WEATHER FORECAST

Online Internship

(3170001)

Report Submitted by

RAUSHAN KUMAR SINGH – 210570107091

in partial fulfilment for the award of the degree of

Bachelor of Engineering

in

Computer Engineering

Marwadi Education Foundation, Rajkot

Gujarat Technological University, Ahmedabad

2024-25

Faculty of Engineering (057)

Marwadi Education Foundation

Faculty of Engineering

Computer Engineering Department

2024-25

CERTIFICATE

Date:

TO WHOM IT MAY CONCERN

This is to certify that, Mr./Mrs. **RAUSHAN KUMAR SINGH** Enrolment No. **210570107091** Student of Computer Engineering, has successfully completed a two weeks Internship in the field of **WEATHER FORECAST** during the period of **29/06/2024 TO End 12/ 07/ 2024** for **2 WEEKS**. During the period of his/her internship program with us, He had been exposed to different processes and was found sincere and hardworking.

Institute Mentor
Prof. Ronak Doshi
Assistant Professor
Mentor's Signature

Head of Department
Dr.Krunal Vaghela

INTERNSHIP CERTIFICATE

GUJARAT TECHNOLOGICAL UNIVERSITY

CERTIFICATE FOR COMPLETION OF ALL ACTIVITIES AT ONLINE PROJECT PORTAL

B.E. SEMESTER VII, ACADEMIC YEAR 2024-2025

Date of certificate generation : 14 August 2024 (01:32:24)

This is to certify that, ***Raushan Kumar Singh*** (Enrolment Number - 210570107091) working on project entitled with ***WEATHER*** from ***Computer Engineering*** department of ***MARWADI EDUCATION FOUNDATIONS GROUP OF INSTITUTIONS,RAJKOT*** had submitted following details at online project portal.

Internship Project Report	**Completed**

Name of Student : Raushan Kumar Singh

Name of Guide : HOD_057_07

Signature of Student : _____

*Signature of Guide : _____

Disclaimer :
This is a computer generated copy and does not indicate that your data has been evaluated. This is the receipt that GTU has received a copy of the data that you have uploaded and submitted as your project work.

*Guide has to sign the certificate, Only if all above activities has been Completed.

Acknowledgments

I would like to express my deepest gratitude to everyone who has supported me throughout the completion of this research on weather patterns and climate change.

Firstly, I am profoundly grateful to my advisor**, Prof. Ronak Doshi**, for his unwavering support, expert guidance, and invaluable insights. His mentorship has been instrumental in the success of this study.

I would also like to thank the entire team at the Meteorological Institute for providing access to critical data and resources. Special thanks to **Prof. Ronak Doshi**, and the technical staff for their assistance with data collection and analysis.

A special thanks to my family for their unconditional love and encouragement. Their patience and understanding have been a constant source of strength for me throughout this journey.
Lastly, I am grateful to all the researchers and scientists whose work has laid the foundation for this study.

Thank you all.
Raushan Ku Singh
Enrolment :- 210570107091

INDEX

Abstract

This study analyzes weather patterns and their impacts, focusing on the effects of climate change on a local and global scale. By examining historical weather data, we identify trends and anomalies in temperature, precipitation, and extreme weather events. Our findings show significant changes in weather patterns, with an increase in the frequency and intensity of events like hurricanes, droughts, and heatwaves.

The research also explores the socio-economic implications, emphasizing the heightened vulnerability of certain regions. Our analysis highlights the need for effective adaptive strategies and policies to mitigate climate change impacts.

The results provide insights for policymakers and researchers, enhancing the understanding of climate dynamics. Future work will aim to improve predictive models and broaden the study to include more geographical areas and climatic variables.

List of Tables

List of Figures Weather

List of Symbols, Abbreviations and Nomenclature

Symbol	**Abbreviations**
WWW	World Wide Web
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheets
JS	JavaScript
RAM	Random Access Memory
JSON	Java Script Object Notation
API	Application Programming Interface

Chapter 1: Introduction

1.1 Definition

Weather is a fundamental aspect of our daily lives, influencing a wide range of human activities and natural processes. From agriculture and transportation to health and safety, weather patterns play a crucial role in shaping our environment and society. Understanding weather phenomena is not only essential for predicting short-term conditions but also for comprehending long-term climate trends and their impacts.

