IoT-enabled Automated Greenhouse Monitoring System

Group 7 Members:

- Hoang Truong Hai Tran - 400389466
- Juliusz Gasior- 400490100
- Leyi Jiang - 400546864
- Tianchen Zhou - 400547964
- Haytham Eladam - 400381494

Table of Content

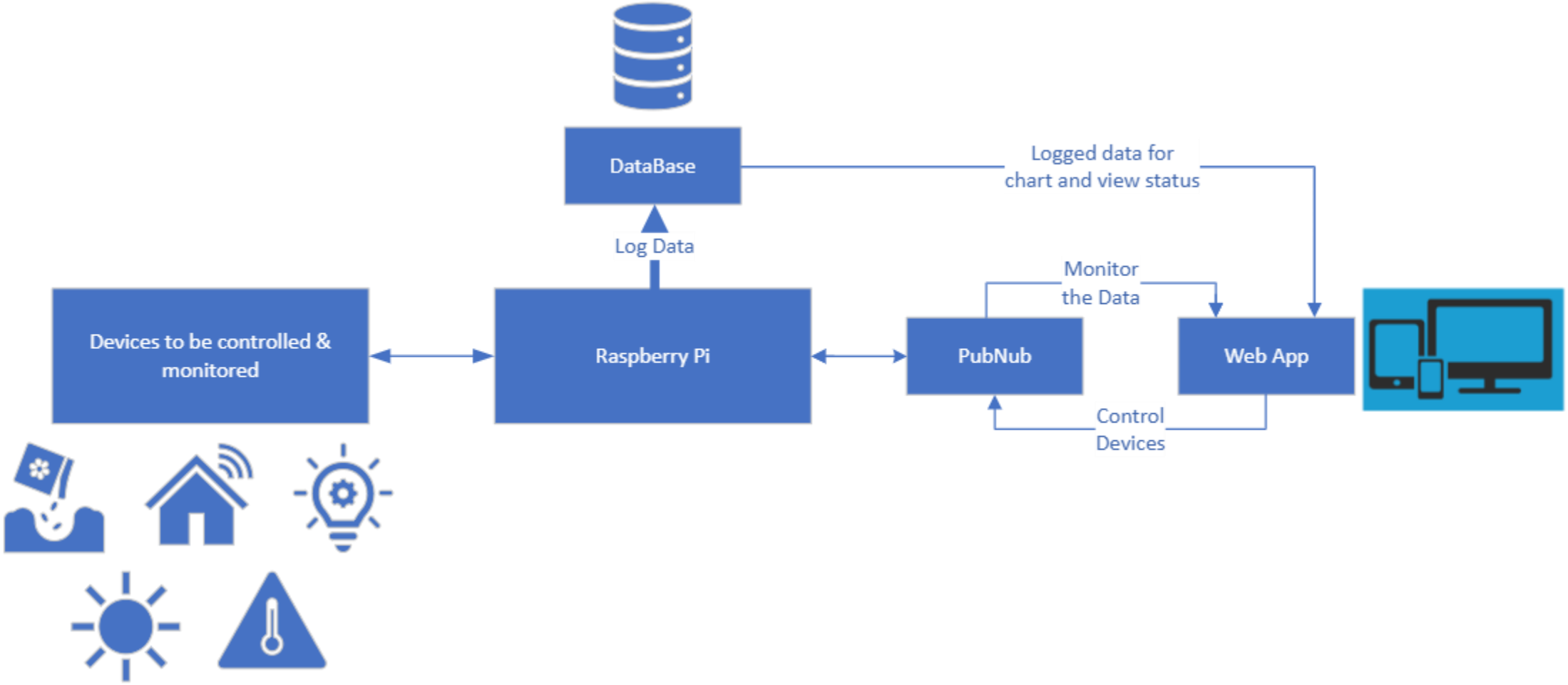
Introduction

- An IoT-enabled Automated Greenhouse Monitoring System is like a high-tech house.
- It uses a network of smart devices and sensors to keep an eye on everything we need like the right temperature, humidity, and light levels.
- Imagine a device that not only monitors the conditions within a greenhouse but also communicates in real-time (and eventually learns and adapts) to create the optimal environment for plant growth. Today, we're going to take you through the exciting features of our IoT greenhouse device, and how we've integrated PubNub and database capabilities to enhance its functionality.

