RCC INSTITUTE OF INFORMATIONTECHNOLOGY

NIIT MINOR PROJECT

Prepared by

Team Leader- Sayantan Mondal

Project Member 1- Abhraneel Roy

Project Member 2- Devjit Chowdhury.

Project Member 3- Subhranil Saha

Project Member 4- Manali Mukherjee

Project Member 5-Basant kumar Shaw

Date- 06/08/24

INDEX

➢ Screenshots of the Webpage

I. Project overview

The React Weather Application is designed to be a user-friendly and interactive tool for accessing real-time weather data. It utilizes the OpenWeatherMap API to retrieve weather information for any location entered by the user. The application dynamically updates the user interface to display relevant weather details, including temperature, humidity, wind speed, and a visually appealing background image that changes based on the current weather conditions.

II. Description

The React Weather Application provides users with a convenient way to obtain up-to-date weather information. The app fetches data from the OpenWeatherMap API and presents it in a clear and intuitive interface. The dynamic background image feature enhances the user experience by visually representing the current weather conditions, making the application more engaging and informative.

III. Introduction

Weather forecasting plays a crucial role in our daily lives, helping us prepare for various weather conditions. With the advancements in technology, users can now access real-time weather data through various platforms. This project demonstrates the development of a simple yet effective weather application using React and the OpenWeatherMap API. The app is designed to be responsive and interactive, offering users accurate weather information with a visually appealing interface.

IV. Software Requirements

- **Operating System:** Windows/Mac/Linux
- **Node.js:** v14 or above
- **React:** v17 or above
- **Axios: v0.21 or above (for API requests)**
- **OpenWeatherMap API Key:** (Free or Paid subscription)
- **React Animated Weather:** v1.7 or above
- **CSS**: For styling the components

V. Specifications

- **API Integration:** The application uses the OpenWeatherMap API to fetch weather data.
- **Real-Time Updates:** The app updates weather information in real-time upon user input.
- **Responsive Design:** The app updates weather information in real-time upon user input.
- **Dynamic Background :** Background images change based on the current weather condition.
- **Time and Date Display;** The current time and date are displayed dynamically and update every second**.**

VI. Functional Approach

1. **State Management:** The app uses React's useState hook to manage the state of the weather data and the user's location input.
2. **API Call**: When the user presses "Enter" after inputting a location, an API call is made using Axios to retrieve weather data from OpenWeatherMap.
3. **Dynamic Rendering:** The app dynamically updates the UI based on the data received from the API, such as temperature, weather description, and background image.
4. **Continuous Time Update:** The app uses setInterval to update the displayed time every second.

VII. Decision Tree

1. **Location Input:** User inputs a location.
2. **API Request:** The app checks if the "Enter" key is pressed; if so, it triggers an API call.
3. **Weather Data Received:**
   - Yes: Update the weather details and background image.
   - No: Show default weather information or an error message.
4. **Display Information:** Render the location, temperature, weather description, date, and time.
5. **Time Update:** Continuously update the time every second.

VIII. Design & Implementation

- **Component Structure:** The app is built using a single main component . The structure is simple, with all UI elements handled within this component.
- **UI Design:** The interface is minimalistic, with a search bar at the top for location input, followed by weather details. The background image covers the entire screen, creating an immersive experience.
- **Weather Icons and Backgrounds:** Different weather conditions trigger different background images and icons, which are imported and used within the component based on the weather data.
- **Error Handling:** Basic error handling is implemented to manage invalid API responses or user input.

IX. Testing

1. Unit Testing

Each component of the app, including the state management and API call functions, was tested independently.

2. Integration Testing

The integration between the search functionality, API response, and UI updates was thoroughly tested to ensure smooth operation.

3. Cross-Browser Testing

The app was tested on different browsers to ensure compatibility.

4. Responsive Testing

The application was tested on various screen sizes to ensure it is fully responsive.

X. Usage

1. Setup:

- Ensure Node.js is installed on your system.
- Clone the repository and navigate to the project directory.
- Run “npm install” to install the required dependencies.

2. **Obtaining API Key:**

   Sign up on OpenWeatherMap to obtain a free API key.

3. **Running the Application:**

   - Replace the placeholder API key in the code with your actual API key.
   - Use the command “npm start” to run the application locally.
   - Open your browser and go to http://localhost:3000 to interact with the app.

4. **Using the App:**

   - Enter a city name in the input field and press "Enter".
   - The app will display the current weather information for the entered location.
   - The background will change according to the weather condition of the location.
   - The app will also display the current time and date, updating every second.

XI. Conclusion

The React Weather Application is a functional and visually appealing tool that allows users to check the weather in real-time for any location. By leveraging React and the OpenWeatherMap API, the application offers a seamless user experience with dynamic weather updates. The project demonstrates the effectiveness of React in building interactive and responsive web applications.

XII. References

- **React Documentation:** https://reactjs.org/docs/getting-started.html
- **OpenWeatherMap API Documentation:** https://openweathermap.org/api
- **Axios Documentation:** https://github.com/axios/axios
- **React Animated Weather:** https://www.npmjs.com/package/react-animated-weather

XIII. Screenshots of the Webpage