Maame Efua Boham
Maame Yaa Osei
Molife Chaplain
Ariel Woode

Abstract/Summary

The problem the team is currently attempting to solve falls within the domain of climate control/regulation. The problem of climate control is focused on identifying the temperature, humidity, brightness and other features of a room and then adjusting those features to suit its occupants. Considering that within Ashesi, there are significantly many more students than there are lecture halls, attempting to tweak the room climate to suit occupants will most likely be futile.

Currently, in order for students to find a classroom they believe will be the right temperature and humidity for their comfort, students must manually move from class to class to select the best room. This is time consuming and tedious and often not as worth the effort. As such, students often end up in a class that does not match up to their requirements because they it was closer and less time consuming to find than their ideal room.

This also presents an issue for school authorities in the way of power consumption. Students that do not find a class with the right temperature and humidity, may choose to occupy an empty class that they can adjust to their preference. This increases the amount spent on powering these facilities.

Based on the problem domain and the challenges of current solutions as stated above, the task the team has taken on is a system that identifies the temperature and humidity elements of various rooms in the university and provides that information to students in an easy to understand graphical format in near real-time. This would allow them to make informed decisions as to what lecture hall would best suit their needs at any given point in time. The system is built using a Raspberry Pi and a Temperature and Humidity sensor.

The final solution is a user-friendly webpage that displays the temperature and humidity of a given space. It uses a MongoDB Atlas, Cloud MQTT, Node.js, Bootstrap, Charts.js and AJAX to implement this.

Introduction: Justification/Motivation

https://ieeexplore.ieee.org/abstract/document/7072989

This paper by Amoo, Guda, Sambo, and Soh discusses the importance of control over room temperature, a key measure when it comes to ensuring good working conditions, and also a health and safety issue. The design used in the paper involves the use of a microcontroller, PIC16F876A, and peripheral devices such as an LM35 temperature sensor, LCD display unit to form an all-encompassing single system. The system takes user input as to preferred temperature statistics, compares those to what is already present in the room, and uses that output to make a decision as to whether the AC or heating system should be activated.
The finished system had a result accuracy of 95% regarding the control decision.

Challenges of related work

Electromagnetic relays (EMRs) are used as the actuators in the project. However, these are quite noisy and the project could have benefited from the use of Silicon Controlled Switches as they are much quieter. In addition, the system is limited by its prototypical nature as this means that it cannot be expanded to multiple rooms. If the system was applied to a space such as Ashesi, it would not be able to effectively cater to the entirety of the campus. The system is supposed to regulate the room temperature using the AC, however, in the event of the AC failing, there are no relevant redundancies to ensure that there is no feedback system obstruction or system crash. The system takes control of the regulation once the user sets optimum temperatures and for a situation where the person is asleep in a room, massive temperature drops could lead to complications like pneumonia. On the flip side, if the system fails to work, the whole room could turn unregulated and cause situational risks like AC leakage or room overheating. The system does not have a set benchmark under or over which it cannot change the AC; this means that a malfunction in the system could cause temperature overregulation (high temperature) or under-regulation (low temperatures) both of which have an impact on the room's occupants. Given that this is an embedded system, the system will be difficult to upgrade or repair, which means in most cases the system needs a complete overhaul and if this happens data cannot be transferred from one system to another.

Methodology

Requirement Specification

Research insights:

Our research involved a focus group of 8 Ashesi students. Questions we posed sought insights on:

- **The relevant environmental factors students seek information on when working on campus.** This informed our decision to measure temperature, humidity and light intensity as a major concern that the group relayed was that they would like to know the temperature and brightness level of rooms on campus to assess whether they would like to study there or not.
- **Student preferences on how the information gathered from sensors should be visualized.**
  The students in the focus group relayed that they prefer to see current temperature and a time series chart of temperature in the rooms under consideration to be able to know the typical temperatures of rooms over given periods of time.

