**Real-Time Weather Monitoring Application**

This application monitors weather conditions in real-time using the OpenWeatherMap API. It provides summarized insights, alerts, and visualizations for various cities in India.

**Table of Contents**

* [Features](#features)
* [Getting Started](#getting-started)
* [Prerequisites](#prerequisites)
* [Installation](#installation)
* [Usage](#usage)
* [Configuration](#configuration)
* [Testing](#testing)
* [Contributing](#contributing)
* [License](#license)

**Features**

* Fetch real-time weather data for multiple cities.
* Display temperature, weather conditions, and alerts.
* Generate historical weather summaries and visualizations using charts.
* User-configurable temperature thresholds for alerts.

**Getting Started**

**Prerequisites**

* Node.js (v14 or later)
* npm or yarn package manager

**Installation**

1. Clone the repository:

bash

Copy code

git clone https://github.com/yourusername/weather-monitoring-app.git

cd weather-monitoring-app

1. Install dependencies:

bash

Copy code

npm install

# or

yarn install

**Usage**

1. Sign up at [OpenWeatherMap](https://openweathermap.org/) and get your API key.
2. Create a .env file in the root directory of your project and add your API key:

bash

Copy code

REACT\_APP\_WEATHER\_API\_KEY=your\_api\_key\_here

1. Start the application:

bash

Copy code

npm start

# or

yarn start

**Configuration**

* You can configure the temperature threshold for alerts in the Settings component of the application.

**Testing**

Run the following command to execute tests:

bash

Copy code

npm test

# or

yarn test

**Contributing**

Feel free to fork the repository and submit pull requests for improvements or bug fixes.

**License**

This project is licensed under the MIT License.

**Project Structure**

Make sure your project has a clear structure. Here’s an example:

csharp

Copy code

weather-monitoring-app/

│

├── public/

│ ├── index.html

│ └── favicon.ico

│

├── src/

│ ├── components/

│ │ ├── WeatherDashboard.js

│ │ ├── WeatherSummary.js

│ │ ├── WeatherAlert.js

│ │ ├── WeatherChart.js

│ │ └── Settings.js

│ ├── services/

│ │ └── weatherService.js

│ ├── App.js

│ ├── index.js

│ └── App.css

│

├── .gitignore

├── package.json

├── README.md

└── yarn.lock / package-lock.json

**package.json Dependencies**

Make sure your package.json includes the necessary dependencies for your project. Here’s a sample:

json

Copy code

{

"name": "weather-monitoring-app",

"version": "1.0.0",

"private": true,

"dependencies": {

"axios": "^0.27.2",

"chart.js": "^3.7.0",

"react": "^17.0.2",

"react-chartjs-2": "^3.0.4",

"react-dom": "^17.0.2",

"react-scripts": "5.0.0"

},

"scripts": {

"start": "react-scripts start",

"build": "react-scripts build",

"test": "react-scripts test",

"eject": "react-scripts eject"

},

"eslintConfig": {

"extends": [

"react-app",

"react-app/jest"

]

},

"browserslist": {

"production": [

">0.2%",

"not dead",

"not op\_mini all"

],

"development": [

"last 1 chrome version",

"last 1 firefox version",

"last 1 safari version"

]

}

}

**Final Steps**

* **Run the Application**: Ensure you can run npm start or yarn start without issues. Test all features.
* **Check for Errors**: Look for any console errors or warnings.
* **Run Tests**: Use npm test or yarn test to ensure all tests pass.
* **Commit Your Code**: Make sure to commit your code regularly with meaningful messages. Once everything is working and the README is complete, push your changes to your GitHub repository.

bash

Copy code

git add .

git commit -m "Completed weather monitoring app with README"

git push origin main