Greenhouse Operation Automation

- Greenhouse environment conditions to be monitored through the web page:
  - Air temperature and humidity
  - Soil moisture
  - Light intensity
  - Rain sensing
- Greenhouse operation control:
  - Ventilator fan: Control the ambient temperature by the user decision while observing the actual temperature
  - Heater: Control the humidity by the user decision while observing the actual humidity
  - Light bulb: Control the lighting status by the user decision while observing the actual light intensity.
  - Watering pump: Control the soil water by the user decision while observing the actual soil saturation.
  - Drainage servo: Control the drainage servo pump in case there is a flooding or watering.

System Architecture



Components

- Input devices:

1. DHT11 sensor is used to measure the temperature & humidity.
2. Soil moisture sensor
3. Photo resistor sensor that is connected to A2D MCP3008 converter to measure the light intensity
4. Raindrop sensor module.

- Output devices:

1. Ventilator fan: we used DC motor with external power supply by a battery 9 VDC.
2. Heater: we used LED light for demonstration.
3. Light bulb: we used LED light for demonstration.
4. Watering pump: DC submersible water pump.
5. Drainage servo: DC Servo Motor.

- IOT Device: Raspberry Pi 4 8Gb

Communication and Database

- Pubnub is used as gateway to collect data from the raspberry pi and send it to the web page to be monitored & controlled.
- MySQL is hosted on AWS and is collecting the data directly from the raspberry pi to be logged. In addition, the historical data can be monitored through the web page trends.

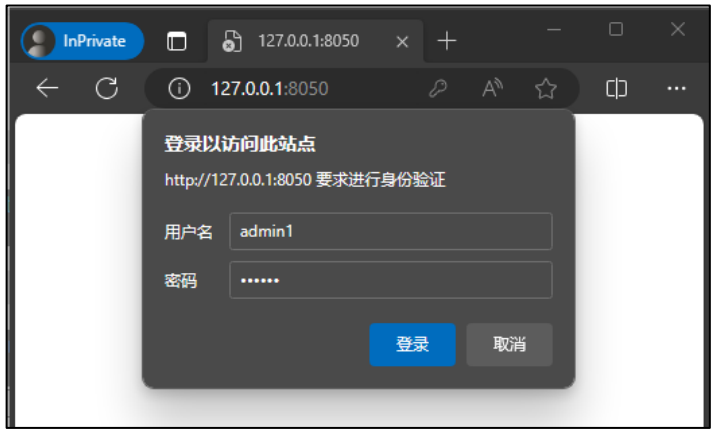
iot_data | Enter a SQL expression to filter results (use Ctrl+Space)

