🌤️ Weather App – Project Report

# 1. Project Overview

The Weather App is a client-side web application designed to provide users with real-time weather updates for cities worldwide. Developed using HTML, CSS, and JavaScript, the application integrates with the Open-Mateo API to access accurate weather data. It elegantly presents critical information, including temperature, wind speed, and current weather conditions, in a visually appealing format.

# 2. Objectives

- To develop a weather application utilising AI-assisted development tools.

- To enable users to access the current weather conditions of any designated city.

- To refine prompts by employing the TRACI framework.

- To utilise AI assistance for debugging and testing processes.

- To enhance skills in utilising public APIs and implementing asynchronous JavaScript.

- To generate fundamental test cases with guidance from AI.

- To initiate HTTP requests to the Open-Meteo Weather Forecast platform.

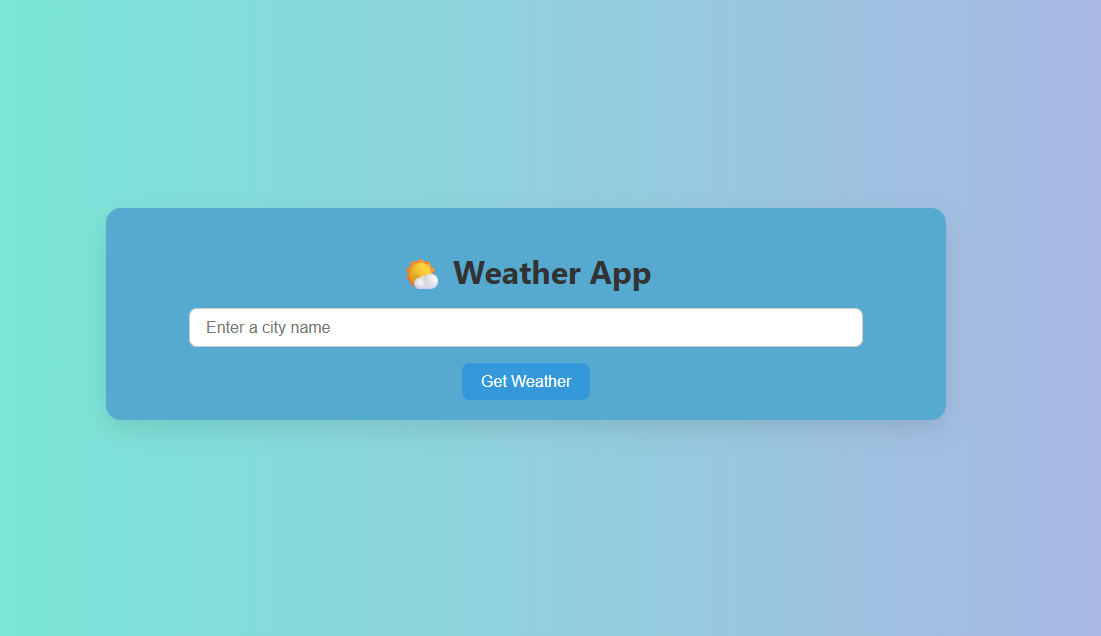
- To construct a responsive and interactive user interface.

# 3. Technologies Used

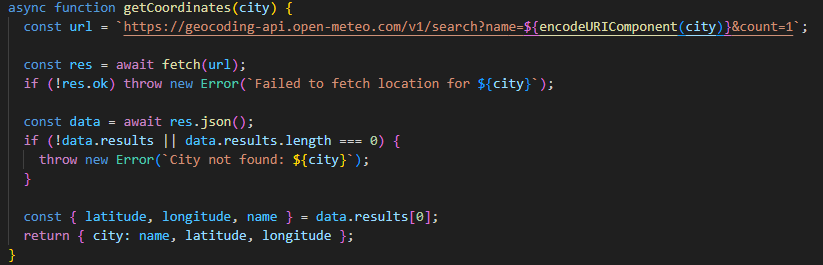
|  |  |
| --- | --- |
| Technology | Purpose |
| HTML | Structure of the web page |
| CSS | Styling and layout |
| JavaScript | API calls, data handling, interaction |
| Open-Meteo API | Geolocation and weather data |

# 4. How It Works

1. Input: The user enters the name of a city.



2. Geocoding: JavaScript uses the Geocoding API to find the city's latitude and longitude.

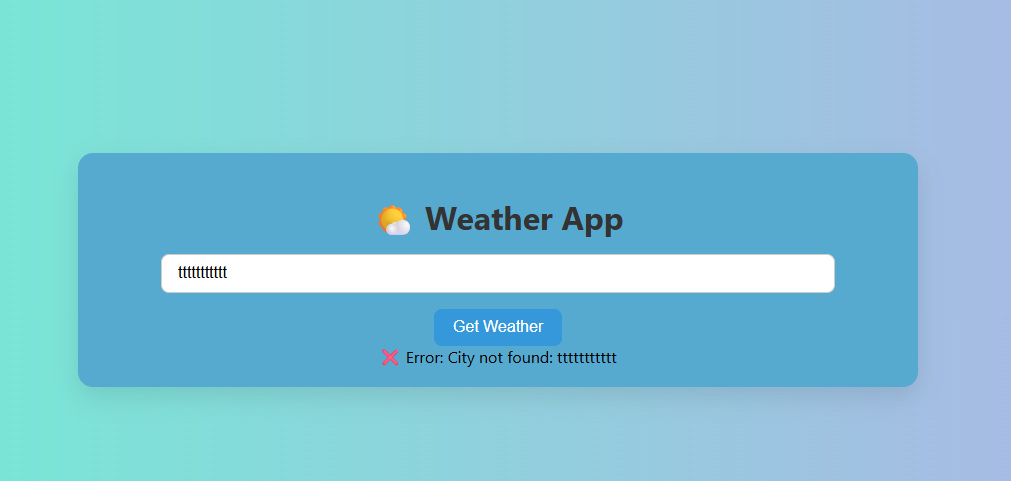
  
3. Weather Fetching: The app fetches current weather using the Forecast API.

  
4. Display: The result is displayed, including temperature and a corresponding weather description.



# 5. Core Features

- The user can input any valid city name.  
- Fetches and displays real-time temperature and conditions.  
- Handles API errors and invalid input gracefully.

  
- Clean, responsive design that works on both desktop and mobile.

# 6. Challenges Faced

While developing the weather app, I encountered several challenges related to working with APIs and JavaScript. One of the main difficulties was managing asynchronous operations — ensuring that results were displayed only after data from both the geolocation and weather APIs had been successfully retrieved.  
I also faced challenges in handling errors gracefully and maintaining user feedback during loading states. Additionally, I relied on AI assistance to debug and test various components, which helped improve functionality and reduce bugs during development.

# 7. Future Improvements

- Add a 3-day or 7-day forecast.  
- Include weather icons and background changes based on conditions.  
- Store recent search history.  
- Support multiple languages.

# 8. Conclusion

The Weather App successfully demonstrates the use of APIs, asynchronous JavaScript, and front-end development principles. It serves as a practical tool and a solid foundation for more complex weather-based applications.