



PROJECT REPORT

ON

“WEATHER.IO:Weather App”

A Project Report Submitted in Partial fulfillments of Requirements for the Award of the Degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

Submitted To



Submitted By

S.NO	Team Id	Register Number	Student Name
1	LTVIP2023TMID05256	SBAP0005256	Sreerama Threeveni
2		SBAP0005295	Battu Anusha
3		SBAP0005297	Deepala Gowri Akhila
4		SBAP0005300	Javvaji Kavya
5		SBAP0005306	Shaik Sanjarimehtab

Contents

Chapter 1: INTRODUCTION

- 1.1 Overview A brief description about your project
- 1.2 Purpose The use of this project. What can be achieved using this.

Chapter 2: LITERATURE SURVEY

- 2.1 Existing problem Existing approaches or method to solve this problem
- 2.2 Proposed solution What is the method or solution suggested by you?

Chapter 3: THEORITICAL ANALYSIS

- 3.1 Block diagram Diagrammatic overview of the project.
- 3.2 Hardware / Software designing Hardware and software requirements of the project

Chapter 4: RESULT

- 4.1 Final findings (Output) of the project along with screenshots.

Chapter 5: ADVANTAGES & DISADVANTAGES

- 5.1 List of advantages and disadvantages of the proposed solution

Chapter 6: APPLICATIONS

- 6.1 The areas where this solution can be applied

Chapter 7: CONCLUSION

- 7.1 Conclusion summarizing the entire work and findings.

Chapter 8: FUTURE SCOPE

- Enhancements that can be made in the future.

Chapter 1

INTRODUCTION

Weather forecasting is the application of science and technology to predict the state of the atmosphere for a given location. Ancient weather forecasting methods usually 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. However, not all of these predictions prove reliable.

1.1 Overview A brief description about your project

Weather forecasting is the application of science and technology to predict the conditions of the atmosphere for a given location and time. People have attempted to predict the weather informally for millennia and formally since the 19th century.

Weather forecasts are made by collecting quantitative data about the current state of the atmosphere, land, and ocean and using meteorology to project how the atmosphere will change at a given place.

Once calculated manually based mainly upon changes in barometric pressure, current weather conditions, and sky conditions or cloud cover, weather forecasting now relies on computer-based models that take many atmospheric factors into account. Human input is still required to pick the best possible model to base the forecast upon, which involves pattern recognition skills, teleconnections, knowledge of model performance, and knowledge of model biases.

The inaccuracy of forecasting is due to the chaotic nature of the atmosphere, the massive computational power required to solve the equations that describe the atmosphere, the land, and the ocean, the error involved in measuring the initial conditions, and an incomplete understanding of atmospheric and related processes. Hence, forecasts become less accurate as the difference between current time and the time for which the forecast is being made (the range of the forecast) increases. The use of ensembles and model consensus helps narrow the error and provide confidence in the forecast.

1.2 Purpose The use of this project. What can be achieved using this.

Purpose

The Weather.io is a web application that provides real-time weather information for a specified location. It utilizes the OpenWeatherMap 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.

Smart Notifications Based on The Current Weather Forecast

The Weather Forecast app for Android and iOS can't be completed without considering this must-have feature, which is called smart notifications or push notifications based on the current weather forecast. As such feature keeps reminding your app users to take an umbrella if there is about to rain outside or prepare themselves for heat strokes or high UV rays.

Hyper-Local Forecast

The next important feature that you can't miss before creating weather app is hyper-local forecast which predicts rain, storm, and weather changes with a per minute accuracy based on the user's current location. To develop this feature, our developers make use of Dark Sky API to gain access to the same data.

Weather Visualizations With Stunning Maps

Along with the prediction of current weather conditions, it is also important to offer your weather forecasting app users with stunning maps which allows them to explore weather conditions in the past and the future.

There are several reasons why weather forecasts are important. They would certainly be missed if they were not there. It is a product of science that impacts the lives of many people.

