|  |  |  |
| --- | --- | --- |
| **Tech Saksham**  Final Project Report  **Track Name** |  |  |

**“WEATHER APP”**

**“CMR ENGINEERING COLLEGE”**

|  |  |
| --- | --- |
| **ROLL NO** | **NAME** |
| 208R1A0466 | ANUGU SWETHA |
| 208R1A0471 | BELLALA SRIJA |
| 208R1A0495 | KUMBA PRAVALIKA |
| 208R1A04A2 | NARRA SATHVIKA |

|  |  |
| --- | --- |
|  |  |
|  | Trainer Name |
|  | Ankit Dixit |

**ABSTRACT**

Weather forecasting is the prediction of the state of the atmosphere for a given location using the application of science and technology. This includes temperature, rain, cloudiness, wind speed, and humidity. Weather warnings are a special kind of short-range forecast carried out for the protection of human life. Weather warnings are issued by the governments throughout the world for all kinds of threatening weather events including tropical storms and tropical cyclones depending upon the location. The forecast may be short-range or Long-range. It is a very interesting and challenging task.

This report provides a basic understanding of the purpose and scope of weather forecasts, the basic principles and the general models developed for forecasting.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Table of Contents** | **Page No.** |
| 1 | Chapter 1: Introduction | 1 |
| 2 | Chapter 2: Services and Tools Required | 5 |
| 3 | Chapter 3: Project Architecture | 7 |
| 4 | Chapter 4: Architecture Blocks Detail Working | 8 |
| 5 | Chapter 5: Project Budget | 9 |
| 6 | Conclusion | 10 |
| 7 | References | 11 |
| 8 | Code | 12 |

**CHAPTER 1**

**INTRODUCTION**

* 1. **Overview**

Creating a weather forecast application using HTML, CSS, and JavaScript allows developers to build a user-friendly interface for accessing up-to-date weather information. In such an application, HTML forms the structural foundation, CSS adds styling and layout, while JavaScript interacts with APIs to fetch and display real-time weather data. Users can enter a location, and upon submission, JavaScript triggers an API request to retrieve weather details, including temperature, conditions, and forecasts. These details are then dynamically populated into the HTML elements, providing a visually appealing and informative weather forecast. Overall, this combination of technologies enables developers to create a dynamic and interactive weather forecast application that enhances user experience and keeps them informed about the weather conditions in their chosen location.

**1.2Feature**

* Location-Based Forecast: Allow users to enter or automatically detect their location to provide weather forecasts specific to their area.
* Current Conditions: Display the current temperature, weather conditions (e.g., sunny, rainy, cloudy), and real-time data such as wind speed and humidity.
* Hourly Forecast: Provide an hourly breakdown of weather conditions for the current day, including temperature, precipitation, and wind details.
* Extended Forecast: Offer a multi-day forecast, typically up to 7-10 days, so users can plan ahead.
* Weather Maps: Include interactive weather maps that allow users to explore radar images, satellite imagery, and other meteorological data.
* Weather Alerts: Push notifications or alerts for severe weather conditions such as storms, hurricanes, or extreme temperatures.
  1. **Advantages**
* Accessibility: HTML, CSS, and JS can be used to create web applications that are accessible to a wide range of users, including those with disabilities. This ensures that weather information is available to everyone.
* Real-time Updates: JavaScript allows you to fetch and display real-time weather data from APIs, providing users with up-to-date information. This is crucial for accurate weather forecasts.
* User-Friendly Interface: With HTML and CSS, you can design a user-friendly and visually appealing interface for displaying weather information. This can include interactive maps, charts, and animations to make the data more engaging and understandable.
* Cross-Platform Compatibility: Web-based weather forecasts can be accessed on various devices and platforms, including desktop computers, tablets, and smartphones, without the need for platform-specific development.
* Integration with Location Services: JavaScript can access the user's location, allowing for location-based weather forecasts. This provides personalized weather information based on the user's current location.
  1. **Scope**
* Geographical Coverage: Determine the geographical area you want to cover with your weather forecast. It could be a local city, a region, a country, or even a global forecast service.
* Weather Parameters: Specify the weather parameters you want to forecast, such as temperature, precipitation, wind speed, humidity, UV index, air quality, etc. Different users may have varying needs.
* Time Horizons: Decide the time horizons you want to provide forecasts for. This can include short-term (e.g., hourly or daily), medium-term, and long-term forecasts.
* Interactive Features: Determine if you want to include interactive features like map-based visualization, radar and satellite imagery, historical weather data, or user-generated content such as weather reports or photos.
* Data Sources: Identify the data sources you'll use for weather information. This might include government weather agencies, private weather data providers, or a combination of sources. Ensure the data is reliable and regularly updated**.**
  1. **Future Work**
* Improved Forecast Models: Continuously work on improving the accuracy and precision of weather forecasts by refining the underlying prediction models. Incorporate advanced meteorological techniques, machine learning, and AI algorithms for better predictions.
* High-Resolution Forecasting: Invest in high-resolution modeling to provide more detailed and localized forecasts. This is especially important for users in areas with diverse microclimates.
* Severe Weather Detection: Enhance the system to detect and predict severe weather events such as hurricanes, tornadoes, floods, and wildfires, and provide timely alerts and notifications.
* Weather Data Sources: Explore additional data sources, such as weather station networks, IoT devices, and crowd-sourced weather data, to improve data quality and coverage.
* Historical Data Analysis: Enable users to access historical weather data for research, trend analysis, and climate studies. Historical data can be valuable for various industries, including agriculture, insurance, and urban planning.

