Summary

The system developed for this assignment is a weather application that tracks the temperature and humidly. The sensor used for this application is the DHT11 sensor as it includes both temperature and humidly. The DHT11 was used as it is low in cost and can track both temperature and humidly.

In this given context, the application is used to for weather monitoring. However, once implemented, the technology can be used for various applications that requires tracking temperature and humidly. One example of this is food storage and preservation. Where food storage and preservation are used in various contexts such as retail stores, food processing plants and refrigerated warehouses.

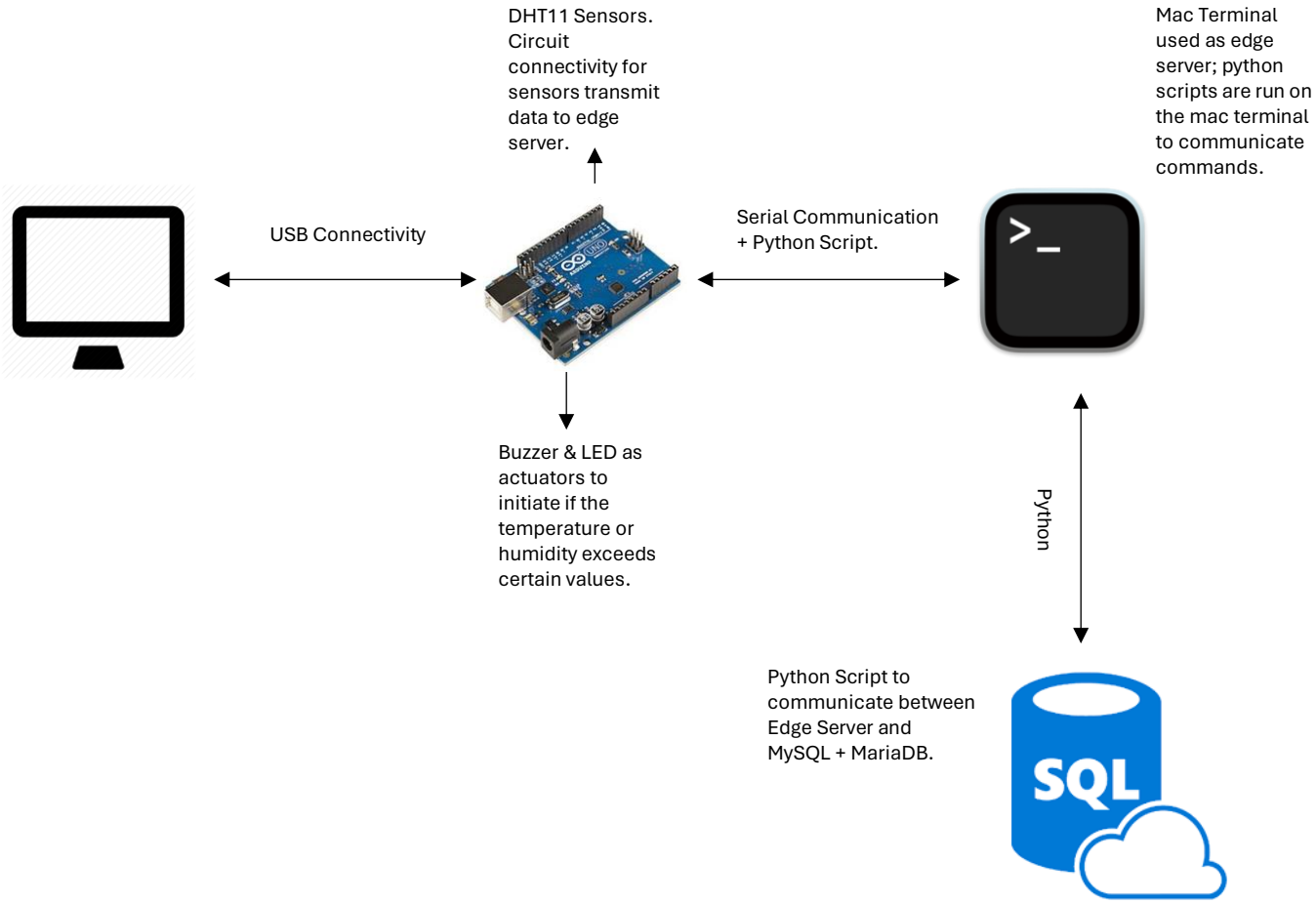
The DHT11 sensor is a versatile and cost-effective device used for measuring temperature and humidly. It communicates through only 3 pins (5v, GRND, and pin number) making it easy to integrate with Arduino. This straightforward set up makes it a simple to integrate with Arduino boards. After establishing the correct wiring connections, leveraging its functionalities becomes accessible by installing the appropriate libraries. For this project, the Adafruit DHT sensor library was selected, but a range of options exist to cater to different applications and preferences.

The system designed for this project consists of the DHT11 sensor, Arduino board, LED, and Buzzer. The DHT11 sensor is responsible for picking up the temperature and humidly and transmit the information via the serial communication to the edge server, which in this case the mac terminal is used instead of Raspbian Desktop. On the edge server, the received data is logged into MySQL + Maria database named Iot_database, with dedicated tables under Sensor_readings. While real-time tracking is the goal of the system, this project focuses on establishing the core functionality which is reading sensor data via the terminal, transmitting it to the database and employing python script for edge analytics. This script incorporates conditional statements to trigger LED and buzzer alerts when temperature and humidly exceed a pre-defined threshold.