This introduction sets the stage for a thorough examination of weather patterns and their implications, providing a foundation for the subsequent analysis and discussion. The findings of this study will contribute to the broader understanding of climate dynamics and offer valuable insights for policymakers, researchers, and the general public.

1.2 PROJECT DESCRIPTION:

- **Product Perspective:**

This app provides commercial weather forecasting services worldwide. it will utilize ideas passed by open weather map and national weather services.

- **System Interface:**

The user interface for the task will have a site that will have the live feed alongside the data about the climate. This site will utilize html, CSS, JavaScript and API requests for site.

- **Product Functions:**

- It will show the current weather of the provided location.
- It will show the current wind speed of the location.
- It will also show the humidity level of the provided location.

- **Operating Environment:**

Today weather app is an web application which you can run using your favourite browser. In the application provided details and information is provided by the open weather map website which will help us for API data fetching

Assumptions:	**Dependencies**
Data Accuracy	Api Integration
Stable Internet connection	Geolocation Services
User Location Accuracy	Server Infrastructure
Data Source Availability	Data Storage
	Internet Connectivity

Functional Requirements | Todays Weather App:

- **Software Requirements:**

This software package is developed using html, CSS for frontend and JavaScript for the backend. Using Vs Code as a text editor and Google Chrome for the execution of our code.

Required Specifications for our Device:

- **Operating System**: Windows 7, 8, 9, 10.
- **Language**: Html , Css , Javascript.
- **API**: Openweathermap,API

- **Hardware Requirements:**

- **Processor**: Intel core i3 or above for a stable experience and fast retrieval of data.
- **Hard Disk**: 4GB and above
- **RAM**: 256 MB or more, recommended 2 GB for fast reading and writing capabilities which will result in better performance time.

PROJECT SCOPE:

A weather forecasting application is a software tool designed to provide users with up-to-date and accurate information about current and future weather conditions. These applications leverage data from meteorological sources, satellites, and weather stations to deliver forecasts, real-time weather updates, and other related information.

The primary goal of a weather forecasting app is to offer users the ability to plan their activities based on anticipated weather conditions

PROJECT REQUIREMENTS:

This is the next phase after the submission of the synopsis report. We can do this process before the Synopsys report creation as well, It is all depends upon the project and their requirements. Here after getting an overview about the project now we can easily do the requirement gathering for our project.

Chapter 2 : Project Analysis

This project analyzes historical and current weather patterns to understand their evolutionand implications. The analysis is structured into several key components:

Data Collection:

Sources: Data was gathered from reputable meteorological agencies, including the National Weather Service and international climate databases.

Types of Data: Temperature records, precipitation levels, and records of extreme weather events (e.g., hurricanes, droughts).

Methodology:

Statistical Analysis: Employed statistical tools to identify trends and anomalies in the historical data.

Computational Models: Used machine learning models to predict future weather patterns based on historical data.

Geospatial Analysis: Mapped weather changes to visualize geographical impacts and regional variations.

Findings:

Temperature Trends: A significant rise in global temperatures over the past century, with notable increases in the last few decades.

Precipitation Patterns: Shifts in precipitation patterns, with some regions experiencing more intense rainfall and others facing prolonged droughts.

Extreme Weather Events: An increase in the frequency and intensity of events such as hurricanes, heatwaves, and floods.

Socio-Economic Impacts:

Vulnerable Regions: Identified regions most affected by changing weather patterns, particularly those with limited resources to adapt. Economic Costs: Analyzed the economic impact of weather-related disasters, including damage to infrastructure and agricultural losses.Public Health: Assessed the impact on public health, highlighting the rise in weather-related illnesses and fatalities.

Recommendations:

Adaptive Strategies: Suggested adaptive measures for vulnerable regions, such as improved infrastructure, better forecasting systems, and emergency preparedness plans.

Policy Implications: Highlighted the need for comprehensive climate policies at local, national, and international levels to mitigate the adverse effects of climate change.

Conclusion:

This project provides a comprehensive analysis of weather patterns and their impacts. The findings underscore the urgency of addressing climate change and implementing effective strategies to mitigate its effects. Further research and continuous monitoring are essential to enhance our understanding and response to these ongoing changes.