The following is a list of various reasons why weather forecasts are important:

1. Helps people prepare for how to dress

(i.e. warm weather, cold weather, windy weather rainy weather)

2. Helps businesses and people plan for power production and how much power to use

(i.e. power companies, where to set thermostat)

3. Helps people prepare if they need to take extra gear to prepare for the weather (i.e. umbrella, rain coat, sun screen)

4. Helps people plan outdoor activities

(i.e. to see if rain/storms/cold weather will impact outdoor event)

5. Helps curious people to know what sort of weather can be expected

(i.e. a snow on the way, severe storms)

6. Helps businesses plan for transportation hazards that can result from the weather

(i.e. fog, snow, ice, storms, clouds as it relates to driving and flying for example)

7. Helps people with health related issues to plan the day

(i.e. allergies, asthma, heat stress)

8. Helps businesses and people plan for severe weather and other weather hazards

(lightning, hail, tornadoes, hurricanes, ice storms)

9. Helps farmers and gardeners plan for crop irrigation and protection

(irrigation scheduling, freeze protection).

Pre-Requisites

To complete this project, you will need:

- A code editor (such as Visual Studio Code, Sublime Text, or Atom)
- A web browser
- An internet connection
- HTML, CSS or Bootstrap, and JavaScript knowledge
- OpenWeatherMap API key (sign up at <https://openweathermap.org/> to obtain one)

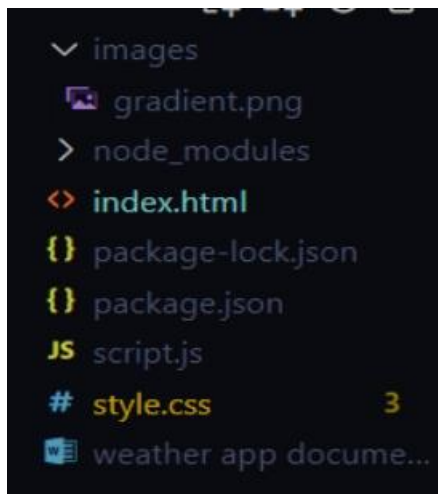
Project Flow

To accomplish the objectives, we will complete the following activities:

- Set up the project structure
- Design and implement the user interface
- Connect to the OpenWeatherMap API
- Fetch weather data based on user input
- Update the UI with the fetched weather data

Project Structure

The project structure will include the following files:



Set up the project structure

Create a new project folder for the Weather App.

Inside the project folder, create the following files/folders:

1. **index.html**
2. **style.css**
3. **script.js**

Chapter 2

LITERATURE SURVEY

2.1 Existing problem Existing approaches or method to solve this problem

Problem Statement

Weather prediction is a useful tool for informing populations of expected weather conditions. Weather prediction is a complex topic and poses significant variation in practice. We will attempt to understand and implement a weather prediction application using the linear regression.

It is important to exactly determine the rainfall for effective use of water resources, crop productivity and pre-planning of water structure.

Another primary thing that the Weather application requires is access to the location. If the location is turned off on your device, the weather application cannot access precise weather information. Hence the problem of the weather app not working.

The Weather application may not be working on your device because of a poor internet connection. Because of that, the weather application won't be able to gather the necessary information. Likely, other causes include:

The device location is turned off.

Outdated Weather Application

Outdated System Software

The Weather App server may be down or under maintenance.

Method of forecasting

Data collection is important for weather forecasting which is collected through land-based station, ocean-based station, air-based station and space-based stations.

Forecasting method can be divided into three broad categories:

- **Conventional or synoptic method:** This method involves detailed analysis of current weather reports from a large area. The current weather patterns are related with the past analogous situation and forecasts are prepared on the assumption that current situation will behave based on past analogous situations. This method is useful for short range forecasts.
- **Statistical methods:** This method is useful for short and medium range forecasting. Regression equations or other sophisticated relationships are established between different weather elements and the resulting climate. Normally selection of predictors or weather parameters is based on a possible physical relationship with the predictant. Multiple regression equation is developed to predict annual rainfall based on 16-parameters by Indian meteorological department.
- **Numerical weather prediction method:** The behaviour of atmosphere is represented by a set of equations based on physical laws governing air movement, air pressure and other information. This technique is suitable for medium range forecasts.

