**README FILE**

**Instructions for running the project:**

* Ensure all files are saved in the same folder. Then, open the project folder and double-click on the index.html file to view it in a browser.

**Approach:**

* Designed a well-structured HTML layout featuring a search form, a geolocation button, and a weather display section. Incorporated Bootstrap for responsiveness, ensuring seamless functionality on both desktop and mobile devices. Applied CSS styling to create a clean, modern, and user-friendly interface.
* Utilized the fetch() API to retrieve weather data based on the user-inputted city name. Integrated the Geolocation API to automatically obtain weather details for the user's current location. Implemented a temperature toggle feature, allowing users to switch between Celsius (°C) and Fahrenheit (°F).
* Displayed error messages for invalid city names or failed API requests.
* Handled geolocation permission denial gracefully, providing user feedback.

**Challenges Facedand Solutions:**

1. **Geolocation Not Working**

Problem:

* The browser blocked geolocation on insecure (HTTP) sites.
* Users denied location permissions, causing geolocation failure.

Solution:

* Ensured the app runs on HTTPS to allow geolocation.
* Improved error handling with specific messages for permission denial and location unavailability.

1. **Incorrect Weather Data for Location**

Problem:

* Sometimes, geolocation provided incorrect weather data or didn't update properly.

Solution:

* Used latitude & longitude instead of city name when fetching weather.
* Ensured real-time updates by re-fetching data when users clicked the "Use My Location" button.

1. **Handling API Errors (Invalid City, Network Issues, API Limits)**

Problem:

* Entering an invalid city name caused the app to break and API calls failed due to rate limits or incorrect API keys.

Solution:

* Implemented proper error handling using catch () to display user-friendly messages**:**

1. **Temperature Unit Toggle (Celsius & Fahrenheit)**

Problem:

* Users wanted to switch between °C and °F, but data was only fetched in one unit.

Solution:

* Implemented a unit toggle button that prefetches data in the selected unit and Used metric for °C and imperial for °F in the API request.