**CHAPTER 2**

**SERVICES AND TOOLS REQUIRED**

**2.1 Services Used**

* Weather Stations: Ground-based weather stations provide real-time data on temperature, humidity, wind speed and direction, air pressure, and precipitation at specific locations.
* Weather Balloons (Radiosondes): These instruments are launched into the atmosphere to collect data on temperature, humidity, and atmospheric pressure at different altitudes.
* Satellite Imagery: Weather satellites orbit the Earth, capturing images and data on cloud cover, sea surface temperatures, and other atmospheric parameters.
* Radar Systems: Doppler radar systems are used to detect precipitation, storm movement, and intensity, providing critical data for severe weather warnings.

**2.2 Tools and Softwares used**

* Radarscope: A tool used by meteorologists to analyze radar data for precipitation, storm tracking, and severe weather detection.
* NOAA Weather Satellite Imagery: Various software tools and websites provide access to satellite imagery for monitoring weather patterns.
* Weather Data APIs:
* OpenWeatherMap API: Provides access to current weather data, forecasts, and historical weather information.
* Weather APIs from National Meteorological Services: Many countries offer APIs for accessing official weather data.

**2.2.2 HTML**

This HTML template provides a basic structure for describing your weather forecast project, including information about the project's purpose, key features, how to use it, and how users can get started. You can further enhance and style the HTML content to make it visually appealing and informative for your target audience. Additionally, you can link the "sign up" and "download" anchors to your project's registration or app download pages as needed.

**CHAPTER 3**

**PROJECT ARCHITECTURE**

**3.1 Architecture**

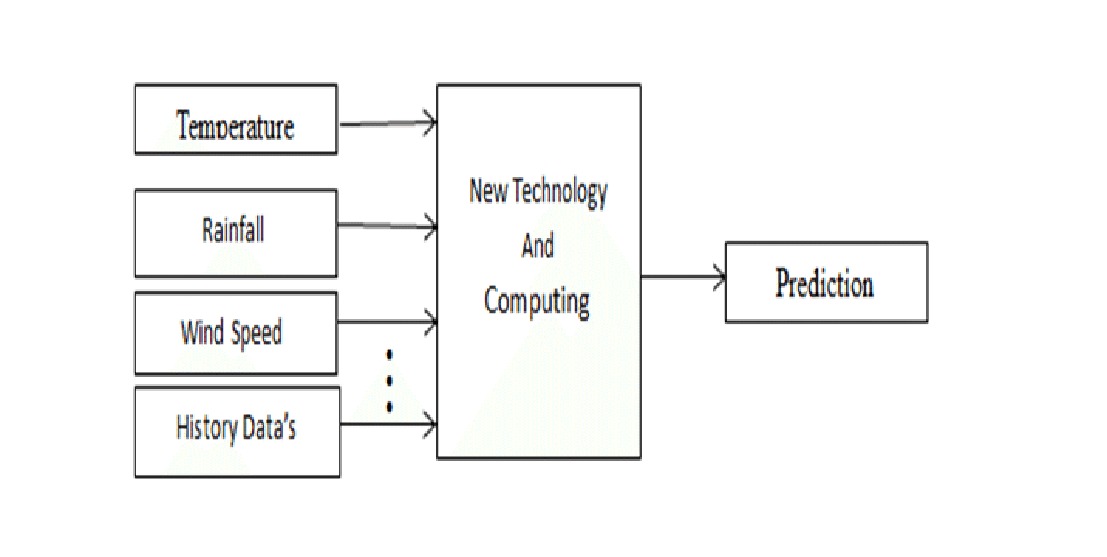
**USER FRONTEND**

|  |  |  |
| --- | --- | --- |
|  | **HTML 5** |  |

**CHAPTER 4**

**ARCHITECTURE BLOCKS DETAIL WORKING**

**4.1 Blocks**



**CHAPTER 5**

**PROJECT BUDGET**

It won’t cost anything for our project. But if anyone want to use this project for commercial purpose the budget could be as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No** | **Cloud Services and Coding Cost** | **Single Price (Rs)** | **Total** |
| 1 | Domain cost | 600/-(approx.) | 600/- |
| 2 | Web hosting | 200/- (approx.) | 200/- |
| 3 | Maintance | 500/- (approx.) | 500/- |
| Total | | | 1300/- |

**CONCLUSION**

In conclusion, a weather forecast project plays a crucial role in our daily lives, offering invaluable insights into the ever-changing atmospheric conditions that impact our activities and decisions. As technology advances and our understanding of meteorology deepens, weather forecasting continues to improve in accuracy and accessibility.

Weather forecasting projects bring together a diverse set of tools, technologies, and data sources, ranging from advanced numerical weather prediction models to user-friendly web and mobile applications. These projects empower individuals, industries, and governments to make informed choices, plan for weather-related events, and respond effectively to severe weather threats.

Moreover, weather forecasts are essential for various sectors, including agriculture, transportation, emergency management, renewable energy, and more. They aid in optimizing resource allocation, minimizing risks, and supporting sustainable practices.

**REFERENCES**

<https://public.wmo.int/>

<https://www.ecmwf.int/>

<https://www.weather.gov/>

<https://www.noaa.gov/>

**CODE**

[**https://github.com/AnuguSwetha/weather**](https://github.com/AnuguSwetha/weather)