Job Simulation: Live Weather Dashboard

Project Title:Design and Implementation of a Live Weather Dashboard

Role:Front-End Development Intern

Technology Stack:React.js, Material UI (MUI), React Hooks (useState, useEffect)

Project Objective

The goal of this project is to develop a **responsive, single-page weather dashboard** that displays live weather information for multiple cities. The application will use React and Material UI to ensure a modern UI/UX experience. It introduces key front-end concepts such as **real-time data fetching**, **dynamic rendering**, and **state management** using React Hooks.

Task Summary

As a part of this simulation, interns are expected to:

- Build a single-page application (SPA) that interacts with a public weather API (e.g., Open-Meteo).
- Allow users to **select a city from a dropdown list**.
- Display **current weather data** such as temperature, weather condition, and city name.
- Dynamically **style the UI** based on weather conditions (e.g., hot, cold, cloudy).
- Handle asynchronous operations gracefully with loading indicators.

Functional Requirements

- Use useState to manage:
    - Selected city
    - Fetched weather data
    - Loading state
- Use useEffect to:
    - Fetch new weather data each time the selected city changes.
- Provide a **dropdown menu** with 3–5 pre-defined cities.
- Display the following weather details:
    - **City Name**
    - **Temperature**
    - **Weather Condition**
    - **Last Updated Time**
- Show a **loading spinner** (<CircularProgress>) while data is being fetched.

User Interface Requirements

Use **only Material UI (MUI)** components to build the entire interface:

- <Select> and <MenuItem>: City selector dropdown
- <Card>: To present weather details
- <Typography>: For text elements
- <Grid>: For responsive layout
- <CircularProgress>: While loading

Dynamic Styling (based on temperature or condition):

- **Red background**: Hot

- **Blue background**: Cold
- **Grey background**: Cloudy or overcast

All layout and styling should be done using the sx prop and MUI's responsive Grid system.

Project File Structure

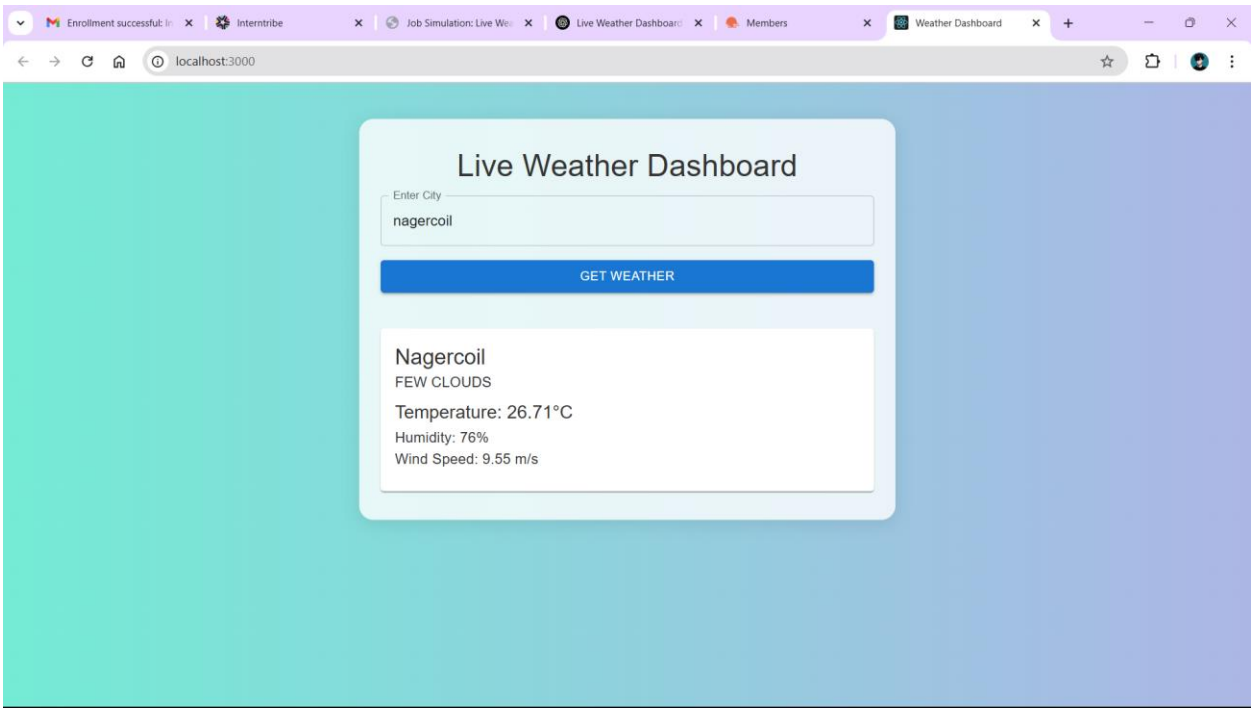
File	Purpose
App.jsx	Main component: layout, dropdown, manages state
WeatherCard.jsx	Reusable UI component to display weather data
api.js *(optional)*	Utility file for API requests

Learning Outcomes

By completing this simulation, you will gain experience in:

- Managing component state with useState
- Handling side effects and API calls with useEffect
- Creating clean, responsive layouts using MUI's Grid system
- Using conditional rendering and styling based on real-time data
- Structuring React applications for scalability and readability

Project


Enrollment successful: In
Interntribe
Job Simulation: Live Wea
Live Weather Dashboard
Members
Weather Dashboard
localhost:3000
Live Weather Dashboard
Enter City
nagercoil
GET WEATHER
Nagercoil
FEW CLOUDS
Temperature: 26.71°C
Humidity: 76%
Wind Speed: 9.55 m/s