

CM3110 – Mobile App Development

Coursework: *Will the Sun Shine Again?* App Report

Weather API Service Choice

The *OpenWeatherMap* API service was selected for this coursework component as, from the example APIs provided, the JSON results provided best matched the intentions I had for what the app would display to the user. Alongside this, the increments of 3 hours over the 5 days worked well due to how easily each day/24-hour span could be identified as groups of 8 forecasts.

The API service itself restricts the amount of calls made using a free account but does not specify the number explicitly. However, for the purposes of this coursework it has been assumed that this will not cause any negative effects. It is recommended in the documentation that the duration between API calls should be no more than once every 10 minutes if fresh data is expected as the result each time. As a user will typically only download the forecast for a single location and this is stored within the database, the stored forecast will be displayed to the user and hopefully discourage the desire to repeatedly download fresh forecasts.

The service is called using the *api.openweathermap.org/data/2.5/forecast?q=Location,uk&units=metric&APPID=* URL and utilises the following API key: **b3dddf9efa1e2a08b8be9074c87df59e**.

Which Requirements Have Been Implemented

The following requirements have been successfully implemented and tested:

1. Interlinking Screens: there are a total of 5 activities – a welcome screen and a preferences screen included - all of which the user can access through buttons.
2. Storing Preferred Locations: shared preferences are used to store a maximum of 10 preferred locations for the user. These are stored using user given input in the form of selection from a spinner.
3. Downloading Data from a Web Service: by clicking a button, the API is called and returns forecasts for the current location as a JSON String.
4. Parsing XML or JSON: the JSON received from the web service download is parsed and stored in suitable objects.
5. Managing Data in an SQLite Database:
 - a. the data received from the web service download, once parsed, is stored in an SQLite database using the Room system. This includes location name, date, time, temperatures, humidity, wind speed, wind direction and rainfall amounts.
 - c. the user is also able to delete all the stored data.
6. Search:
 - a. the user is able search stored data for a specified location based on temperatures within a specific time frames.

The following requirements have been mostly implemented and completely tested:

6. Search:

- a. the user is able search stored data for a specified location based on temperatures/humidity/wind speeds either greater than or less than a given value.
 - i. This has been implemented, but the queries do not return the correct results.

Which Requirements Have Not Been Implemented

- 5. Managing Data in an SQLite Database:
 - b. the user can browse all the stored data.
 - i. This has not been implemented at all.