2.1 USE CASE DIAGRAM

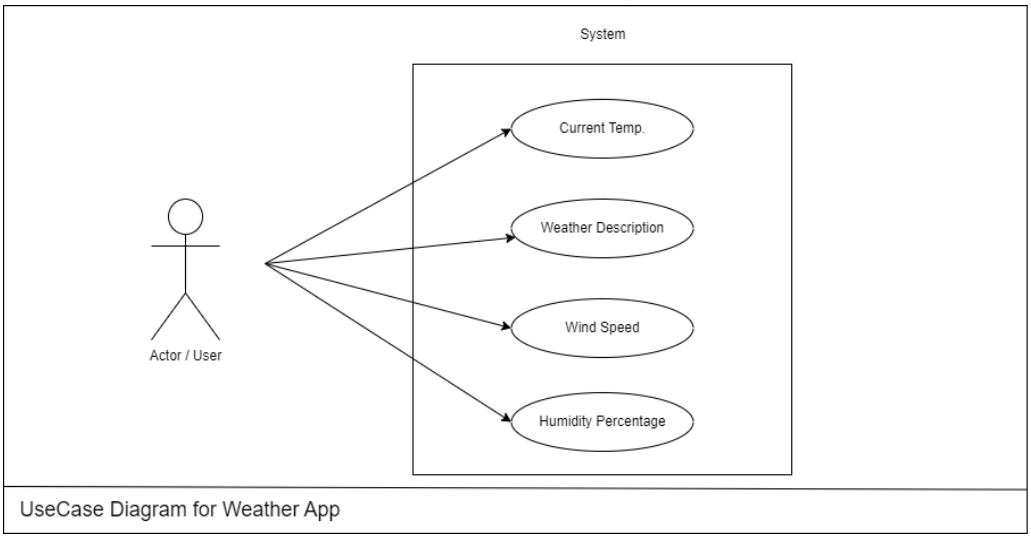

Figure 2.1 Use Case Diagram

2.2 Object Diagram

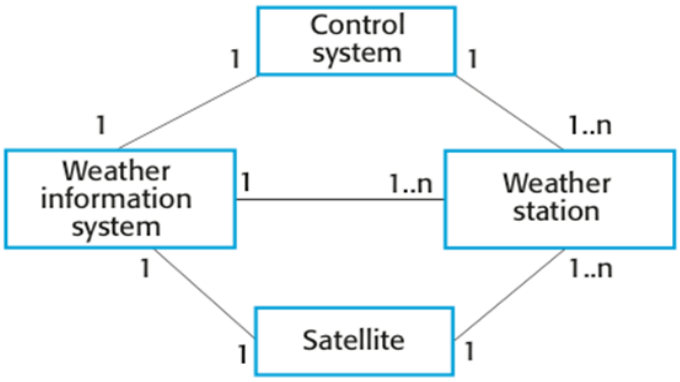

Figure 2.2 Object Diagram

2.3 Class Diagram

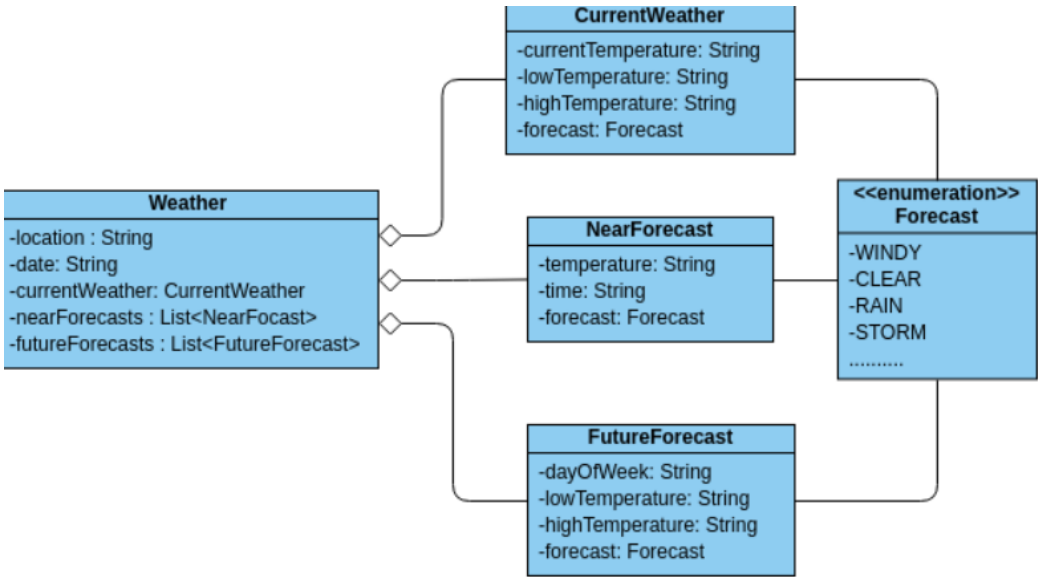

Figure No 2.3 Class Diagram

2.4 Data Diagram

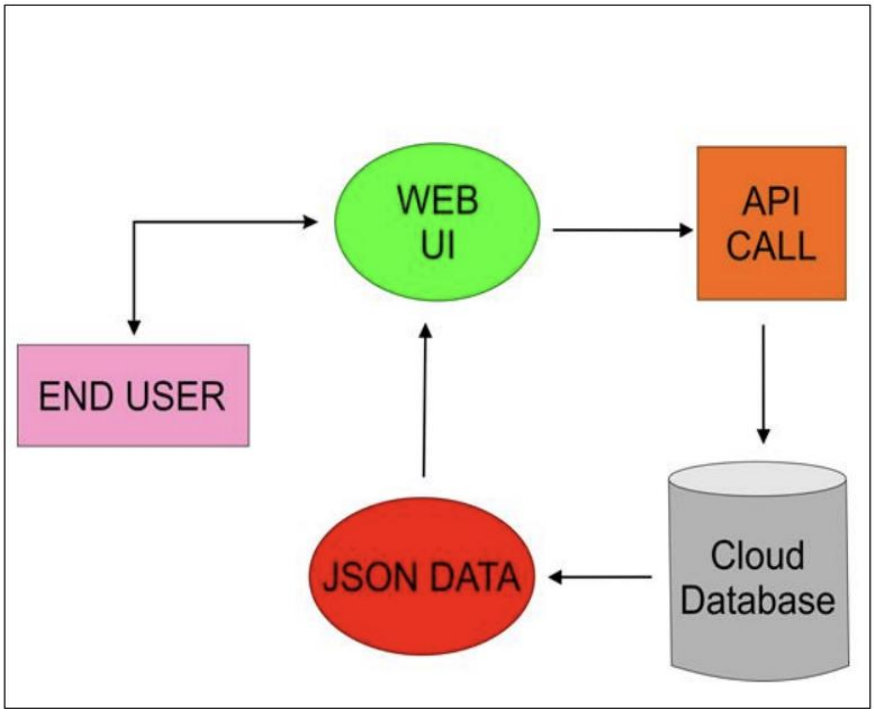

Figure No. 2.4 Data Diagram

2.5 Sequence Diagram

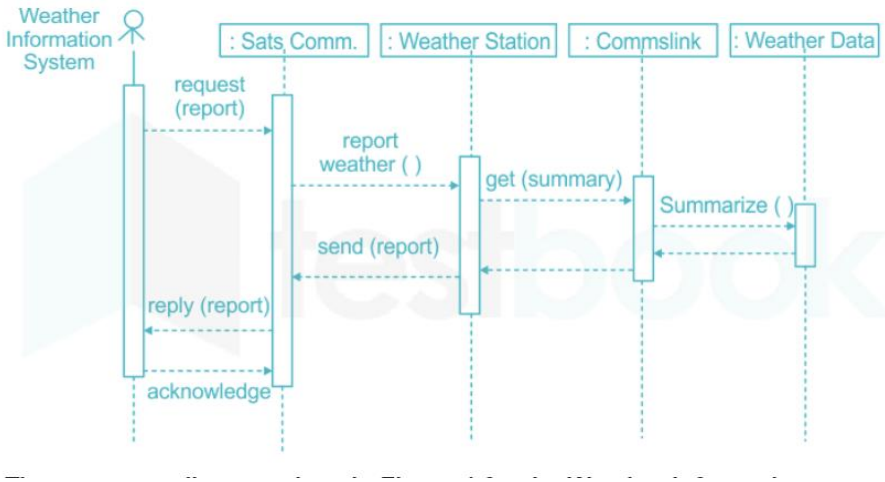

Figure No 2.5 Sequence Diagram

2.6 Activity Diagram

