

# Project Plan

Course: IoT HT2023

Group: 2-1

**Topic: IoT (data) protocols performance evaluation: MQTT, XMPP**

## Student names and email addresses:

Golam Sobhani Chowdhury	<a href="mailto:goch6657@student.su.se">goch6657@student.su.se</a>
Adit Ishraq	<a href="mailto:adis6811@student.su.se">adis6811@student.su.se</a>
Jaber Ahmed	<a href="mailto:jaah7248@student.su.se">jaah7248@student.su.se</a>

## 1. Project Goals:

To evaluate the performance of the data protocols - MQTT, and XMPP in different types of use cases or scenarios that may arise in IoT projects. Our evaluation will help IoT developers easily select the most performant protocol for the problem they are solving.

## 2. Project Proposal:

We will first identify different types of use cases that may lead to different architectural patterns. We will do that remaining in the scope of the design and nature of the three protocols.

Next, we would identify performance metrics relevant to each use case. Then we would design and implement a sample solution for each use case with all three protocols (in some cases it might not be possible to implement a solution for a use case by one protocol) and record the metrics.

Once the sampling is done, we would be able to summarize to have a conclusion clearly showing the reasoning behind choosing one protocol over the other in a particular situation.

**Use case 1:** IoT Communication in Low-Bandwidth Networks (e.g. sensing air quality in remote locations with limited connectivity)

**Use case 2:** Real-Time Communication (actuator responding to a motion sensor, for example turning light 'on' when a person enters the toilet and 'off' when he leaves)

**Project scope:** We would target **Phase 1** to be fully completed first, then **optionally** we would extend the scope with **Phase 2**

**Phase 1:** Use case 1 with protocols MQTT, XMPP

**Phase 2:** Use case 2 with protocols MQTT, XMPP

## 3. Project Outline/Output:

Summary suggesting the use of appropriate protocols based on measured performance metrics given a usecase type.

## 4. Design:

### 4.1 Infra setup:

One raspberry PI connected to 5 sensors, another Raspberry PI connected to 5 actuators. An MQTT server/broker, an XMPP server. Simulation script would simulate 500 sensors and 500 actuators. We may need to simulate lots of Raspberry PI programmatically.

Mosquito for MQTT

Openfire for XMPP

### 4.2 Performance metrics: (All amount as Averages)

**Use Case 1:** (Limitation = Bandwidth)

**Connection Establishment Time:** The time taken to establish a connection between the client and the server.

**Message Delivery Latency:** The average time taken for a message to be delivered from the sender to the receiver.

**Scalability:** The impact of load(hundreds to thousands) on the above two metrics.

**Use Case 2:** (Requirement = Instant Messaging)

**Connection Establishment Time:** The time taken to establish a connection between the client and the server.

**Message Delivery Latency:** The average time taken for a message to be delivered from the sender to the receiver.

**Presence indicator latency:** The average time taken for a presence indicator (ONLINE, OFFLINE, READY) to be transferred.

## 5. Development:

With low bandwidth condition, we need to simulate low bandwidth condition with traffic controlling software. Also, we need to simulate hundreds of devices sending messages to each other. We can do so by python scripts running on Raspberry PI or simulation of a Raspberry PI. For instant messaging, we need to simulate hundreds of devices connected and chatting with each other. We would set proper file based logging of

timestamps at nodes of the network (Raspberry PI, server/broker). We would create python scripts to calculate metrics from raw data in csv files.

## 6. Testing:

Once experimentation of the sampling is successful, we would run tests with proper number of parameters that would generate csv files with sufficient data to calculate the metrics.

## 7. Timing:

(iteration 1 ~ MVP, iteration 2 ~ complete implementation)

Step	Man hours (Capacity = 300)
<b>Phase 1</b>	
Design iteration 1	15
Development iteration 1	80
Testing and Fixing iteration 1	15
Adressing feedback from seminars and supervision sessions	10
Design iteration 2	10
Development iteration 2	50
Testing and Fixing iteration 2	15
Adressing feedback from seminars and supervision sessions	20
<b>Phase 2</b>	
Design, Development, Testing	50
<b>Seminar and Report Preparations</b>	20
<b>Target Total</b>	285

**8. Deliverables:** source code, project report, sampling data