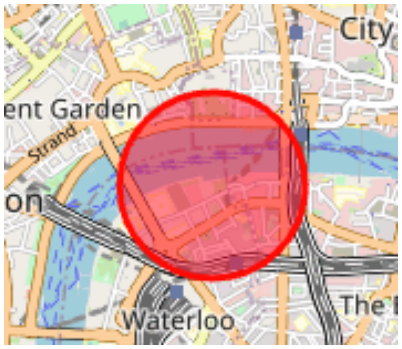
UI and Database Integration Improvement Scenario For Environmental Recorder

Group 3 - Alex Escatel, Mike Squires, Daniel Alzeidan, Hamza Ali

Scenario UI Improvement

Currently the UI is very barebones. The login screen and map screen have outdated fonts and styles. In this scenario, the UI will be modernized and the flow of the program will be improved by adding a navigation bar to the top of the page. This navigation bar will ideally include a logout/login button, search bar, and some sort of map reset button. First the navigation bar will be added and then the styling will be applied so that the functionality works before wasting time on styling.

The map also has a feature to add red circles around waypoints to show dangerous areas. For example if the CO2 levels are too high or maybe the temperature is above or below a certain point. In this scenario we will do testing with this feature to see the feasibility of it. The idea would be that a red circle would appear around the arduino's location that is reading in dangerous air quality readings.



Database Improvement & Security

Currently, the database does not store any live data and there is no functional way to store data or keep tabs on live devices. In order to move the database to be conducive to a more functional release the following actions need to be taken:

1) Maintain a list of live devices and facilitate communication between devices. This data should be able to be broadcast to the front end administrator panel in order to see the status of each device and all devices registered.
2) Schedule polling updates from node devices to receive the most current data.

With hardware and software keys being vulnerable in the public domain, scattered throughout various locations security has become an utmost concern for our product. To mitigate the ability for infiltrators and hijackers to steal devices, submit false data, completely hijack the database with a rogue api key we have proposed the following security measures:

1) Have the application make requests to the node devices on a regular basis. If the data returned is not in the format expected the device should be ignored.
2) The only device with the API key should be the host application.

Requests made to the node devices can be facilitated through a network connection to a local web server. The Arduino node devices are already equipped with a wifi module which will be connected to a secure internet connection 24/7. In this case there is no value in intercepting the sensor data since it is publicly available through our application already. With these changes hijackers will have no way to directly access host application or database.