Project Weather Forecasting Web Application

Abstract

This project focuses on the development of a Weather Forecasting Web Application using HTML, CSS, Bootstrap, and the OpenWeather API. The application is designed to be fully responsive and works seamlessly across all devices, including desktops, tablets, and mobile phones.

The output of the project provides real-time weather information for a selected city, including current temperature, weather conditions, feels-like temperature, humidity, atmospheric pressure, and wind speed. The application also displays Air Quality Index (AQI) data, such as CO, SO_2, O_3, and NO_2, to give users additional environmental insights.

Further features include sunrise and sunset timings, current date and time, hourly weather updates, and a five-day weather forecast, all dynamically generated using API responses. JavaScript embedded within the HTML file handles data fetching and updates the user interface without requiring page reloads.

This project demonstrates the practical use of responsive web design, API integration, and client-side scripting. The application is deployed using GitHub Pages, making it accessible online for demonstration and educational purposes.

Description of OpenWeather API

OpenWeather *is a popular online platform that provides* ***real-time, forecast, and historical weather data*** *through web-based* ***Application Programming Interfaces (APIs)****. These APIs allow developers to access accurate weather information for any location in the world by sending HTTP requests and receiving responses in* ***JSON format****.*

OpenWeather APIs are widely used in ***web applications, mobile applications, and IoT systems*** *to display live weather conditions such as temperature, humidity, wind speed, atmospheric pressure, air quality, and weather forecasts.*

How to Create an OpenWeather API Key (API ID)

Step 1: Open the OpenWeather Website

Visit the official OpenWeather website:

👉 ***https://openweathermap.org***

Step 2: Sign Up / Log In

- *Click on* ***Sign Up*** *if you are a new user*
- *Enter:*
  - *Email address*
  - *Username*
  - *Password*
- *Verify your email*
- *If you already have an account, click* ***Sign In***

Step 3: Generate API Key (API ID)

1. *After logging in, go to your* ***Account Dashboard***
2. *Open the* ***API Keys*** *section*

3. *A **default API key** will be generated automatically*
4. *You can also create a new API key by giving it a name (e.g., Weather Project)*

*This API key (also called **API ID**) is used to authenticate requests made from your application.*

Project Output Summary

- *Displays real-time weather data for any searched city*
- *Shows temperature, feels-like temperature, and weather condition*
- *Displays humidity, pressure, and wind speed*
- *Provides Air Quality Index (AQI) details*
- *Shows sunrise and sunset timings*
- *Displays hourly weather forecast*
- *Provides a five-day weather forecast*
- *Responsive design compatible with all devices*

Learning Outcomes

- *Understand the fundamentals of web application development using HTML, CSS, and Bootstrap for structuring and styling responsive user interfaces.*
- *Implement responsive web design principles to ensure the application functions effectively across multiple devices and screen sizes.*
- *Integrate third-party RESTful APIs, specifically the OpenWeather API, to retrieve and display real-time weather and air quality data.*

- *Apply asynchronous programming concepts in JavaScript to handle API requests and dynamically update content without page reloads.*
- *Manipulate the Document Object Model (DOM) to render live data such as temperature, humidity, wind speed, AQI, and forecasts.*
- *Analyze and process JSON data received from external APIs and convert it into meaningful visual output for end users.*
- *Design user-friendly dashboards that present complex data in a clear and organized manner.*
- *Identify and debug frontend issues related to API integration, data handling, and UI rendering.*
- *Demonstrate practical understanding of real-time web applications, bridging theoretical knowledge with hands-on implementation.*