Functional and Non-Functional Requirements:

Functional:

- Students must be able to see layman information(in degrees celsius) on room temperature in individual rooms.
- Students must be able to access the service via the internet.
- The system must get data from the sensor package at a constant rate.
- The system must be able to provide real-time insights on sensor data.
- The sensor package must be able to send data to the system via the internet.
- The sensor package must provide accurate room-temperature and humidity average readings.
- The students should not be allowed to be in direct contact with the sensory kit.

Non-functional:

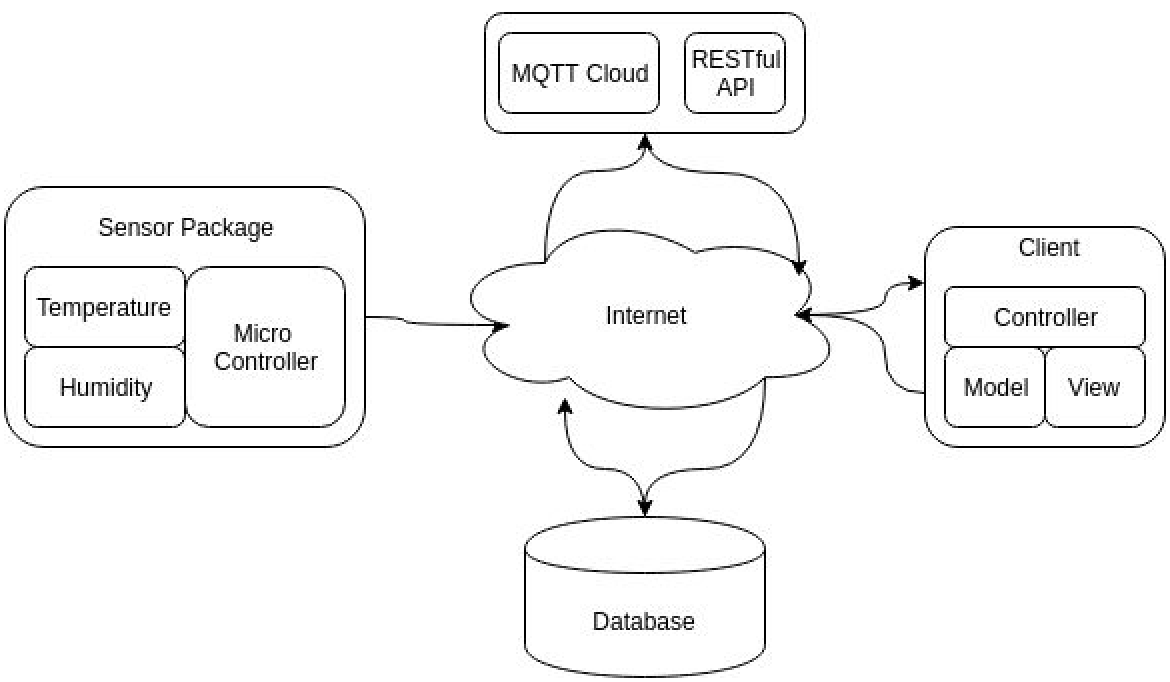
- *Usability*: because the system presents complex data, it should be easy to understand, analyze, and predict patterns from the data as provided on the interface to the user. This means that the information the user gets from the system should be properly processed, easily understandable and ready for other uses.
- *Safety*: the system should be reliably and consistently safe. Which means among other things it must not blow up in people's faces and must ensure the user is not exposed to potentially harmful data, situations and or external programs.

- *Reliability*: the system's data, being pivotal must always be up to date and available for use. This means that the system must always be online and be able to provide relatively accurate data at regular intervals

