Assignment 3 – IoT Environmental Monitoring System using MQTT

Course: CIS600 – Internet of Things

Student Name: Kosti Gowri Sai Madhurya

Date: 03/26/2025

1. Steps Used in Developing the IoT System

1. Created a ThingSpeak Channel

- o Set up three fields: Temperature (Field 1), Humidity (Field 2), and CO2 (Field 3)
- o Created a secure MQTT Device in ThingSpeak to use for authentication

2. Developed a Python-Based Virtual Environmental Station

- o Simulated sensor values for temperature, humidity, and CO2 using Python
- Used the paho-mqtt library to publish data to ThingSpeak every 15 seconds over MQTT

3. Verified Cloud Integration

- o Observed live updates on ThingSpeak charts for all three sensor fields
- Ensured data was being correctly published and stored

4. Created a Data Viewer (viewer.py)

- o Used the ThingSpeak REST API with requests to:
 - Fetch the latest reading from the station
 - Retrieve and display values recorded over the past five hours

5. Organized and Pushed Code to GitHub

- Included all project files and documentation
- Captured screenshots of terminal outputs and ThingSpeak charts
- Created a complete README.md to document setup and usage

2. Screenshots of Output

Screenshots are stored in the screenshots/ folder and include:

• ThingSpeak real-time charts

- Terminal output running station.py
- Terminal output running viewer.py
- GitHub repository overview

3. GitHub Repository URL

GitHub Repo:

https://github.com/madhurya1320/iot-mqtt-assignment

4. Reflection

This assignment provided a great opportunity to implement an end-to-end IoT solution using MQTT and a cloud backend. I gained practical experience in simulating sensor data, securely transmitting it over the MQTT protocol, and visualizing it in real-time using ThingSpeak.

Building the viewer script also helped deepen my understanding of using APIs to fetch and process data. Overall, I found this project to be valuable in reinforcing concepts like cloud integration, sensor simulation, and real-time data handling in IoT systems.

Submitted Files

- station.py
- viewer.py
- requirements.txt
- README.md
- screenshots
- This report