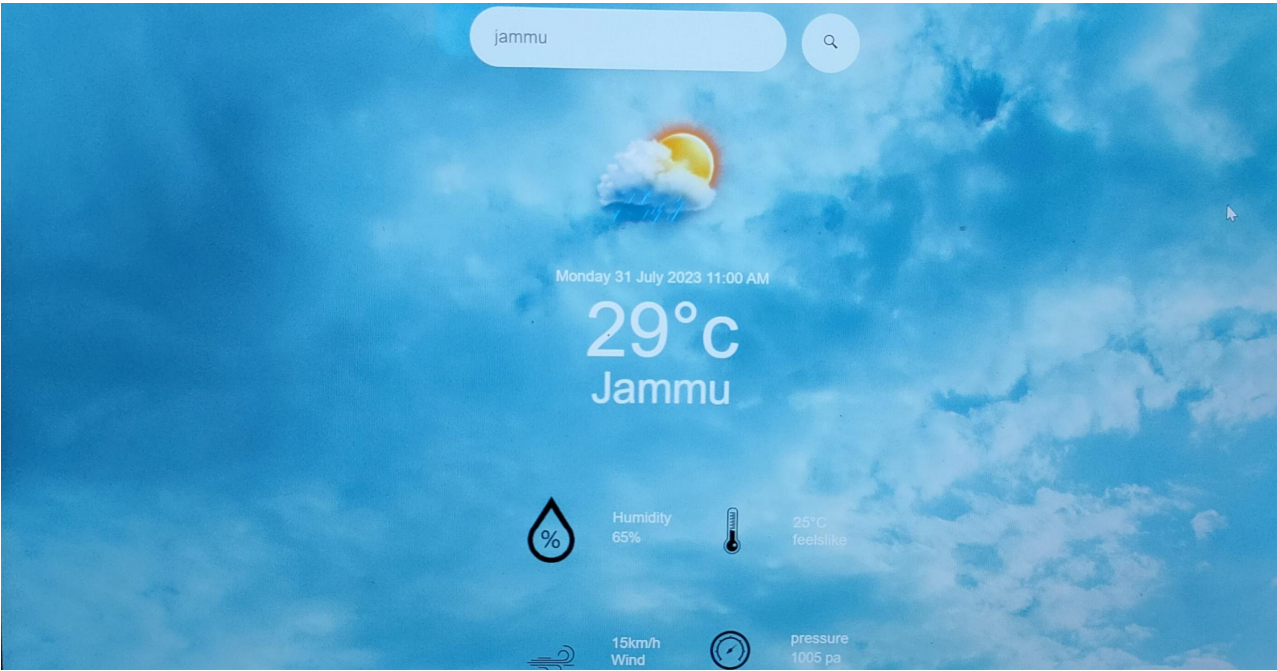
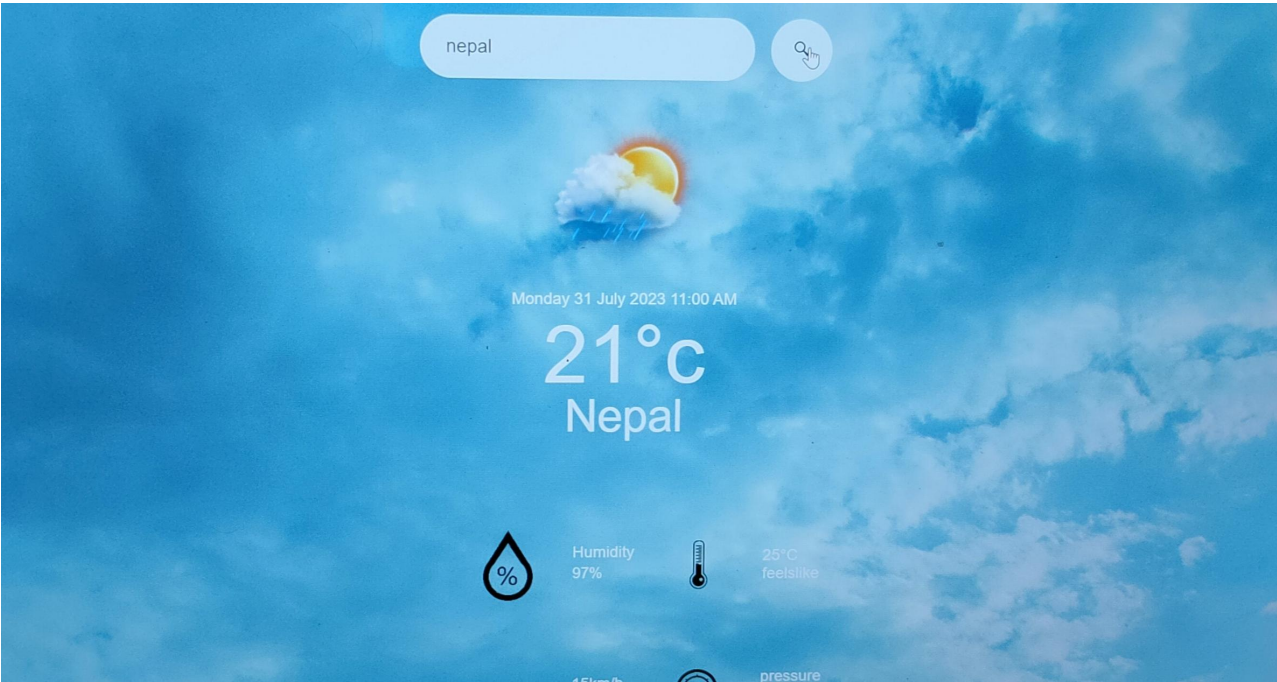
```
let now = new Date();
let date = document.querySelector('.location .date');
date.innerText = dateBuilder(now);
function dateBuilder (d) {
  let months = ["January", "February", "March", "April", "May", "June", "July", "August", "September",
"October", "November", "December"];
  let days = ["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"];

  let day = days[d.getDay()];
  let date = d.getDate();
  let month = months[d.getMonth()];
  let year = d.getFullYear();

  return `${day} ${date} ${month} ${year}`;
}
```

OUTPUT:





5 Advantages and Disadvantages:

- ❖ The weather app being able to predict the data to be gathered to build up a more detailed picture of a nation's climate, trends within it.
- ❖ Weather forecasts are not available for many remote or sparsely populated areas, making it difficult for people in these areas to prepare for severe weather.

6 Applications:

- ❖ A major part of an app is the severe weather alerts and advisories, which the national weather services issue in the case that severe or hazardous weather is expected.
- ❖ Some of the most commonly known severe weather advisories are the severe thunderstorms about areas that are prone to flood.

7. Conclusion:

- ❖ The weather apps are increasingly accurate and useful, and their benefits extend widely across the economy. While much has been accomplished in improving weather forecasts, there remains much room for improvements.
- ❖ Simultaneously, they are developing new technologies and observational networks that can enhance forecaster skill and the value of their services to their users.

8 Future scope

- ❖ The demand for weather and climate forecast information in support of critical decision-making has grown rapidly during the last decade, and will grow even faster in the coming years. Great advances have been made in the utilization of predictions in many areas of human activities.
- ❖ The future of weather applications is promising, with the increasing demand for real-time and accurate weather information. One potential development is the improvement in accuracy through the use of advanced data collection and analysis techniques, as well as sophisticated algorithms.