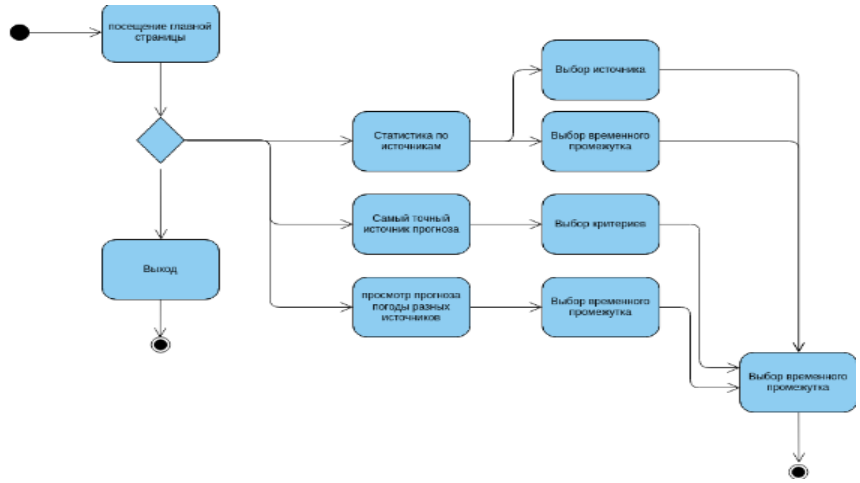

Figure No 2.6 Activity Diagram

2.7 Data Dictionary (Database)

Variable name	Variable type	Variable range or levels	How measured	Units/formats/level coding
Date	date	[yyyy-mm-dd hh:mm_ss]	logged	
Temperature (value)	numeric		logged	°C
turfID	factor	[site]x[block]x[treatment]	defined	A1-C – L7-OTC
Site	factor	[site]	defined	High alpine (H), Alpine (A), Middle (M), Lowland (L)
Treatment	Factor	[treatment code]	Defined	Control (C), OTC
