Requirements [WIP]

Iot sensor monitor system
Dataset: Rain in Australia

Basic Description of project //TODO

A pretend 'real-time' weather information app for Australia. Users can see "current" weather data for their approximate location in the form of map overlays and info widgets. The app also predicts whether it will rain the next day based on current weather conditions. We will train the prediction model on "past data".

USER REQUIREMENTS **(what will users be able to do with the system?)**

Users will be able to use this system to view Australian weather information. They will be able to interact with two main components: a map, and a summary.

- **Map.** The map is interactive and provides visualizations for current weather data anywhere in Australia. Users can choose from different graphic overlays on the map. They can also click anywhere on the map to see the summary for that location.
- **Summary.** The summary provides all the weather information for a specific location. It also provides a prediction for whether it will rain in that location tomorrow.

SYSTEM REQUIREMENTS

Functional Requirements **(what will the system do to support users?)**

- **Location Selection:** Users can select any location in Australia in one of two ways: (1) a search bar, (2) Clicking on a point on the map. Data is provided for the closest weather station to the selected location.
- **Date Selection:** Since we are not actually using live IoT data, the user must enter a date between 2007 and 2017. The system will pretend that the data is live, and as such the rain prediction will be based only off of data collected until the selected date. *(- is this a functional req? I'm not sure)*
- **Summary Component:** The summary section will display all available weather information for the given date and location.
- **Map Component:** The user will be able to visualize current weather data by choosing what information they want to see on an interactive map. Relevant icons will appear over each city to quickly communicate the information.
    - *For example, if the user selects the wind overlay, each city will display an arrow of a certain size to represent wind speed and direction.*
- **Rain prediction**: Given the date and location, the system will provide a prediction on whether it will rain the following day, based on data collected up to that date. This is displayed in the summary.

Non-functional Requirements **(Constraints)**

Product Reqs

- The system must be a web-application.

- The user interface must be flexible (i.e. it must be usable on different devices and window sizes).
- Performance: Search results should be returned within **X** seconds.

Organizational Reqs
- The system must conform to the process and deliverables defined in COSC 310**??**
- Development reqs: The system must be developed in Python (since support is available for that)
- The system must be completed by [date]?

External Reqs
- The system will not collect any user information.
- There will be no log-in system, as we are using publicly available data.

Miscellaneous
- Security: Admin login - *where does this go*
- The system is limited to Australian weather data from 2007-2017

Domain Requirements **(Standards)**
- The system will implement at least basic web accessibility features, as specified in WCAG2 (Web Content Accessibility Guidelines). Most relevant:
    - Pretty much all of 1. Perceivable
        - 1.1 Text alternatives
        - 1.4.4 Resize text, and other 1.4 stuff
    - 2.4 Navigable

- The application will use HTTPS (I assume?)