Prototype 1

The architecture of my weather app, which uses the Open Weather API, has many strengths but also weaknesses that I found out during development and use of my weather app.

The strengths of my weather app are:

1. **Simplicity**: The app’s architecture is simple and easy to understand and maintain. The use of Open Weather API simplifies the process of getting the necessary weather data and display it in my webpage.
2. **Realtime data**: The use of Open Weather API ensures that the app provides real time weather information about the given location, enhancing the user experience.
3. **Modularity**: The architecture is modular with everything being put inside functions allowing for easy addition of new features and changing existing functions.
4. **User Interface**: The user interface of my weather app is very user friendly and simple to use while also being visually appealing, giving a great experience to the user.

The Weaknesses of my weather app are:

1. **Dependency on API**: The app is fully dependent on the Open Weather API so in case of any downtime of the API, slowdown of internet connection or just no internet connection, the app will work slower or just stop working completely.
2. **Limited Features**: The app in its current state is lacking a lot of useful information such as hourly forecasts and weekly forecasts, and the ability to view the weather of past days.
3. **Error Handling**: While error handling is implemented, it is implemented in a very basic form and might not be capable of handling more advanced or complex errors.
4. **Performance**: The performance of the app isn’t fully optimized and depending on the user’s internet connection or many other factors the app could become sluggish and slow to show the weather information.

# Prototype 2

The architecture of my weather app, which uses Open Weather API for fetching data and mysql and php for server side caching of the data has many strengths and weaknesses that I found out during development and use of my weather app.

The strengths of my weather app are:

1. **Simplicity**: The app’s architecture is simple and easy to understand and maintain. The use of Open Weather API simplifies the process of getting the necessary weather data and display it in my webpage.
2. **Modularity**: The architecture is modular with everything being put inside functions allowing for easy addition of new features and changing existing functions.
3. **User Interface**: The user interface of my weather app is very user friendly and simple to use while also being visually appealing, giving a great experience to the user.
4. **Caching / Less Dependency on API**: The website now stores the weather information in a database so that it doesn’t need to rely on the API all the time.

The weaknesses of my weather app are:

1. **Error Handling**: While the app handles more errors now, it is still not handling every error in the best way possible so there is room for improvement.
2. **Performance**: The performance of the app is still not fully optimized and depending on the users internet connection the loading of weather data can be slow and sluggish.
3. **Internet Reliance**: The weather app still relies on internet and without an internet connection it doesn’t work.
4. **Limited Features**: Even though the app now shows weather information of the past 7 days, it only does so for the default city so if the user searches for any other city, the app won’t show the 7 days data.