

## Seminário 5 - MQTT e COAP

Engenharia de Software - 8º Período

Aluno:

Cláudio da Silva Leite

Eric Yuji Ikeda

João Vitor Cunha

Professor:

Mauricio Noris Freire

Disciplina

Arquitetura de Sistemas IoT e Cloud Computig

Data:

13/11/2024

## Introdução

A crescente adoção da Internet das Coisas (IoT) transformou a maneira como interagimos com dispositivos no cotidiano, possibilitando a automação e o controle inteligente em diversos contextos, como residências, indústrias e cidades inteligentes. No âmbito residencial, sistemas IoT permitem a integração de sensores, atuadores e centrais de controle, otimizando o consumo de recursos, melhorando o conforto e promovendo maior segurança.

Este projeto simula um sistema de automação para uma casa inteligente, utilizando sensores ambientais, uma central de controle e um termostato, integrados por meio dos protocolos MQTT e CoAP. O MQTT, conhecido por sua eficiência em comunicação leve e assíncrona, é utilizado para a troca de mensagens entre sensores e a central de controle. Já o CoAP, um protocolo projetado para dispositivos IoT com recursos limitados, permite a comunicação entre a central e o termostato de maneira eficiente.

A proposta do projeto é demonstrar como a integração de diferentes protocolos de comunicação pode criar um sistema IoT funcional e robusto, abordando desafios como o processamento de dados em tempo real e o controle automatizado de dispositivos. Este sistema serve como um exemplo prático de como tecnologias modernas podem ser aplicadas para desenvolver soluções inteligentes e sustentáveis.

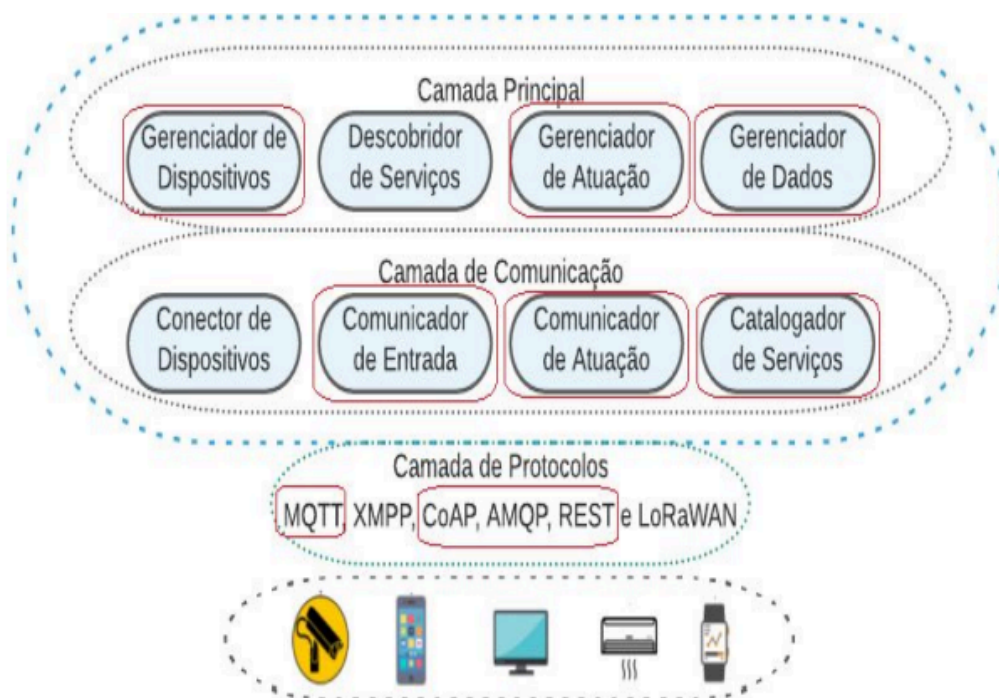


Figura 1, Esboço da Arquitetura de middleware:

<https://repositorio.ufersa.edu.br/server/api/core/bitstreams/fe130a22-c993-44af-bb7f-e0b05caa23b7/content>

## Desenvolvimento

Este projeto simula um sistema IoT (Internet das Coisas) para uma casa inteligente, onde sensores coletam dados ambientais, uma central de controle processa esses dados e um termostato ajusta a temperatura automaticamente. Ele utiliza os protocolos MQTT para comunicação entre sensores e a central, e CoAP para comunicação entre a central e o termostato. A implementação da PoC seguiu etapas bem definidas para garantir uma análise prática. Primeiramente, foi criado um modelo em Python que simulava sensores industriais, gerando dados como temperatura e vibração em tempo real. Esses dados foram transmitidos utilizando tanto o MQTT quanto o CoAP, permitindo uma comparação direta entre os protocolos.

No caso do MQTT, foi utilizado um broker para gerenciar a comunicação entre os sensores e os clientes inscritos nos tópicos. Essa abordagem demonstrou grande eficiência em cenários com alta densidade de dispositivos conectados. Por outro lado, o CoAP permitiu uma comunicação direta com o servidor, utilizando mensagens simples enviadas via UDP, ideal para dispositivos com restrições de recursos. Para visualizar os resultados, foi desenvolvido um dashboard com o framework Flask, que apresentava os dados recebidos em tempo real. A análise considerou métricas como velocidade de transmissão, consumo energético e impacto em redes instáveis, fornecendo uma visão abrangente do desempenho de cada protocolo em diferentes condições.

### Visão Geral

Os testes mostraram que o MQTT é uma escolha sólida para sistemas que exigem alta confiabilidade e distribuição eficiente de dados. Ele se destacou pela capacidade de garantir a entrega das mensagens, mesmo em condições adversas, graças ao suporte nativo à qualidade de serviço (QoS). No entanto, sua dependência de um broker central pode representar um ponto de falha em sistemas mais complexos. O CoAP, por sua vez, demonstrou ser altamente eficiente em dispositivos com poucos recursos, devido à sua leveza e simplicidade. Sua comunicação direta elimina a necessidade de um intermediário, reduzindo a latência e o consumo de recursos. Contudo, a ausência de garantias nativas de entrega pode ser um desafio em redes menos confiáveis.

Ambos os protocolos apresentaram vantagens significativas em seus respectivos contextos. A escolha entre eles deve considerar as necessidades específicas da aplicação, como escalabilidade, criticidade dos dados e limitações dos dispositivos utilizados.

## Componentes do Sistema

### 1. Sensor Ambiental (sensor.py)

Função: Simula a coleta de dados de temperatura, umidade e luz, publicando essas informações via MQTT.

Detalhes Técnicos:

Protocolo: MQTT (Publicador).

Geração de Dados: Os valores são gerados aleatoriamente:

Temperatura: 20.0°C a 30.0°C.

Umidade: 30% a 70%.

Luz: "claro" ou "escuro".

Publicação: Os dados são enviados para o tópico casa/sensor a cada segundo.

### 2. Central de Controle (central.py)

Função: Atua como um intermediário entre os sensores e o termostato, processando os dados recebidos e acionando o termostato, se necessário.

Detalhes Técnicos:

Protocolo MQTT:

Inscreve-se no tópico casa/sensor para receber os dados do sensor.

Processa a mensagem e verifica se a temperatura excede 25°C.

Protocolo CoAP:

Envia um comando POST para o termostato com o payload ON caso a temperatura esteja alta.

Funções principais:

on\_message: Recebe mensagens MQTT e chama process\_sensor\_data.

send\_coap\_command: Envia o comando CoAP para o termostato.

process\_sensor\_data: Verifica se a temperatura exige ativação do termostato.

### 3. Termostato (termostato.py)

Função: Controla o estado do termostato (ON/OFF) com base nos comandos recebidos via CoAP.