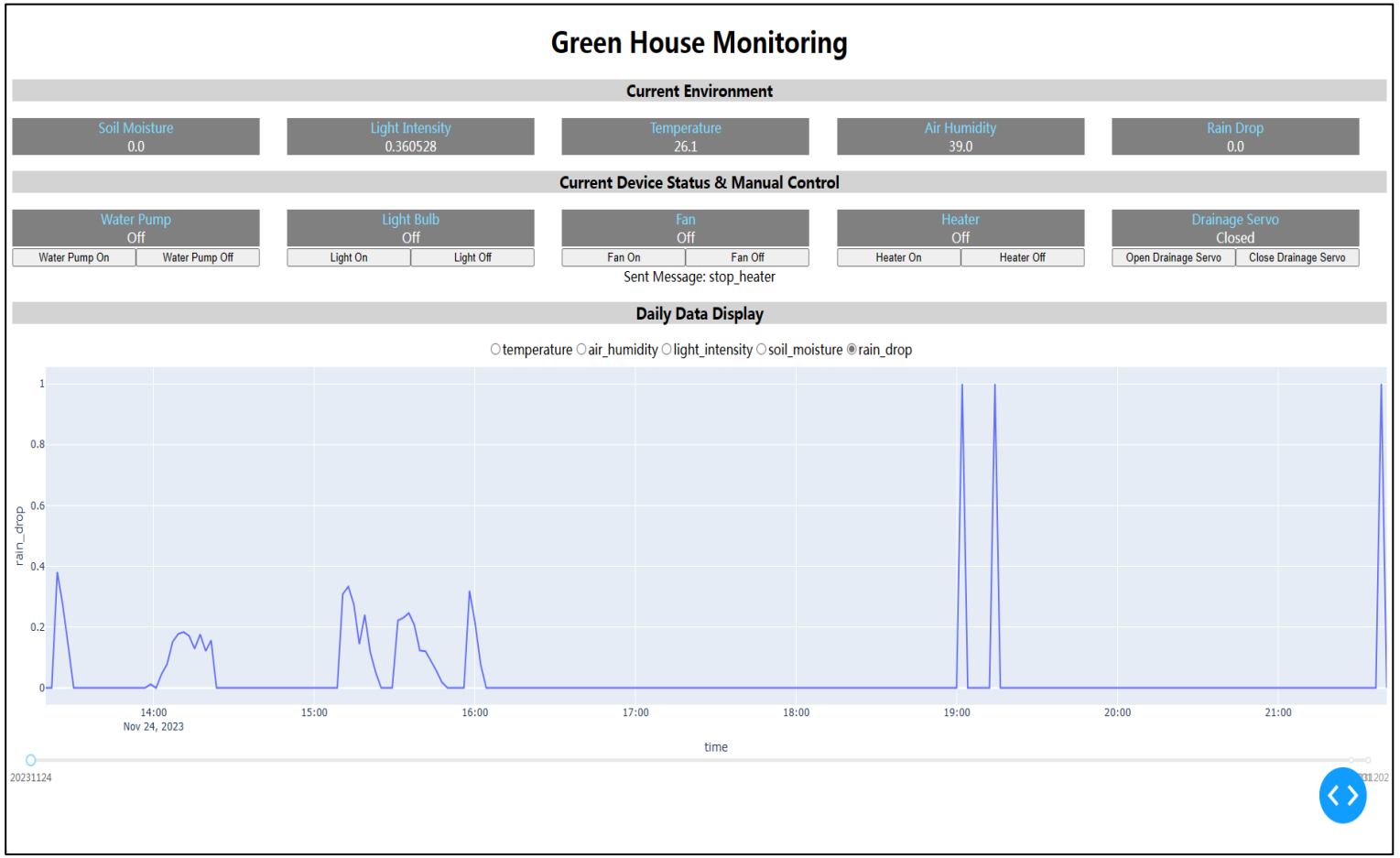
	sensor_id	sensor_type	value	time
1	temp_1	temperature	26.6	2023-11-24 13:19:43
2	humidity_1	air_humidity	18	2023-11-24 13:19:43
3	light_1	light_intensity	0.670249	2023-11-24 13:19:43
4	soil_1	soil_moisture	0	2023-11-24 13:19:43
5	raindrop_1	rain_drop	0	2023-11-24 13:19:43
6	temp_1	temperature	24	2023-11-24 13:19:58
7	humidity_1	air_humidity	17	2023-11-24 13:19:58
8	light_1	light_intensity	0.709331	2023-11-24 13:19:58
9	soil_1	soil_moisture	0	2023-11-24 13:19:58
10	raindrop_1	rain_drop	0	2023-11-24 13:19:58
11	temp_1	temperature	24.8	2023-11-24 13:22:00
12	humidity_1	air_humidity	17	2023-11-24 13:22:00
13	light_1	light_intensity	0.766976	2023-11-24 13:22:00
14	soil_1	soil_moisture	0.319492	2023-11-24 13:22:00
15	raindrop_1	rain_drop	0	2023-11-24 13:22:00
16	temp_1	temperature	24.5	2023-11-24 13:24:03
17	humidity_1	air_humidity	17	2023-11-24 13:24:03
18	light_1	light_intensity	0	2023-11-24 13:24:03
19	soil_1	soil_moisture	0.354665	2023-11-24 13:24:03
20	raindrop_1	rain_drop	0.381045	2023-11-24 13:24:03

iot_device_status | Enter a SQL expression to filter results (use Ctrl+Space)

	device_id	device_class	device_type	status	time
1	fan_1	actuator	fan	0	2023-12-02 13:10:48
2	humidity_1	sensor	humidity	0	2023-11-24 13:24:03
3	light_1	sensor	light_intensity	0	2023-11-24 13:24:03
4	light_bulb_1	actuator	light_bulb	0	2023-12-02 13:10:51
5	raindrop_1	sensor	rain_drop	0	2023-11-24 13:24:03
6	soil_1	sensor	soil_moisture	0	2023-11-24 13:24:03
7	temp_1	sensor	temperature	0	2023-11-24 13:24:03
8	heater_1	actuator	heater	0	2023-12-02 13:10:44
9	pump_1	actuator	pump	0	2023-12-02 13:10:54
10	drainage_servo_1	actuator	drainage_servo	0	2023-12-02 13:10:42