Link to Demonstration:

https://www.dropbox.com/scl/fi/7k4w7qp6ntcb7drtzakc0/IMG_0008.mp4?rlkey=6m7q79reiflzv5gwmpfdghdg6&dl=0

Conceptual Design



Implementation

The implementation of the weather monitoring system begins with assembling the hardware components, including the DHT11 sensor, Arduino board, LED and Buzzer.

Once the hardware setup is complete, the software aspect involves installing necessary libraries, such as the Adafruit DHT sensor library, to enable communication between the DHT11 sensor and Arduino. This library allows for reading data from the sensor and transmitting it to the Arduino board via serial communication.

On the Arduino IDE, a sketch is developed to receive data from the DHT11 sensor, process it and transmit to the edger server (Mac terminal). The Arduino sketch also includes logic to control the LED and Buzzer based on the predefined threshold for temperature and humidity.

The edge server, which runs a python script, receives data from the Arduino via serial communication. The python script is responsible for parsing the incoming data and logging it into MySQL + Maria database named Iot_database and performs edge analytics. Conditional statements in the python script trigger alerts, activating the LED and Buzzer when the temperature and humidly values exceed predetermined thresholds. In this system, two python scripts are employed onto the edge server to separate functionalities which leads to better design and improve troubleshooting in the future. The first python script is purely responsible for reading the data and transmitting it to the database. The second script establishes communication between Arduino and provides conditional statements and exception handling.

Resources: List the online tutorials, guides, manuals, software that was utilised in the creation of your system.

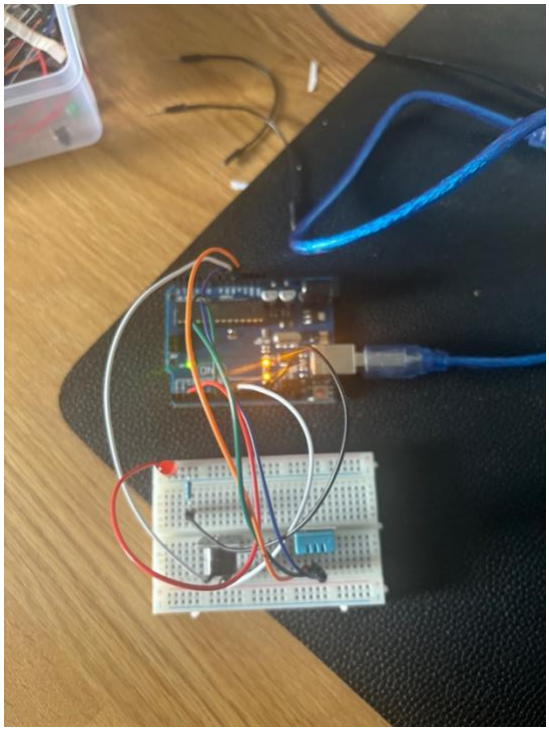
https://github.com/adafruit/DHT-sensor-library
https://www.arduino.cc/reference/en/
https://components101.com/sensors/dht11-temperature-sensor

Libraries

- **Adafruit DHT sensor library**
- **Python – Time**
- **Python – my.MySQL.connector**
- **Python - Import serial**

Appendix: All related Source Code, Sketches, Scripts and etc from the project

id	timestamp	temperature	humidity
1	2024-04-17 10:45:53	22.6	60
2	2024-04-17 10:45:55	22.6	60
3	2024-04-17 10:45:57	22.6	60
4	2024-04-17 10:45:59	22.6	60
5	2024-04-17 10:46:01	22.6	60
6	2024-04-17 10:46:03	22.6	60
7	2024-04-17 10:47:50	22.6	60
8	2024-04-17 10:47:50	22.6	60
9	2024-04-17 10:47:50	22.6	60
10	2024-04-17 10:47:50	22.6	60
11	2024-04-17 10:47:50	22.6	60
12	2024-04-17 10:47:50	22.6	60
13	2024-04-17 10:47:50	22.6	60
14	2024-04-17 10:47:50	22.6	60
15	2024-04-17 10:47:50	22.6	60
16	2024-04-17 10:47:50	22.6	60
17	2024-04-17 10:47:50	22.6	60



All scripts and Arduino sketches are uploaded in a separate file.

Andy Ibrahim - 100685973

Documentation

Establish connection to edge server via serial communication.

- Open the terminal.
- Run: **ls /dev/tty**
- Run: **/dev/cu.usbmodem1301**
- Compile and run the Arduino sketch into the board. There's no need to switch directories.
- Switch directory to your python script directory.
  cd <*python script directory*>
- Run the python3 script.py to run the python script on the terminal.
- Run the python3 scriptC.py to run the second python script on the terminal.
- Script.py should facilitate reading data and store into database.
- ScriptC should facilitate conditional statements.

Connect to Database

Database name: iot_data;
Table name: sensor_data;

Access Database in Terminal:
mysql -u root -p

- Enter password.

USE iot_data;

SELECT * FROM sensor_readings;

DELETE FROM sensor_readings; - DELETED ALL DATA FROM THE TABLE

DELETE FROM sensor_readings WHERE value = 'your_value_here';

SHOW TABLES; SHOWS TABLES IN THE DATABASE