depth	factor	[code]	defined	air (30 cm), ground (0 cm), soil (−5 cm)

Chapter 3: Project Implementation

3.1 Implementation of Weather Forecast Application Frontend Development

In this stage we are going to develop our frontend part of the project.

This is how our page will looks like:

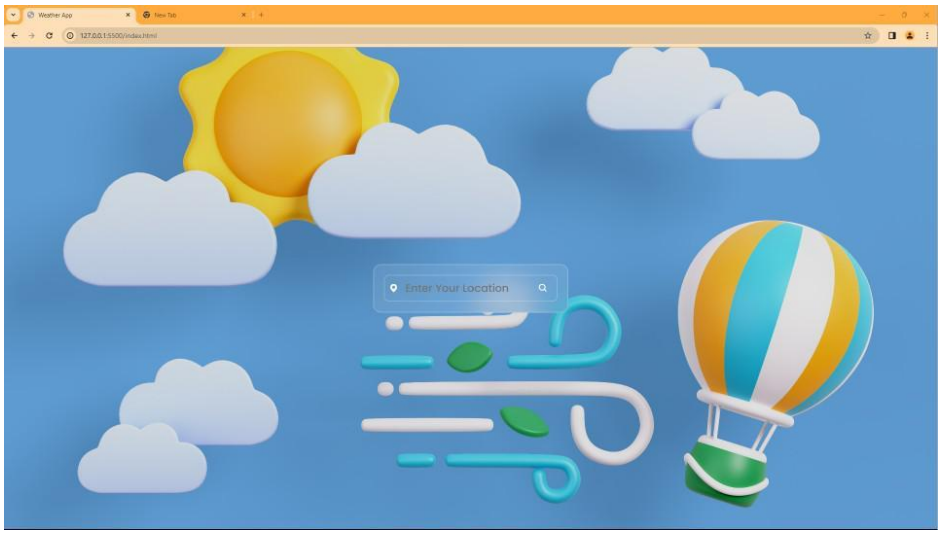

Figure No 3.1 Home Page

- After providing the city value by user the app will show us the Temperature of the city with weather description along with the humidity and wind speed in that city.

