**Reflective** **Report on weather app Architecture**

The architecture of the weather app is designed with simplicity and functionality in mind, focusing on providing users with a clean and intuitive interface for retrieving weather data. Below are the strengths and weaknesses of the app’s architecture: -

**Strengths:**

**1.**clear separation of concerns: the app maintains a clear separation between its frontend (HTML and CSS) the business logic (JavaScript), and data retrieval (Api calls). This ensures that each part of the codebase can be developed and maintained independently, making the app scalable and easier to update.

2.user-Friendly Interface: The app’s interfaces simple, with a minimalistic design that ensures ease of use. The input field and search button are intuitively placed, and the weather information is displayed in a visually appealing way with appropriate details such as humidity, pressure, wind, speed etc.

**Weaknesses:**

1. Error Handling: while the app does check for missing or invalid city names, the error handling can be improved. The user is only alerted with a generic error message in case of a failure. A more robust error-handling mechanism could provide users with more context, such as specific error code (e.g. invalid city name, network issues) to enhance the user experience.
2. No offline mode: the app relies on an internet connection to fetch weather data, which means it doesn’t work offline. Implementing an offline mode with cached data or fallback options would improve the app’s robustness and user experience.

In conclusion, while the app’s architecture is functional and simple, there are opportunities for improvement, especially in terms of error handling, security, and additional features, these areas and be addressed to enhance the app’s usability, security and overall performance.