**MQTT System Setup Using Docker**

**Overview**

This document outlines the steps to create an MQTT system using Docker containers. The system consists of:

* A Mosquitto broker running in a Docker container.
* A Python script that acts as both a publisher and subscriber, also running in a Docker container.

**Prerequisites**

* **Docker**: Ensure Docker is installed on the machine.
* **Python**: Ensure Python is installed for local development and testing.
* **Mosquitto Client Tools**:

**Step 1: Create a Docker Network**

Create a user-defined bridge network to allow communication between Docker containers.

docker network create mqtt\_network

**Step 2: Run the Mosquitto Broker in Docker**

Start the Mosquitto broker in a Docker container on the created network.

docker run -it --network mqtt\_network -p 1883:1883 -v D:\livello\mqtt\_reader\mosquitto.conf:/mosquitto/config/mosquitto.conf --name mosquitto eclipse-mosquitto

**Step 3: Create the Python Script**

Create a Python script named mqtt\_reader.py that will act as both the publisher and subscriber.

mqtt\_reader.py

**Step 4: Create a Dockerfile for the Python Script**

Create a Dockerfile to build a Docker image for Python application.

**requirements.txt**

Create a requirements.txt file that includes the necessary libraries:

gmqtt

**Step 5: Build the Docker Image for the Python Script**

In the directory containing Dockerfile, run the following command to build the Docker image:

bash

docker build -t mqtt\_reader .

**Step 6: Run the Python Script in Docker**

Run Python script in a Docker container on the same network as the Mosquitto broker:

bash

docker run -it --network mqtt\_network mqtt\_reader

**Step 7: Verify the Setup**

1. **Check Logs**: Monitor the logs of both the Mosquitto broker and Python script to ensure that messages are being published and received correctly.
2. **Expected Output**: log messages indicating that the subscriber has received the message published by the publisher.

**Troubleshooting**

* **Connection Issues**: If the Python script cannot connect to the broker, ensure that both containers are on the same network and that the broker is running.
* **Firewall Settings**: Ensure that firewall is not blocking port 1883.

**Conclusion**

By following these steps, we have successfully set up an MQTT system using Docker, where a Python script acts as both a publisher and subscriber, connecting to a Mosquitto broker running in another Docker container. This setup allows for efficient message passing and testing of MQTT functionality. Feel free to modify any sections to better fit your specific needs or preferences! If you have any further questions or need additional assistance, let me know!