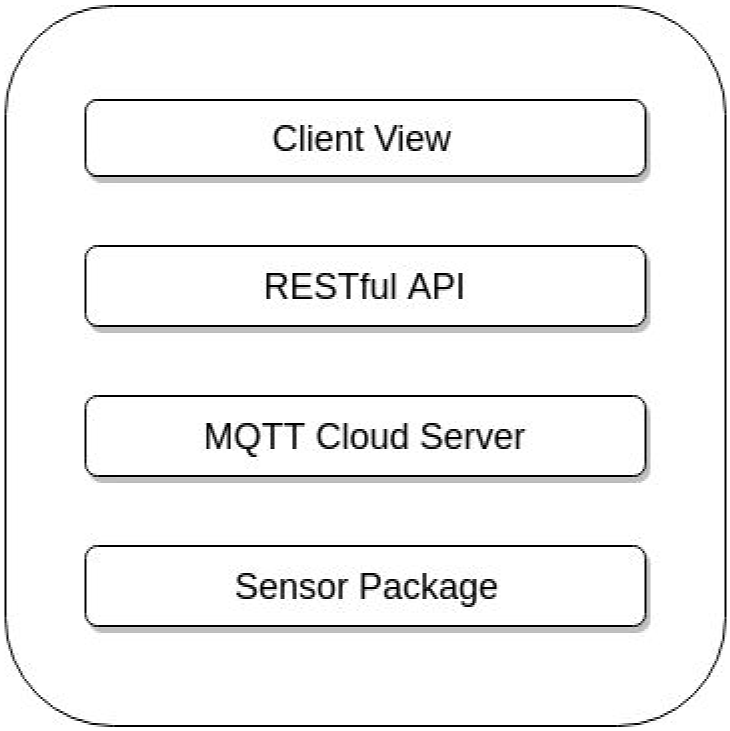
Analysis and Design

Architectural patterns

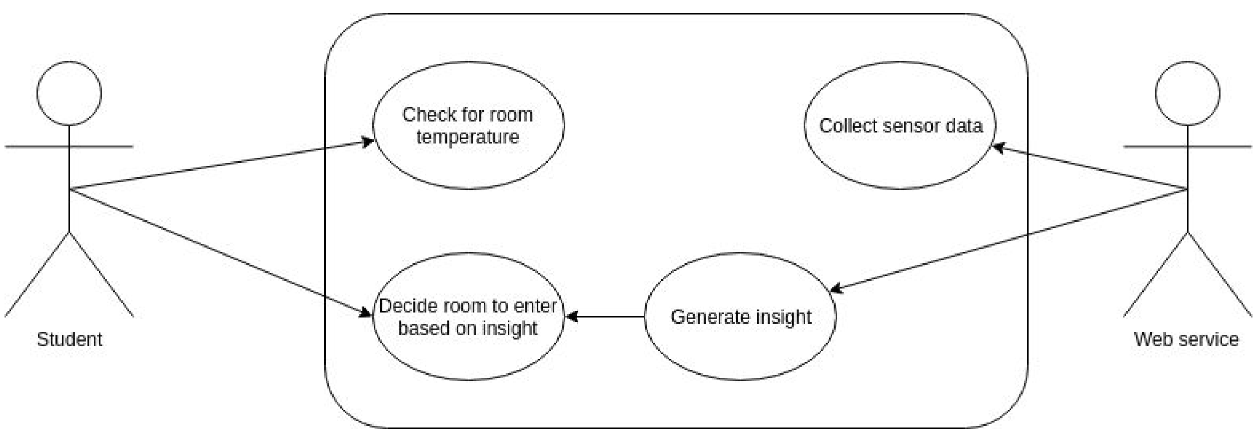
Restful Web service:

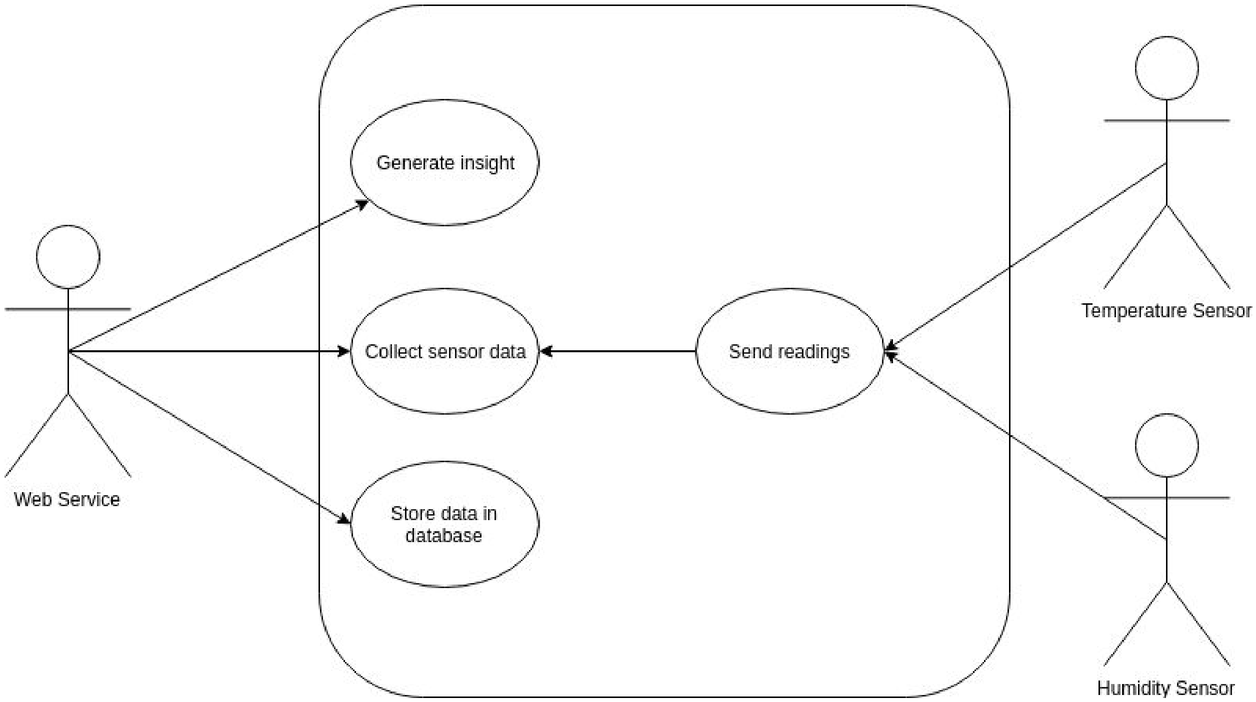


Layered View:

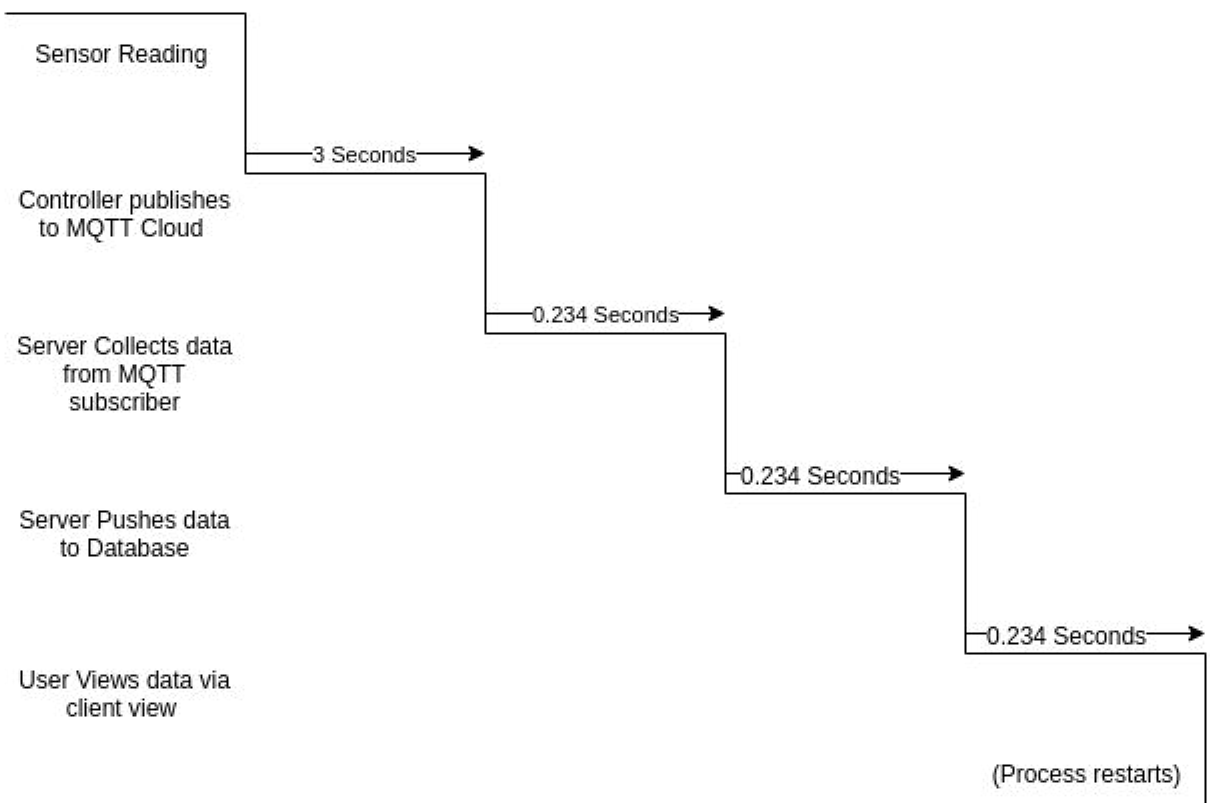


UML diagrams





Time Series Diagram



Implementation

Technologies chosen:

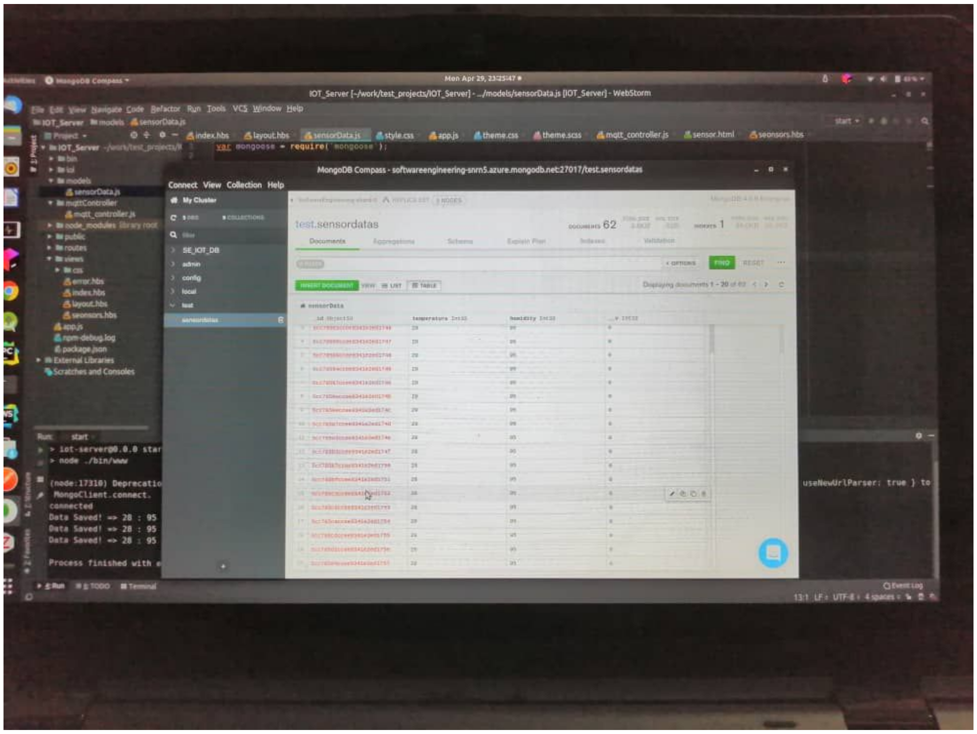
- Raspberry Pi
- DHT11 sensor
- MongoDB Atlas
- Cloud MQTT
- Node.js
- Bootstrap
- Charts.js
- AJAX