2.2 Proposed solution What is the method or solution suggested by you?

PROPOSED SOLUTION

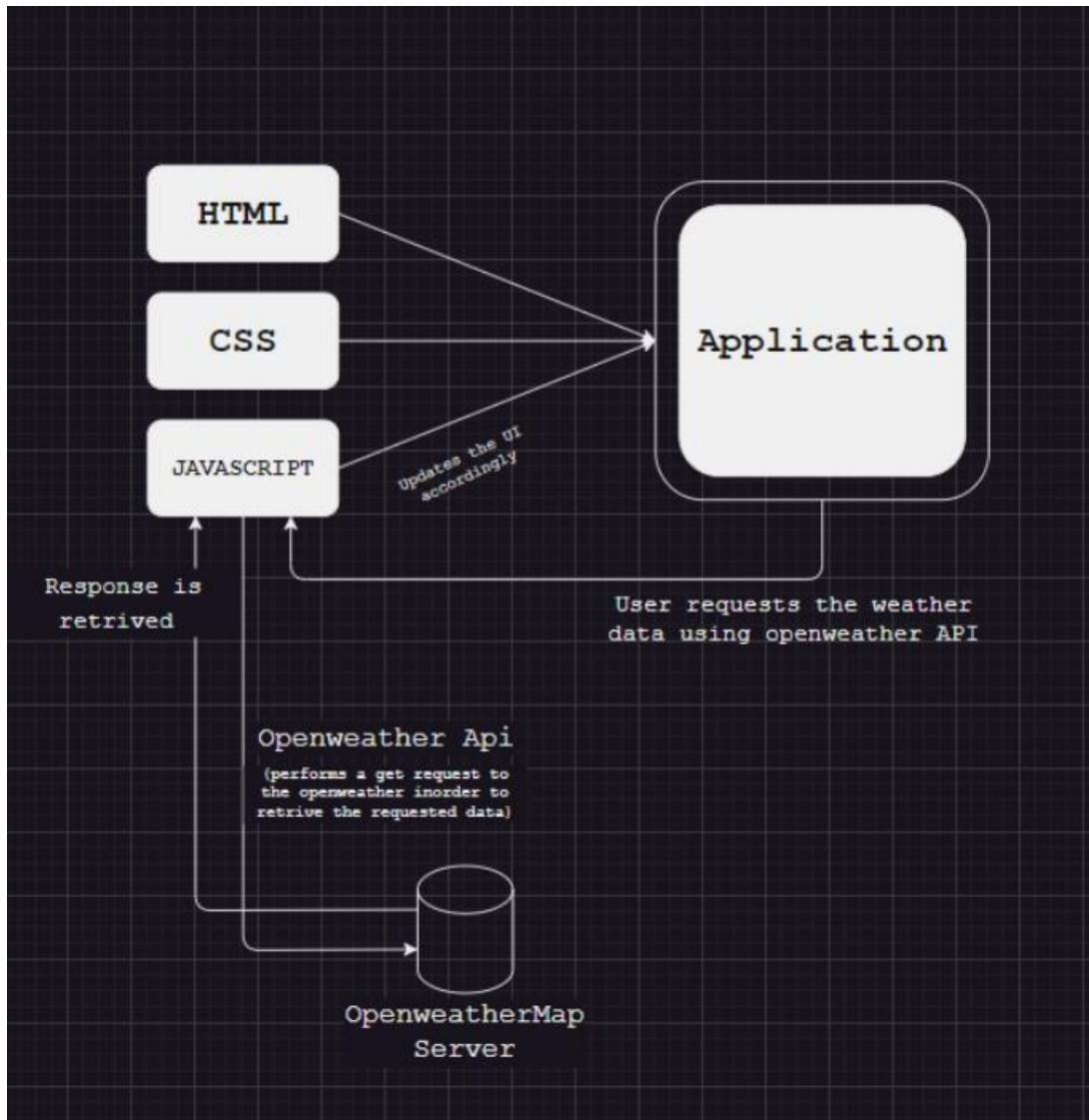
Every weather app has different methods of gathering information. Some apps depend upon government satellite while some use their own technology and team of meteorologist. Most commonly used tools are weather satellite and Doppler radar.