Webpage





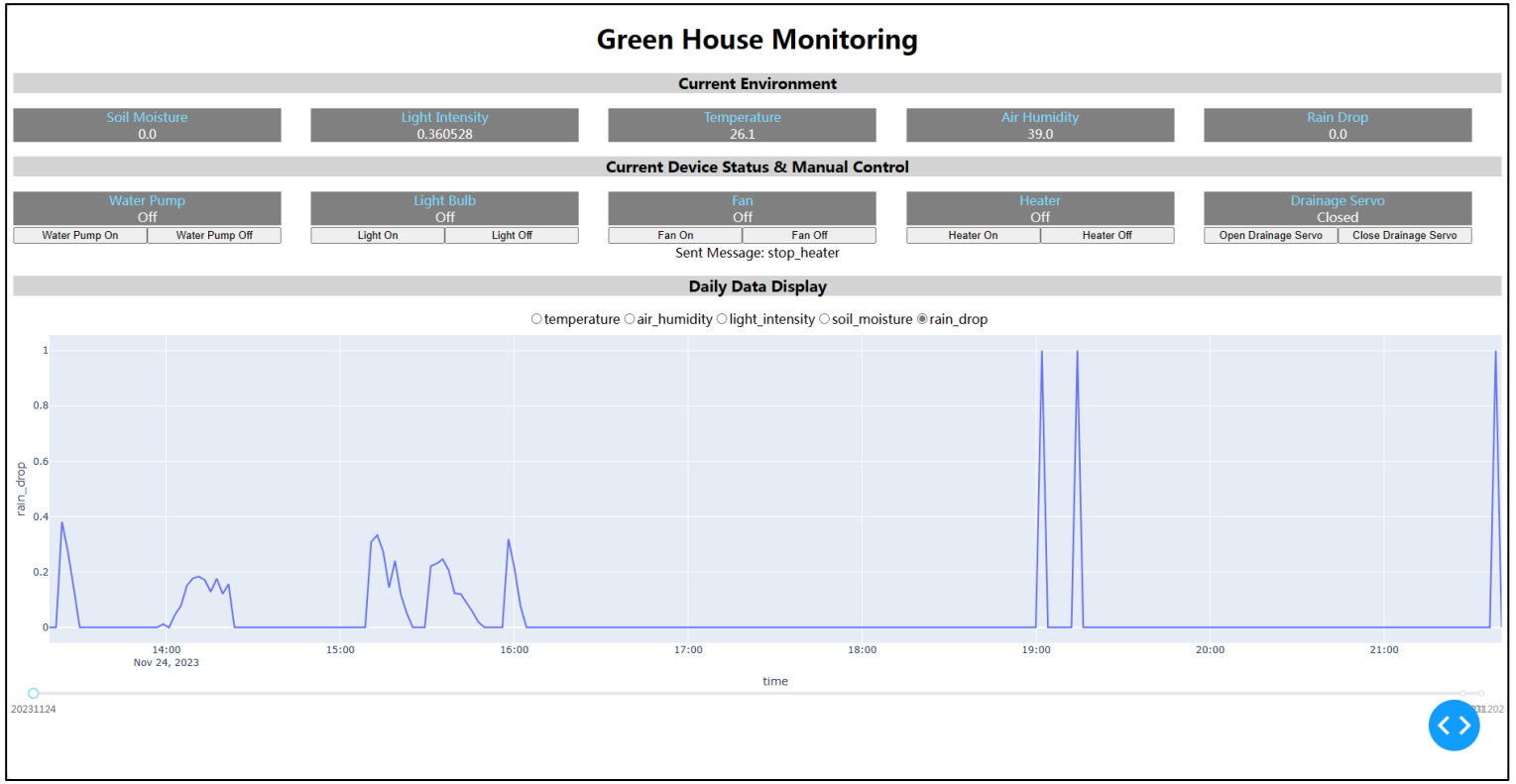
Functions

1. Authentication
2. Show the real-time environment and device status, which can change as the data change.
3. Allow the user to click buttons to publish messages on Pubnub.
4. Show the daily data figure of the environment status. It allows the user to choose the features of environment and the date.