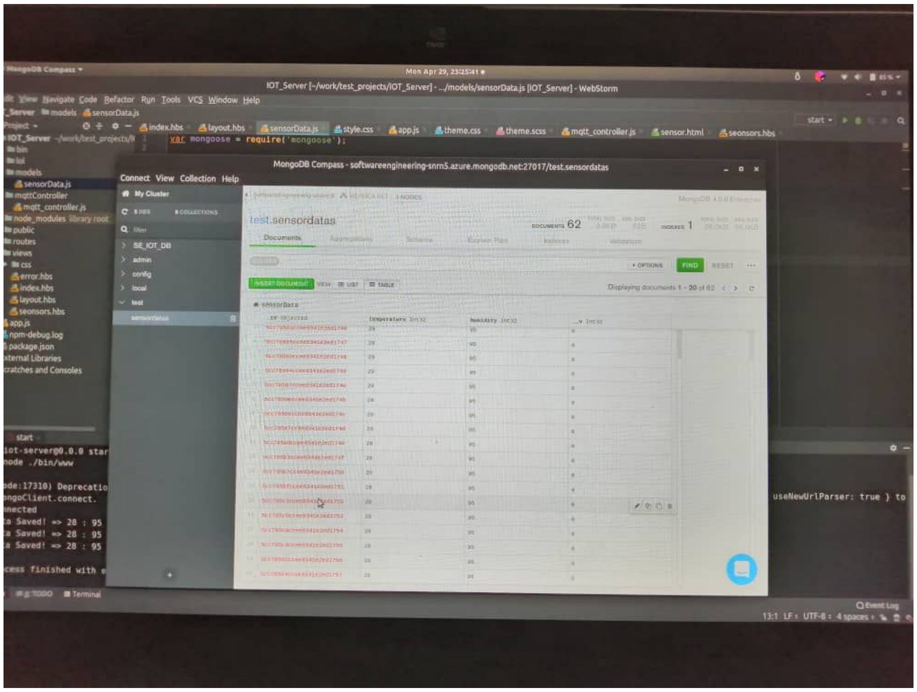
Justification of technologies

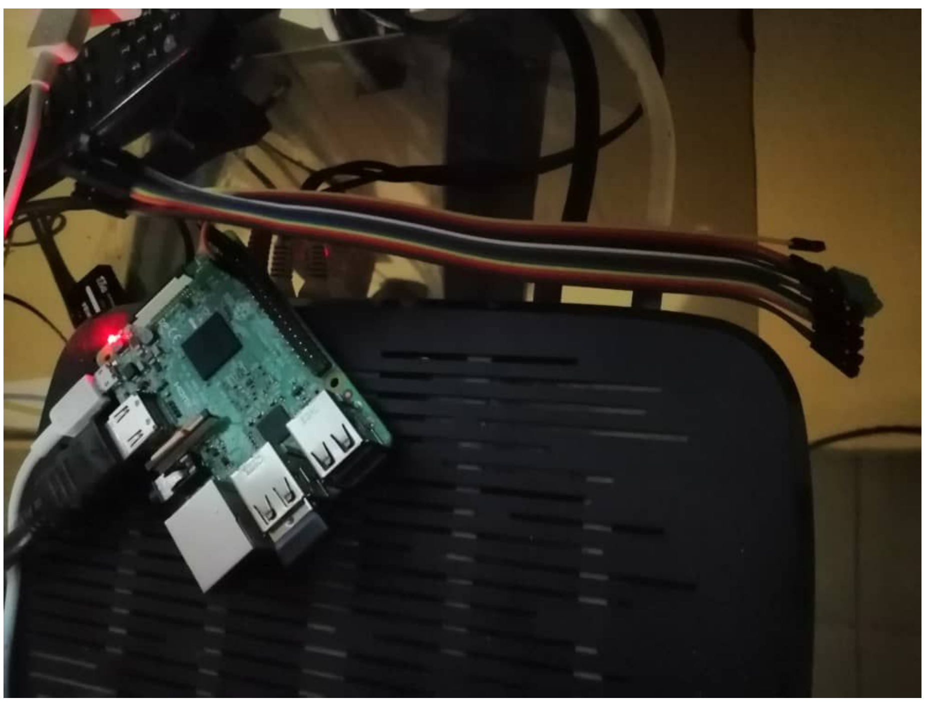
One of the biggest challenges of the Arduino system we intended on using initially is its need for a dedicated WiFi module to run its connection. As only a microcontroller, it also requires that all processes be managed off the system, thus making it more difficult to work with. This informed our decision to switch to using a Raspberry Pi instead as it comes with a dedicated ethernet port and WiFi module.