Our weather app can produce the temperature and other weather conditions of any city without any location accessing. Our app can work on any type of system whether it is updated or outdated.

Chapter 3

THEORITICAL ANALYSIS

3.1 Block diagram Diagrammatic overview of the project.



As you can see in the diagram above, a standard client-server architecture has three parts:

- **Web service repository:** This is a library of web services built to serve external requests for information. The served information is usually a little piece of information, like a number, a word, some variables, etc. For example, a flight number, a package tracking

number, the status of an order (one letter), etc. This library is usually documented in great detail since external applications will call the functions it contains.

- **Web service controller:** This module communicates the information in the web service repository with the service requesters. When an external service requester calls a certain function from the web service repository, the web service controller interprets the call and looks for the function in the web server repository. Then it executes the function and returns a value to the requester.
- **Database server:** This server contains the tables, indexes, and data managed by the core application. Searches and insert/delete/update operations are executed here.
- **Service requesters:** These are external applications that request services from the web service repository through the internet, such as an organization requesting flight information from an airline or another company asking the package carrier for the location of a package at a given moment.

3.2 Hardware / Software designing Hardware and software requirements of the project

HARDWARE AND SOFTWARE DESIGNING

Android mobile with a minimum version 2.2.

The processor is not less than 500MHZ.

RAM > 170mb.

SD card with a minimum of 512 MB.

Resolution is not less than 480*800pixs.

Recommended System Requirements

Often manufacturers of games will provide the consumer with a set of requirements that are different from those that are needed to run a software. These requirements are usually called the recommended requirements. These requirements are almost always of a significantly higher level than the minimum requirements, and represent the ideal situation in which to run the

software. Generally speaking, this is a better guideline than minimum system requirements in order to have a fully usable and enjoyable experience with that software.

Hardware Requirements

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatible, and sometimes incompatible hardware devices for a particular operating system or application. The following sub-sections discuss the various aspects of hardware requirements.

Architecture

All computer operating systems are designed for a particular computer architecture. Most software applications are limited to particular operating systems running on particular architectures. Although architecture-independent operating systems and applications exist, most need to be recompiled to run on a new architecture. See also a list of common operating systems and their supporting architectures.

Processing power

The power of the central processing unit (CPU) is a fundamental system requirement for any software. Most software running on x86 architecture define processing power as the model and the clock speed of the CPU. Many other features of a CPU that influence its speed and power, like bus speed, cache, and MIPS are often ignored. This definition of power is often erroneous, as different makes and models of CPUs at similar clock speed often have different throughput speeds.

Memory

All software, when run, resides in the random access memory (RAM) of a computer. Memory requirements are defined after considering demands of the application, operating system, supporting software and files, and other running processes. Optimal performance of other unrelated software running on a multi-tasking computer system is also considered when defining this requirement.

Secondary storage

Data storage device requirements vary, depending on the size of software installation, temporary files created and maintained while installing or running the software, and possible use of swap space (if RAM is insufficient).

Display adapter

Software requiring a better than average computer graphics display, like graphics editors and high-end games, often define high-end display adapters in the system requirements.

Peripherals

Some software applications need to make extensive and/or special use of some peripherals, demanding the higher performance or functionality of such peripherals. Such peripherals include CD-ROM drives, keyboards, pointing devices, network devices, etc.

Software Requirements

Software requirements deal with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or prerequisites are generally not included in the software installation package and need to be installed separately before the software is installed.

Platform

A computing platform describes some sort of framework, either in hardware or software, which allows software to run.^[2] Typical platforms include a computer's architecture, operating system, or programming languages and their runtime libraries.

Operating system is one of the requirements mentioned when defining system requirements (software). Software may not be compatible with different versions of same line of operating systems, although some measure of backward compatibility is often maintained. For example, most software designed for Microsoft Windows XP does not run on Microsoft Windows 98, although the converse is not always true. Similarly, software designed using newer features of

Linux Kernel v2.6 generally does not run or compile properly (or at all) on Linux distributions using Kernel v2.2 or v2.4.

APIs and drivers

Software making extensive use of special hardware devices, like high-end display adapters, needs special API or newer device drivers. A good example is DirectX, which is a collection of APIs for handling tasks related to multimedia, especially game programming, on Microsoft platforms.

