Weather Forecast Application

Name : Udugundla Praveen
Email: upraveen776@gmail.com
GitHub: https://github.com/Praveen775D/Weather_Forecast_Project

INDEX (Table of Contents)

1. Introduction

Weather information plays an important role in daily life for planning travel, work, and outdoor activities.
This project focuses on developing a Weather Forecast Application using JavaScript, HTML, and Tailwind CSS that provides accurate and real-time weather information in a simple and user-friendly interface.

---

2. Objective of the Project

The main objectives of this project are:

- To fetch real-time weather data using a weather API
- To display current and extended weather forecasts
- To allow users to search weather by location or current GPS position
- To design a responsive and interactive user interface
- To follow proper Git version control practices

---

3. Project Overview

The Weather Forecast Application allows users to:

- Search weather using city name, pincode, or village
- Get current location weather using GPS
- View today’s weather details such as temperature, humidity, and wind speed
- View a 5-day extended weather forecast
- Experience dynamic UI changes based on weather conditions

---

4. Technologies Used

Technology	Purpose
HTML5	Structure of the application
Tailwind CSS	Styling and responsive design
JavaScript	Logic and API integration
OpenWeatherMap API	Fetching weather data
Leaflet.js	Map-based weather interaction
Git & GitHub	Version control

---

5. Project Structure

```
Weather_Forecast_Project/
│
├── index.html
├── src/
│   ├── css/
│   │   └── style.css
│   ├── js/
│   │   └── app.js
│
├── README.md
├── Weather_Forecast_Application_Documentation.pdf
└── .gitignore
```

---

6. Weather API Integration

- OpenWeatherMap API is used to fetch weather data.

- API provides:
  - Current weather details
  - Temperature, humidity, wind speed
  - Weather condition icons
- API calls are handled using JavaScript fetch().

---

7. User Interface Design

- UI is designed using Tailwind CSS
- Simple and clean layout
- Responsive design supports:
  - Desktop
  - iPad Mini
  - iPhone SE
- Dynamic background colors change based on weather conditions:
  - Sunny → warm gradient
  - Rainy → blue/rain theme
  - Cold → cool gradient

---

8. Features Implementation

Location Search

- Users can search weather using:
  - City name
  - Pincode
  - Village / District / State

Current Location Weather

- Uses browser GPS to detect user location
- Automatically fetches weather data

Temperature Unit Toggle

- Allows switching between °C and °F
- Applied only to today's temperature

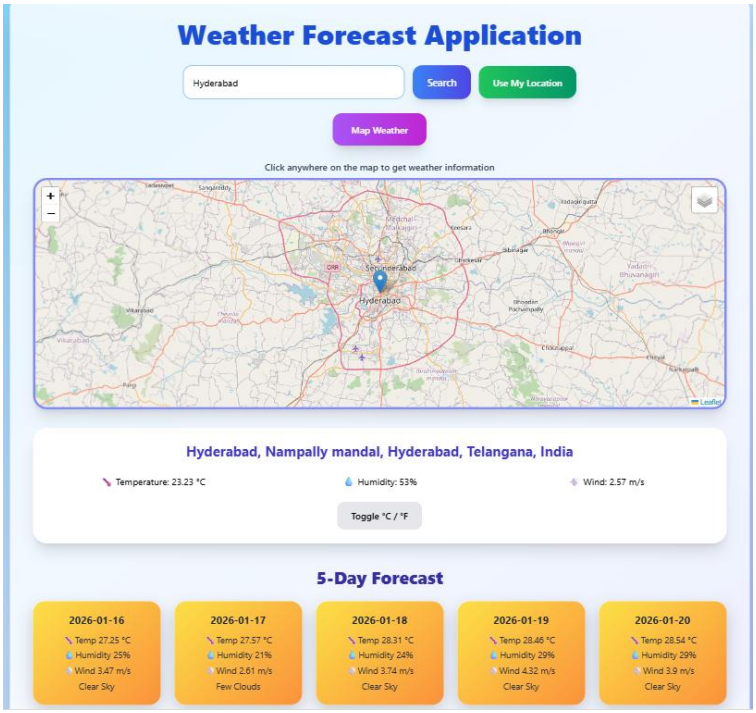
Recently Searched Locations

- Stored using local storage
- Dropdown appears only after first search
- Clicking a location updates weather instantly

---

9. Extended Forecast Module

- Displays 5-day weather forecast
- Each forecast card shows:
  - Date
  - Temperature
  - Wind speed
  - Humidity
  - Weather icon
- Cards are visually styled for better readability



---

10. Error Handling & Validation

- Handles:
  - Invalid location names
  - Empty search input
  - API errors
- Errors are shown clearly on UI
- JavaScript alert boxes are avoided

---

11. Version Control (Git Usage)

- Git is used throughout the project
- Meaningful commit messages are written
- Separate commits for:
  - HTML
  - CSS
  - JavaScript
  - README
- .gitignore is added to exclude unnecessary files

---

12. How to Run the Project

1. Clone or download the repository
2. Open src/js/app.js
3. Add your OpenWeatherMap API key
4. Open index.html in a browser
5. Search for a location or use map/GPS

---

13. Limitations

- Internet connection is required
- API request limits depend on OpenWeatherMap plan
- Offline mode is not supported

---

14. Future Enhancements

- Hourly weather forecast
- Dark mode support
- Weather notifications
- Offline caching using Service Workers
- User login and preferences

---

15. Conclusion

This Weather Forecast Application successfully meets the project objectives by providing real-time weather information with a responsive and interactive UI.
The project demonstrates strong understanding of JavaScript, API integration, UI design, and version control practices.