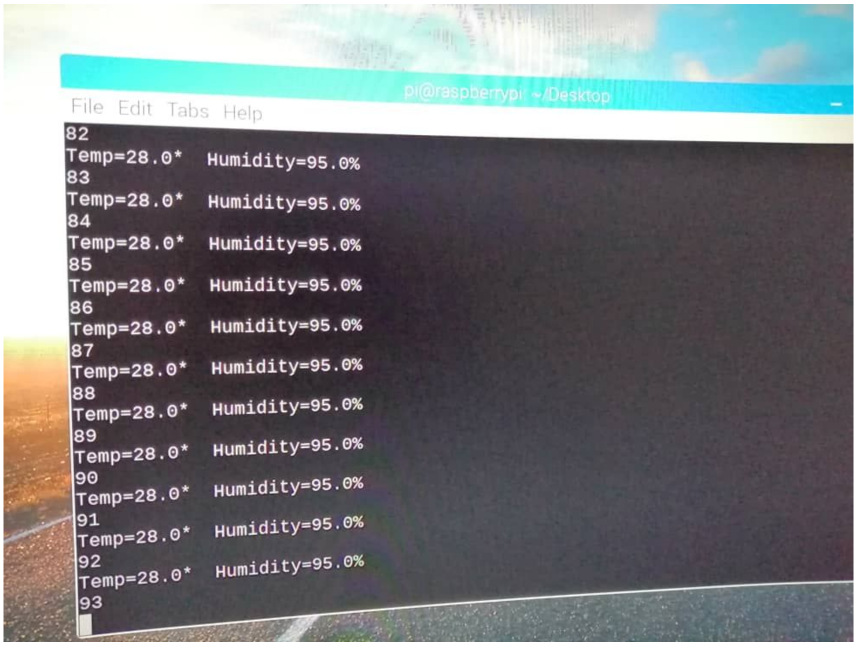
Verification and Validation (10pts)

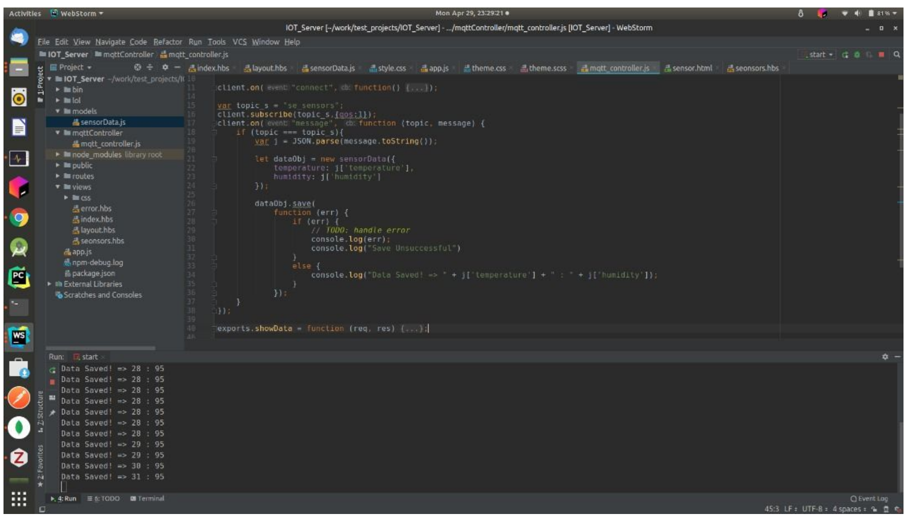
Illustration of testing process:

