



CPE 314 Computer Networks

Project 1

Submitted by

Team Bag_kan_lang_hak

1. Pattaraphum	Chuamuangphan	63070503437
2. Phakin	Kokirdpanich	63070503438
3. Lutfee	Deemae	63070503448
4. Jidapa	Thongnirun	63070503462
5. Purachet	Phanpuk	63070503466

Submitted to

Assoc.Prof. Peerapon Siripongwutkorn, Ph.D.

King Mongkut's University of Technology Thonburi (KMUTT)

Table of contents

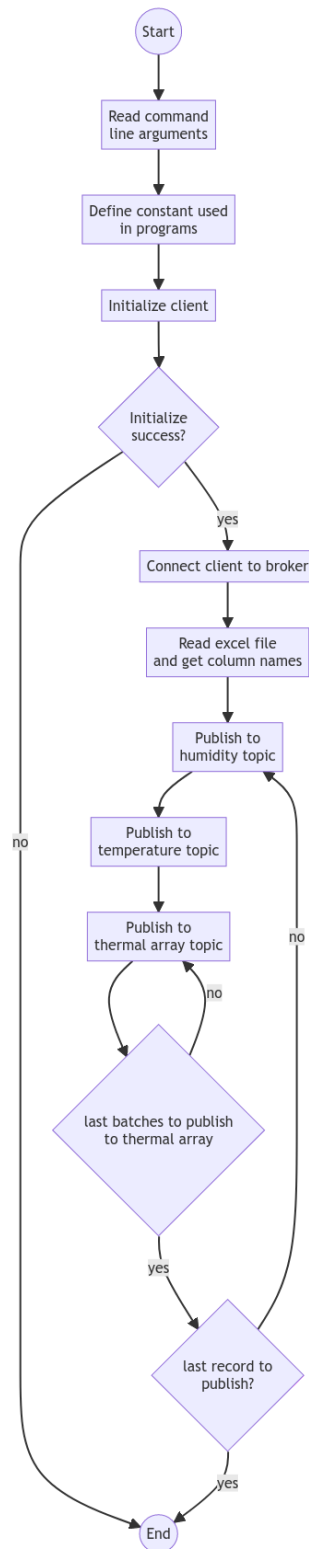
Abstract	3
Flowchart	4
Publisher	4
Subscriber	5
Operations	6
Publisher	6
Broker	6
Subscriber	6

Abstract

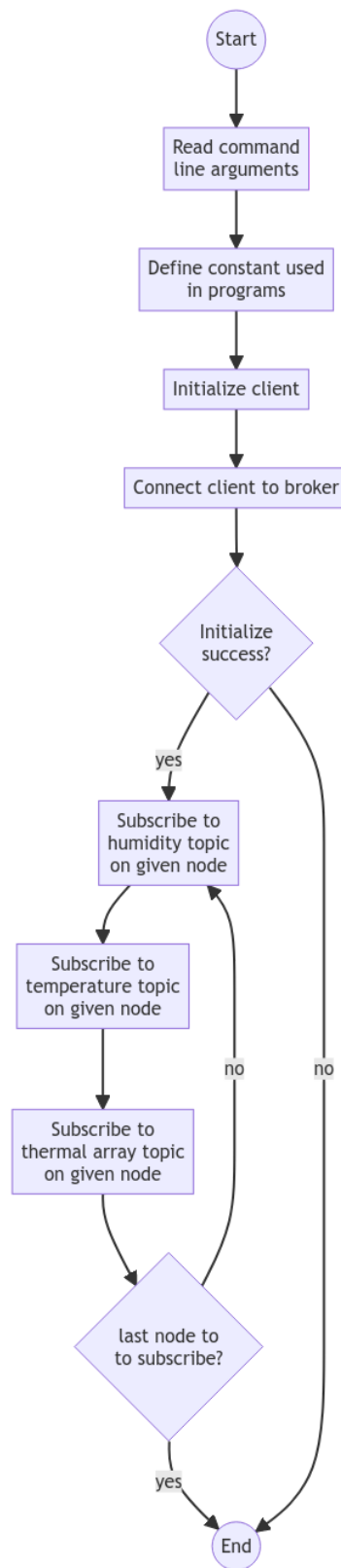
The purpose of this project is to design and implement a MQTT-based IoT application that sends data from an IoT node to a remote database. The communication model is the published-subscribed pattern, which consists of three node roles which are Publisher, Subscriber, and Broker. The tools that we use are Eclipse Mosquitto which is an open source message broker that implements MQTT protocol, Paho MQTT which is a MQTT client library that implements the MQTT protocol. This library provides the client application that can connect to the broker and also allows the client to publish the MQTT message and subscribe topic, and SurrealDB for the database.

Flowchart

Publisher



Subscriber



Operations

Publisher

The operations that are executed by the publisher including reading excel file into the program, connect to the broker by using IP address and port number. Then, for every row in the worksheet from read excel, we send humidity value through topic “nodes/client-id/sensors/humidity/time” where client-id is the client id of that node specified from command line argument, and time is the time of which the value is measured. And also for temperature value, it’s the same as humidity operations, except changing the topic from humidity to temperature. But for the thermal array, we separated the array into multiple batches, and sent it through “nodes/client-id/sensors/thermal/time/batches/batch-number” where batch-number is the number of batches in order.

Broker

The operation performed by the broker is simple, it keeps track of the connection from clients, and keeps track of which topic is subscribed by which client. And it forwards any published message received from a connected client to any client that subscribed to that specific topic.

Subscriber

The operation, in which the subscriber performs is that it connects and subscribes to humidity, temperature, and thermal sensor for array. The topic that it subscribes for humidity is "nodes/node-id/sensors/humidity/#" where node-id is the id for the node that we’re going to subscribe to, and the “#” is for wildcard to receive any other value in. And it also applies to the temperature topic, but changes from humidity to temperature. But for the thermal array, the topic is "nodes/node-id/sensors/thermal+/batches/#" where “+” is for wild card that are not the last section of topic, and in this case it’s a time, and “#” is for the batch-number from the publisher.

After receiving message from any topic, it’s written into the database immediately for that topic, but for thermal array it’s a special case where a query for existing values in the database is needed, and using the existing value combined with the new value received written it to the database.