Web browser

Most web applications and software depend heavily on web technologies to make use of the default browser installed on the system. Microsoft Edge is a frequent choice of software running on Microsoft Windows, which makes use of ActiveX controls, despite their vulnerabilities.

OTHER REQUIREMENTS

Some software also has other requirements for performance. Internet connection (type and speed) and resolution of the display screen are notable examples.

Languages used

HTML

The HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

CSS

CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It

can also be used with any kind of XML documents including plain XML, SVG and XUL. CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

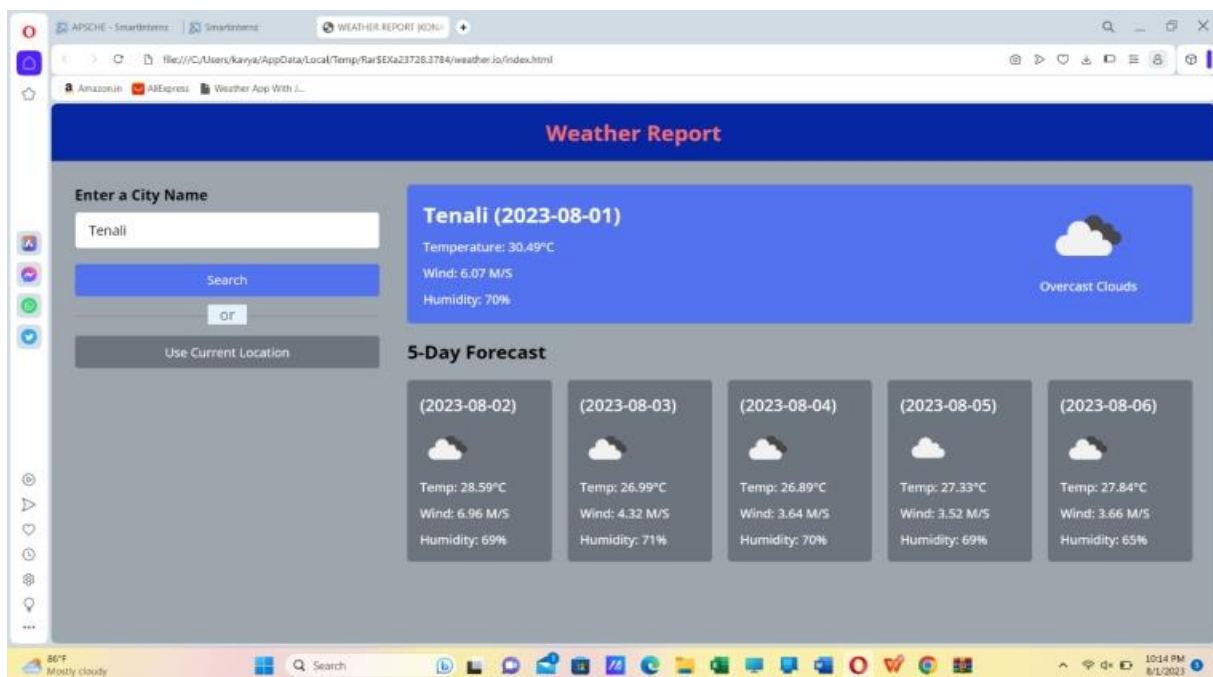
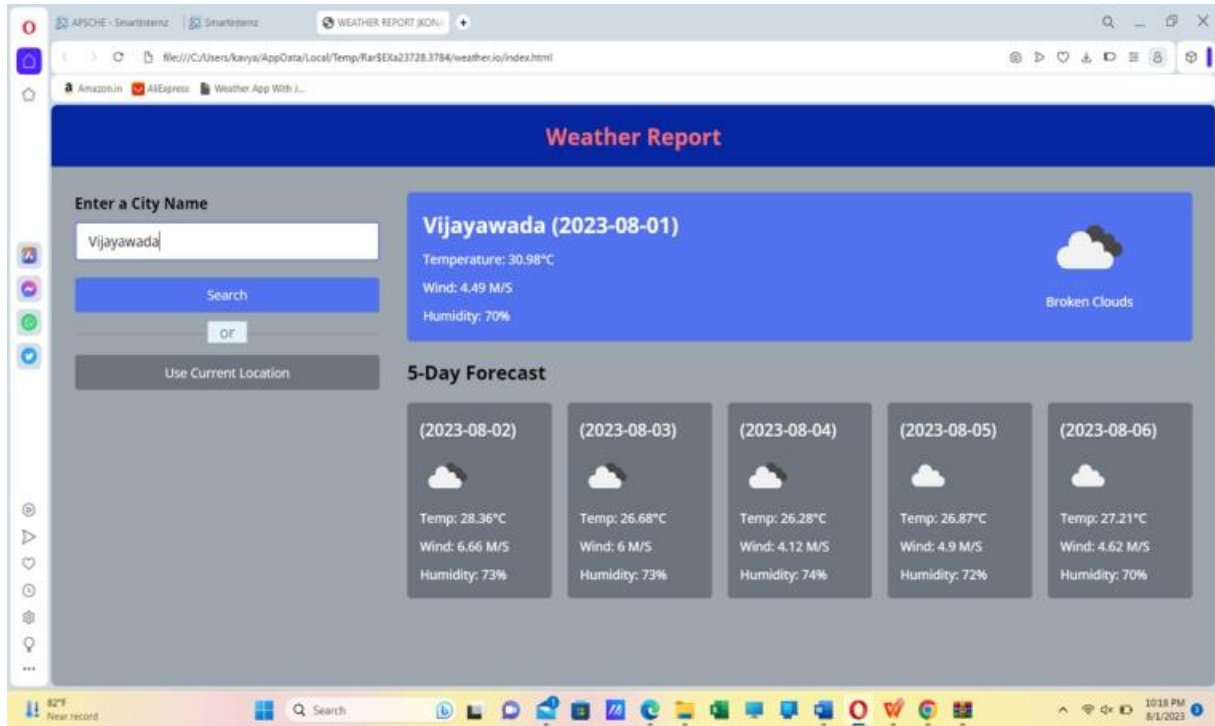
JavaScript