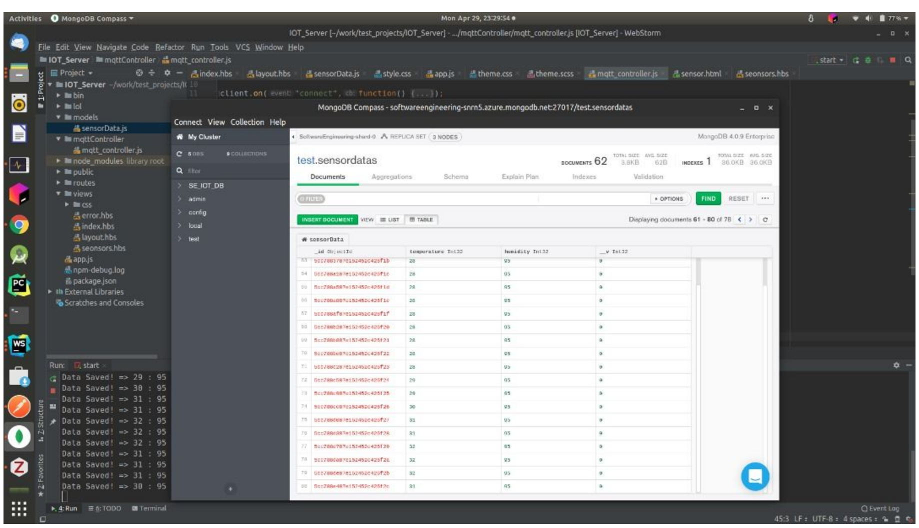



IOT_Server [~/work/test_projects/IOT_Server] - .../models/sensorData.js [IOT_Server] - WebStorm
var mongoose = require('mongoose');
MongoDB Compass - softwareengineering-snrn5.azure.mongodb.net:27017/test.sensordatas
Connect View Collection Help
test.sensordatas
Documents
FIND
RESET
INSERT DOCUMENT
sensorData
Saved! => 28 : 95
Saved! => 28 : 95
Saved! => 28 : 95
useNewUrlParser: true } to




pi@raspberrypi: ~/Desktop
File Edit Tabs Help
82
Temp=28.0* Humidity=95.0%
83
Temp=28.0* Humidity=95.0%
84
Temp=28.0* Humidity=95.0%
85
Temp=28.0* Humidity=95.0%
86
Temp=28.0* Humidity=95.0%
87
Temp=28.0* Humidity=95.0%
88
Temp=28.0* Humidity=95.0%
89
Temp=28.0* Humidity=95.0%
90
Temp=28.0* Humidity=95.0%
91
Temp=28.0* Humidity=95.0%
92
Temp=28.0* Humidity=95.0%
93


Data Saved! => 28 : 95
Data Saved! => 28 : 95
Data Saved! => 28 : 95
Data Saved! => 28 : 95
Data Saved! => 28 : 95
Data Saved! => 28 : 95
Data Saved! => 28 : 95
Data Saved! => 29 : 95
Data Saved! => 29 : 95
Data Saved! => 30 : 95
Data Saved! => 31 : 95


test.sensordatas
Data Saved! => 29 : 95
Data Saved! => 30 : 95
Data Saved! => 31 : 95
Data Saved! => 31 : 95
Data Saved! => 32 : 95
Data Saved! => 32 : 95
Data Saved! => 32 : 95
Data Saved! => 31 : 95
Data Saved! => 31 : 95
Data Saved! => 31 : 95
Data Saved! => 30 : 95

Test cases:
We tested the system (including display, interaction, connectivity) in the Charlotte common room. By holding the sensor we were able to validate that the sensor was actually getting real time temperature and humidity readings and sending them to the database as intended by the system.

Product Installation/Deployment

GitHub: https://github.com/maameyaa18/AshSense

Evolution/Future Work

The dependency of the system on power and internet. We could solve this by connecting the system to a power source that saves power for times when the electricity is off. We also realized we could make the system display more intuitive to the user needs; temperature readings, temperature ratings, and temperature suggestions. When the system server is shut down, the whole system dies, which means that we will need to ensure system redundancy and continuity even after the system server goes down. We also realized we can solve the problem of the internet by ensuring the server and the data gathering components are connected to the same network.

New functionalities that could be added include adding new sensors like pH sensors and light sensors for people with light sensitivities or for farmers who require knowledge or regulation of temperature, humidity, pH and light intensity in order to gain the best crop yield.