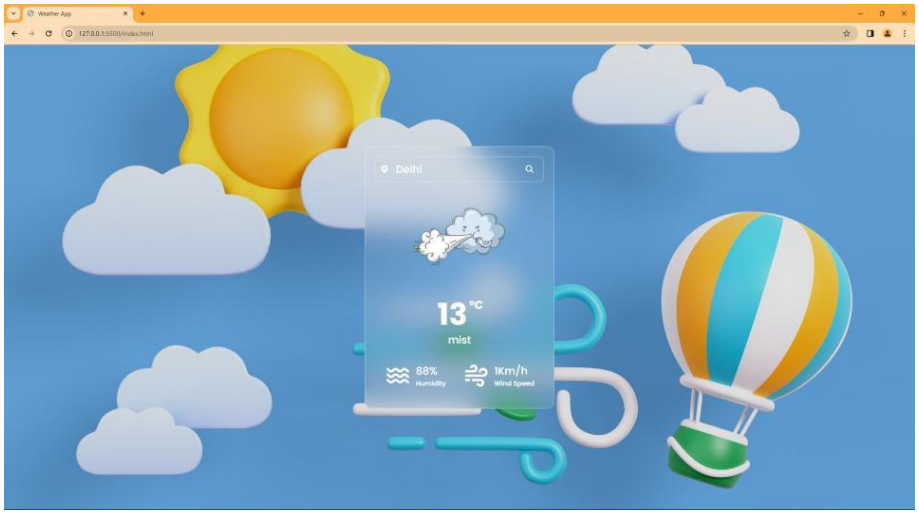

Figure No 3.2 Result Page

- If user provide the any invalid data in the search field then our page will respond with an 404 error

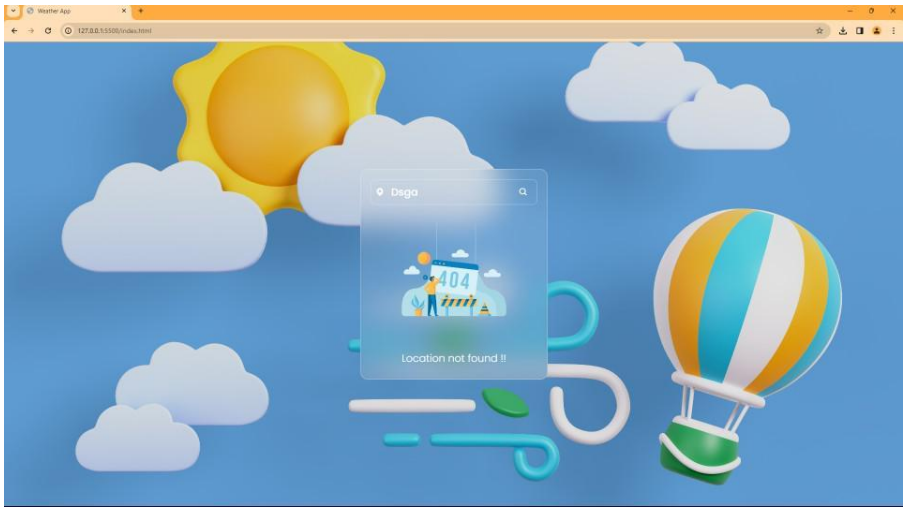

Figure No 3.3 Error Page

3.2 Functionalities of this page:

- User can put the City name and get the weather details:
- Temperature
- Humidity
- Wind Speed
- Weather Description
- Easy User Interface

➢ **Code:**

- Below is the Code for Creating above page:

Script.js

```
const container = document.querySelector('.container');

const search = document.querySelector('.search-box button');

const weatherBox = document.querySelector('.weather-box');

const weatherDetails = document.querySelector('.weather-details');

const error404 = document.querySelector('.not-found');
```

```
search.addEventListener('click',() =>{
    // Enter your API Key in the APIKey variable
    // You can use any weather api for the project
    // Here we are using openweathermap's API which
    // you can find in their website by searching
    // weather API.
    const APIKey = '';
    const city = document.getElementById('search-btn').value;
    if(city==''){
        return; }

fetch(`https://api.openweathermap.org/data/2.5/weather?q=${city}&units=metric&appid=${APIKey}`).the
n(response => response.json()).then(json => {
    if(json.cod == '404'){
        container.style.height = '450px';
        weatherBox.classList.remove('active');
        weatherDetails.classList.remove('active');
        error404.classList.add('active');
        return;
    }
    container.style.height = '560px';
    weatherBox.classList.add('active');
    weatherDetails.classList.add('active');
    error404.classList.remove('active');
    const image = document.querySelector('.weather-box img');
    const temperature = document.querySelector('.weather-box .temperature');
    const description = document.querySelector('.weather-box .description');
    const humidity = document.querySelector('.weather-details .humidity span');
    const wind = document.querySelector('.weather-details .wind span');
        switch(json.weather[0].main){
            case 'Clear':
```

```
            image.src = 'images/clear-new.png';
            break;
        case 'Rain':
            image.src = 'images/rain-new.png';
            break;
        case 'Snow':
            image.src = 'images/snow-new.png';
            break;
        case 'Clouds':
            image.src = 'images/cloud-new.png';
            break;
        case 'Mist':
            image.src = 'images/mist-new.png';
            break;
        case 'Haze':
            image.src = 'images/mist-new.png';
            break;
        default:
            image.src = 'images/clear-new.png';
    }
    temperature.innerHTML = `${parseInt(json.main.temp)}<span>°C</span>`;
    description.innerHTML = `${json.weather[0].description}`;
    humidity.innerHTML = `${json.main.humidity}%`;
    wind.innerHTML = `${parseInt(json.wind.speed)}Km/h`;
  });
});
```

Index.html

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

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Weather App</title>
  <link rel="stylesheet" href="style.css">
  <link href="https://unpkg.com/boxicons@2.1.4/css/boxicons.min.css" rel="stylesheet">
</head>
<body>
  <div class="container">
     <div class="search-box">
        <i class='bx bxs-map'></i>
        <input id = "search-btn" type="text" placeholder="Enter Your Location">
        <button class="bx bx-search"> </button>
     </div>
     <div class="weather-box">
        <div class="box">
           <div class="info-weather">
              <div class="weather">
                 <img src="images/cloud.jpg"></img>
                 <p class="temperature"> 0 <span>°C</span></p>
                 <p class="description">Broken clouds</p>
              </div>
           </div>
        </div>
     </div>
     <div class="weather-details">
        <div class="humidity">
           <i class="bx bx-water"></i>
           <div class="text">
```

```
                <div class="info-humidity">
                    <span>0%</span>
                </div>
                <p>Humidity</p>
            </div>
        </div>
        <div class="wind">
            <i class="bx bx-wind"></i>
            <div class="text">
                <div class="info-wind">
                    <span>0Km/h</span>
                </div>
                <p>Wind Speed</p>
            </div>
        </div>
    </div>
    <div class="not-found">
        <div class="box">
            <img src="images/404-new.png">
            <p> Location not found !!</p>
        </div>
    </div>
  </div>
  <script src="script.js"></script>
 </body>
</html>
```

Style.css

```
@import
url("https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;500;600;700;800;900&dis
play=swap");

*{
```

```
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    font-family: "Poppins",sans-serif;
}
body{
    display: flex;
    justify-content: center;
    align-items: center;
    min-height: 100vh;
    /* background: url('images/background_img.jpg'); */
    background-color: black;
    background-size: cover;
    background-position: center;
}
.container{
    position: relative;
    width: 400px;
    height: 100px;
    background: rgb(245,245,245,0.1);
    backdrop-filter: blur(30px);
    border: 2px solid rgba(255,255,255,.2);
    border-radius: 16px;
    padding: 20px;
    color: #fff;
    transition: height .6s ease;
}
.search-box{
    position: relative;
    width: 100%;
```

```
    height: 55px;
    display: flex;
    align-items: center;
}
.search-box i{
    position: absolute;
    left: 10px;
    border-radius: 10px;;
    font-size: 20px;
}
.search-box input{
    position: absolute;
    width: 100%;
    height: 100%;
    background: transparent;
    border: 2px solid rgba(255,255,255,.2);
    outline: none ;
    border-radius: 10px;
    font-size: 22px;
    color: #fff;
    font-weight: 500;
    text-transform: uppercase;
    padding: 0 48px 0 42px;
}
.search-box input{
    color: #fff;
    text-transform: capitalize;
}
.search-box button{
    position: absolute;
```

