

# Weather Forecast App using React and Tailwind CSS

Create a weather forecast application interface like the one in the provided image using React and Tailwind CSS. The interface should display the current weather for a city along with a four-day forecast.

## Features:

### 1. Search Bar:

- A search bar at the top where users can enter the name of a city to retrieve weather data.

### 2. Current Weather Display:

- Display the selected city's name (e.g., *New York*).
- Show the current temperature (e.g., 17°C) and a corresponding weather icon (e.g., a sun icon for clear skies).
- Include a text description of the weather (e.g., *clear sky*).

### 3. Four-Day Forecast:

- Display weather icons and temperatures for the upcoming four days (e.g., Wednesday, Thursday, Friday, Saturday).
- Each day should have:
  - A weather icon (sunny, cloudy, partly cloudy, etc.).
  - The predicted temperature for that day.

## Functional Requirements:

- City Search: The interface should update the displayed weather data based on the city input. This can be done by simulating data (hardcoded) or using an external API like Open Weather Map.
- Responsive Design: The interface must be fully responsive to adapt to various screen sizes, including mobile, tablet, and desktop.

## Styling:

- Tailwind CSS must be used to style the entire interface.
- Use appropriate background colors and transparency (e.g., semi-transparent cards over a cloud background).

- Ensure the temperature numbers, city name, and day names are styled as per the image with appropriate font sizes and weights.

#### Technical Requirements:

1. ReactJS: The app should be structured as a React component.
2. Tailwind CSS: All styling should be handled using Tailwind CSS utility classes.
3. Icons: Use weather icons (e.g., sun, cloud, etc.) from a free library like Font Awesome or similar.

#### Bonus:

- Real API Integration: Fetch real-time weather data using the Open Weather Map API.
- Animation: Add animations for weather icons (e.g., clouds moving, sun glowing).