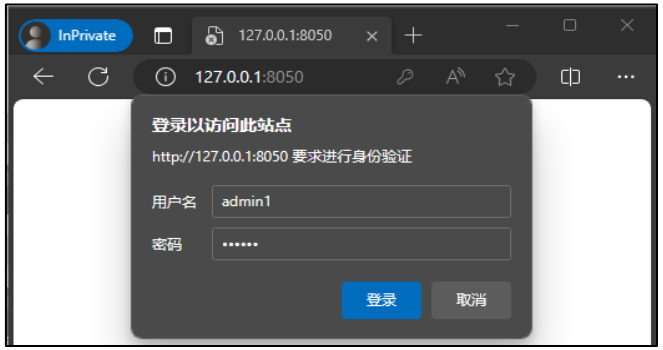
Webpage

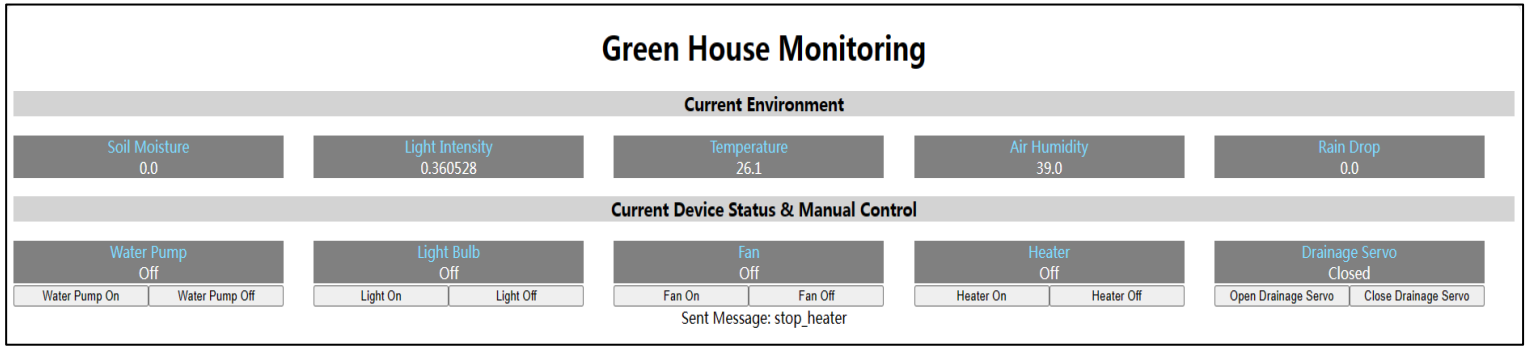
Python Library Dash

- Webpage layout: CSS style
- Real-time data display:
  - Callback function
    Input: Updating intervals
    Ouput: Environment/device status
  - Retrieve from MySQL every second
- Daily data figure:
  - Callback function
    Input: Users' clicks of features and dates, Updating intervals
    Output: Line figure
  - Retrieve from MySQL every second



Webpage





Python Library dash_auth

- Authentication Function:

  user_pass_map = {"admin1": "666666"}

  auth = dash_auth.BasicAuth(app, user_pass_map)

Python Library Dash

- Buttons
  - Callback function

    Input: Users' click

    Output: Publish message to Pubnub and remotely control the devices

Video Link

- https://drive.google.com/drive/folders/1Tp5sQKiH5E7eiwnrnTUcWSaZWBEOiCrC?usp=sharing

Future Work

- Min-Max range limits to enable auto pilot
- ML model to better control plant health
- Additional sensors to monitor:
  - Soil nutritional content (NPK)

Citation

- [1] “Dash Documentation & User Guide | Plotly,” *dash.plotly.com*. https://dash.plotly.com/

Any Questions?
Thank You!