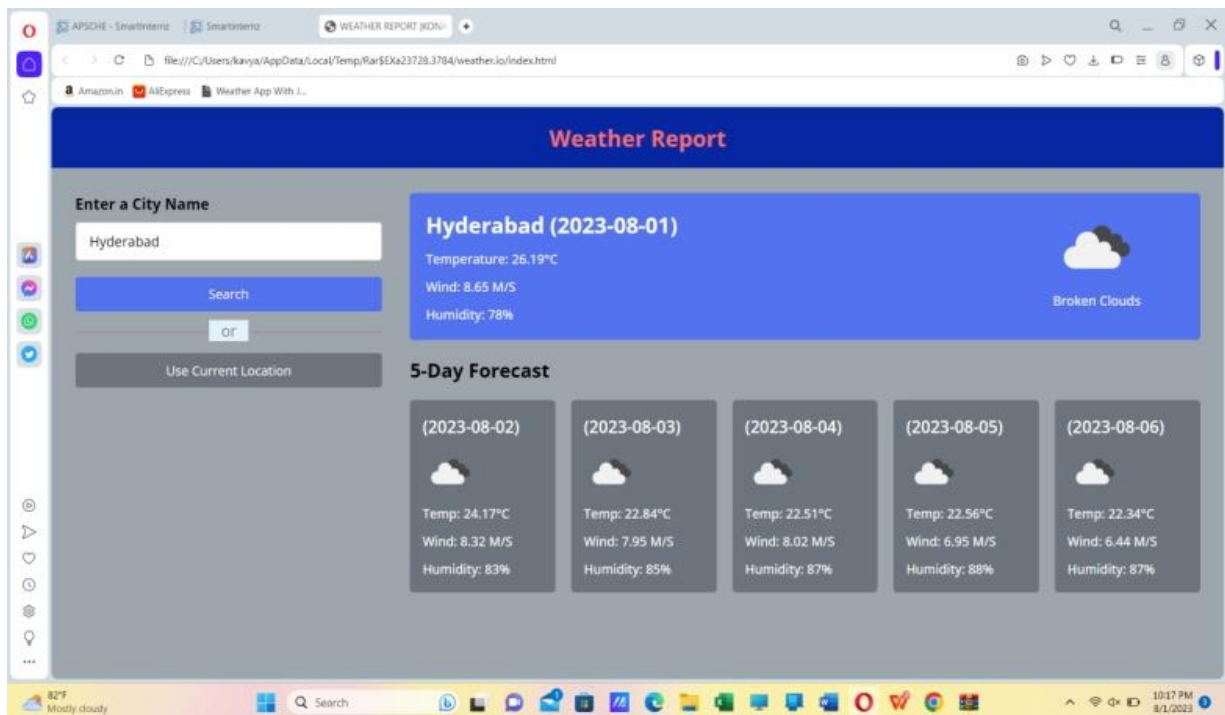
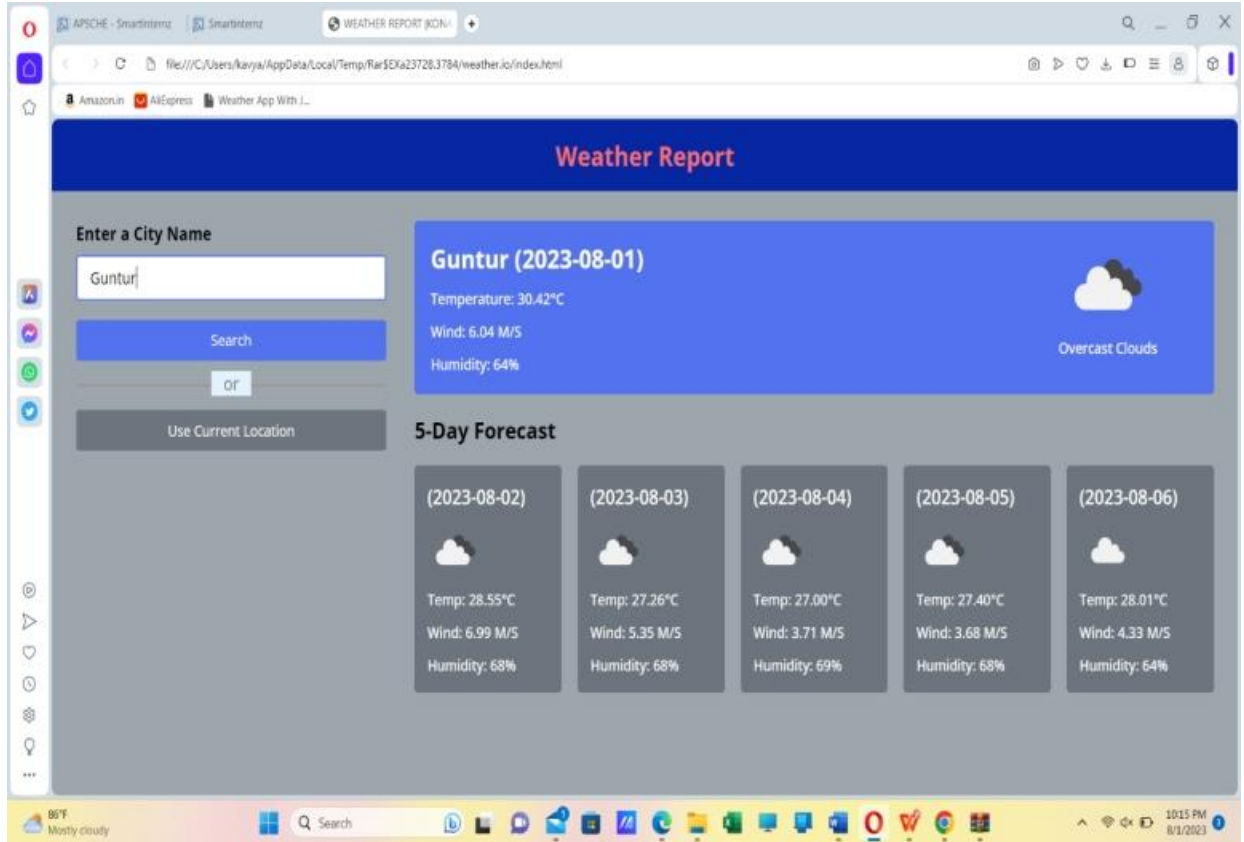
JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