```
    right: 0;
    width: 40px;
    height: 100%;
    background: transparent;
    border: none;
    outline: none;
    font-size: 20px;
    color: #fff;
    padding: 0 40px 0 5px;
    cursor: pointer;
}
.weather-box{
    text-align: center;
    margin: 10px 0;
}
.weather-box,
.weather-details,
.not-found{
    overflow: hidden;
    visibility: hidden;
}
.weather-box.active,
.weather-details.active,
.not-found.active{
    visibility: visible;
}
.weather-box .box,
.not-found .box{
    transform: translateY(-100%);
}
```

```
.weather-box.active .box,
.not-found.active .box{
    transform: translateY(0%);
    transition: transform 1s ease;
    transition-delay: .6s;
}
.weather-box img{
    width: 60%;
}
.weather-box .temperature{
    position: relative;
    font-size: 64px;
    line-height: 1;
    font-weight: 700;
    margin: 20px 0 6px -30px;
}
.weather-box .temperature span{
    position: absolute;
    font-size: 24px;
    margin-left: 4px;
}
.weather-box .description{
    position: relative;
    font-size: 22px;
    font-weight: 500;
}
.weather-details{
    position: absolute;
    bottom: 40px;
    left: 0;
```

```
    width: 100%;
    padding: 0 20px;
    display: flex;
}
.weather-details .humidity,
.weather-details .wind{
    display: flex;
    align-items: center;
    width: 50%;
    transform: translateY(-100%);
}
.weather-details.active .humidity,
.weather-details.active .wind{
    transform: translateY(0%);
    transition: transform 1s ease;
    transition-delay: 1.0s;
}
.weather-details i{
    font-size: 56px;
    margin-right: 10px;
}
.weather-details .humidity{
    padding-left: 20px;
    justify-content: flex-start;
}
.weather-details .wind{
    padding-right: 20px;
    justify-content: flex-end;
}
.weather-details span{
```

```
    display: inline-block;
    font-size: 22px;
    font-weight: 500;
}
.weather-details p{
    font-size: 14px;
    font-weight: 500;
}
.not-found{
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    text-align: center;
    margin-top: 110px;
}
.not-found img{
    width:  65%;
}
.not-found p{
    font-size: 22px;
    font-weight: 200;
    margin-top: 12px;
}
```

CHAPTER 4: SUMMARY

4.1 Future Enhancements for Weather forecasting Application:-

- We can add last 10 days timeline of weather forecasting of the selecte location as well
- Integration of future weather prediction using Machine learning technologies.
- We can add top weather headlines in the nearby locations as well.
- Integration of any natural disaster prediction in the location

4.2 CONCLUSION

- Helps users plan activities weather apps can help users plan activities bases anticipated weather conditions
- Helps users monitor weather patterns weather apps can provide statical information and trends make can help users monitor weather patterns over time and make data-driven decisions.
- Weather apps can help users make better decisions about their day-to -day activities by providing real-time weather information,forecasts,and other weather -related data
- Here are some conclusions that can be drawn from a weather app project

References

1. Data Mining: The Textbook 2015 Edition, Kindle Editionby Charu C. Aggarwal.
2. Data Mining: Concepts and Techniques By Jiawei Han, Jian Pei, Micheline Kamber
3 Weather Forecaster to Research Scientist: My Career in Meteorology by Robert M. Atlas foreword by Dave Jones

1. https://towardsdatascience.com/introduction-to-machine-learning-algorithms-linear-regression-14c4e325882a
2. https://www.kaggle.com/grubenm/austin-weather

1. **JavaScript Official Documentation:** Comprehensive reference for JavaScript, including language features and APIs.MDN Web Docs – JavaScript https://www.w3schools.com/javascript/

For API Integration: • API Documentation (General): Comprehensive resource for understanding API concepts and integration.MDN Web Docs - Working with APIs
1. **Fetch API Documentation**: Detailed guide on using the Fetch API for making network requests.MDN Web Docs - Fetch API
2. **API Tutorial on W3Schools**: Interactive guide on how to work with APIs and perform data operations. https://www.w3schools.com/js/js_api_intro.asp