

I used ChatGPT to assist in understanding API integration, error handling, and UI implementation for my weather application. Below is a detailed interaction showcasing the prompts and responses relevant to my project. I coded all implementations in the project based on this guidance.

Prompts

Prompts and Responses

Q: How do I fetch weather data using the Open-Meteo API?

A: Use Axios to make GET requests to the Open-Meteo API. Here's an example:

```
``javascript
axios.get('https://api.open-meteo.com/v1/forecast', {
  params: {
    latitude: 40.7128,
    longitude: -74.0060,
    hourly: 'temperature_2m',
    daily: 'temperature_2m_max,temperature_2m_min',
    timezone: 'auto',
  },
}).then(response => {
  console.log(response.data);
});
```

Q: How can I display the weather data dynamically in React?

A: Create a state to store the API response and use React components to display it.

Example: `const [weatherData, setWeatherData] = useState(null);`

```
useEffect(() => {  
  axios.get(API_URL).then(response => setWeatherData(response.data));  
}, []);
```

Q: How do I handle errors if a user enters an invalid city name?

A: Add error handling using a try-catch block or `.catch()` with Axios.

```
Example: axios.get(API_URL)  
  .then(response => setWeatherData(response.data))  
  .catch(error => {  
    if (error.response.status === 404) {  
      setErrorMessage('City not found. Please enter a valid city name!');  
    }  
  });
```

Q: How can I toggle the hourly temperature view in my app?

A: Use a `useState` hook to manage the visibility of the hourly temperature section:

```
const [showHourly, setShowHourly] = useState(false);  
  
<button onClick={() => setShowHourly(!showHourly)}>  
  {showHourly ? 'Hide' : 'Show'} Hourly Temperature  
</button>  
  
{showHourly && <HourlyTemperature data={hourlyData} />}
```

Q: How do I make the UI visually appealing using Material-UI?

A: Use Material-UI components like Card, Typography, and Button for styling:

```
<Card>
```

```
  <Typography variant="h5">Weather Information</Typography>
```

```
  <Button variant="contained">Show Hourly Temperature</Button>
```

```
</Card>
```

Q: How can I give a link to ChatGPT so that all my previous prompts can be shown to others?

A: There's no direct feature in ChatGPT to generate a sharable link for your conversation history. However, you can manually copy and share the relevant prompts and responses into a document. Tools like Google Docs or Canva can help format it professionally, and you can then provide the link to that document.

Alternatively, summarize your prompts and responses into a GitHub README or Markdown file. For complete transparency, organize all relevant ChatGPT interactions in a document or PDF and upload it to a platform like Google Drive, GitHub, or any file-sharing service to share with others.