Chapter 4

RESULT

4.1 Final findings (Output) of the project along with screenshots.





Chapter 5

ADVANTAGES & DISADVANTAGES

5.1 List of advantages and disadvantages of the proposed solution

ADVANTAGES:

- Farmers can know when to plant or harvest their crops.
- People can choose where and when to take their holidays to take advantages of good weather.
- Surfers know when large waves are expected.
- Regions can be evacuated if hurricanes or floods are expected.
- Instant information availability. Improved Weather forecast. Easy Flow of Information. Widget Support.
- Aircraft and shipping rely heavily on accurate weather forecasting.
- Real time data is one of the biggest advantages of weather monitoring system.
- The reasons why people have been going in for weather stations is because of the ability to get their information in real-time.
- Real-time alerts are other important advantages of weather monitoring system.
- These real-time alerts can be informed to you, and your family members about the weather, increasing survival, being safe for everyone.

DISADVANTAGES:

- Weather is extremely difficult to forecast correctly
- It is expensive to monitor-so many variables from so many sources
- The computers needed to perform the millions of calculations necessary are expensive
- The weather forecasters get blamed if the weather is different from the forecast
- Weather is extremely difficult to forecast correctly
- It is expensive to monitor so many variables from many sources.

- the computers needed to perform the millions of calculations necessary are expensive
- The weather forecasters get blamed if the weather is different from the forecast
- Confusing Terminology: The terminology used in weather forecasting can be confusing, making it difficult for some people to understand the predictions.
- Reliance on Technology: Weather forecasting relies heavily on technology, and if the technology fails or is unavailable, accurate predictions cannot be made.
- Limited Reach: 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.

Chapter 6

APPLICATIONS

6.1 The areas where this solution can be applied

The most common application of weather apps is for personal use. People use these apps to check the weather forecast before planning their outdoor activities, trips, or daily routines.

Weather apps help users decide what to wear, whether to carry an umbrella, or if it's safe to engage in outdoor sports.

Travelers use weather apps to plan their trips more effectively.

When organizing outdoor events like weddings, parties and contingency plans in case of adverse weather conditions.

Farmers and agriculturalists use weather apps to monitor weather patterns, temperature fluctuations, and precipitation forecasts.

Construction teams use weather apps to schedule tasks, ensure worker safety, and manage project timelines.

Pilots and aviation professionals use weather apps to monitor weather conditions along flight paths and at airports.

This information is crucial for flight planning, route adjustments, and ensuring passenger safety.

These are just a few examples of the many applications of weather apps across different industries and daily life scenarios.

Chapter 7

CONCLUSION

7.1 Conclusion summarizing the entire work and findings.

The weather app provides a convenient and accessible way for users to obtain up-to-date weather information for their desired locations. By utilizing various data sources and possibly incorporating advanced forecasting models, the app can offer accurate and reliable weather forecasts, current conditions, and other relevant meteorological data.

In conclusion, the weather app is a valuable tool for users to plan their activities, make informed decisions, and stay prepared for changing weather conditions. Its user-friendly interface and comprehensive features contribute to its usability and effectiveness.

However, to maintain its relevance and usefulness, the app should continuously update its data sources, improve its algorithms, and consider incorporating user feedback to enhance its overall performance and user experience.

Chapter 8

FUTURE SCOPE

Enhancements that can be made in the future.

The future scope of weather apps is exciting and full of potential, driven by advancements in technology, data availability, and user demands.

Weather apps have already come a long way, providing real-time weather forecasts, radar maps, and other valuable information. However, the future scope of weather apps continues to be promising and exciting, driven by advancements in technology and data analytics.

As technology and our understanding of weather patterns continue to evolve, the future of weather apps might hold even more surprises and innovative features.

The future scope of weather apps is vast and interconnected with various technological advancements. As more data becomes available and as user needs evolve, weather apps have the potential to become indispensable tools for individuals, communities, and even industries that rely on accurate weather information.