Detalhes Técnicos:

Protocolo: CoAP (Servidor).

Timeout: O termostato desliga automaticamente após 10 segundos de inatividade.

Funções principais:

render\_post: Recebe comandos para alterar o estado do termostato.

check\_state: Monitora periodicamente o estado para realizar desligamento automático.

Estados:

ON: Ativado.

OFF: Desativado.

### Fluxo de Dados

Sensor:

Gera dados de temperatura, umidade e luz.

---

Publica os dados no tópico MQTT casa/sensor.

### **Central de Controle:**

Recebe os dados publicados pelo sensor.

Analisa a temperatura e, se necessário, envia um comando CoAP para ativar o termostato.

### **Termostato:**

Recebe o comando CoAP e ajusta seu estado para ON.

Desliga automaticamente após 10 segundos, se não receber novos comandos.

### **Conclusão**

A pesquisa revelou que o MQTT e o CoAP atendem a diferentes demandas em sistemas IoT industriais. O MQTT é indicado para cenários que exigem alta confiabilidade e comunicação distribuída, enquanto o CoAP se mostra ideal para aplicações mais simples, com dispositivos que possuem recursos limitados. Uma abordagem combinada pode ser a solução ideal em sistemas complexos, aproveitando o melhor de cada protocolo. O CoAP pode ser usado para comunicação direta com dispositivos simples, enquanto o MQTT gerencia a distribuição de dados e o monitoramento em tempo real.

A principal lição deste estudo é que não existe uma solução única para todas as aplicações de IoT. A escolha do protocolo deve sempre levar em conta as características do sistema e as exigências do ambiente em que ele será implementado. Propostas flexíveis e adaptáveis são a chave para atender às demandas de um mundo cada vez mais conectado.

## Referências

1. **Constrained Application Protocol**. Vantagens e Limitações. Disponível em:  
<https://www.gta.ufrj.br/ensino/eel878/redes1-2019-1/vf/coap/conclusao.html> Acesso em 28 out.2024 às 19:10 h.
2. MQTT vs CoAP: Comparing Protocols for IoT Connectivity. Disponível em:  
<https://www.emqx.com/en/blog/mqtt-vs-coap> Acesso em: 28 out. 2024, às 19:10 h.
3. BORDIGNON, Lucas Muller. **Análise de protocolos de comunicação para a eficiência energética em sistemas IoT**. 2023. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) — Universidade Federal de Santa Maria, Santa Maria, 2023. Disponível em:  
[https://repositorio.ufsm.br/bitstream/handle/1/31391/Bordignon\\_Lucas\\_Muller\\_2023\\_TCC.pdf?sequence=1&isAllowed=y](https://repositorio.ufsm.br/bitstream/handle/1/31391/Bordignon_Lucas_Muller_2023_TCC.pdf?sequence=1&isAllowed=y). Acesso em: 02 nov. 2024, às 07:00 h.
4. **INTELLIGENT Assistant - Desenvolvimento de Algoritmos para Monitorização dos Consumos de Energia no Chão de Fábrica, Bosch**. 2024. ProQuest. Disponível em:  
<https://www.proquest.com/openview/a867c1127a2d91ff7d8e67a1b930310a/1?pq-origsite=scholar&cbl=2026366&diss=y>. Acesso em: 03 nov. 2024, às 07:30 h.
5. Análise de Desempenho dos Protocolos Coap, Mqtt-SN e Http em Smart Holmes. Disponível em: <http://repositorio.ufc.br/handle/riufc/68365>. Acesso em: 04 nov 2024, às 19:00 h.
6. Medições E Avaliações Comparativas De Desempenho E Energia De Algoritmos De Machine Learning Para Mitigar Ameaças De Disponibilidade Em Ambiente Iot. Disponível em: <https://repositorio.ifpb.edu.br/handle/177683/1736> Acesso em: 11 nov 2024, às 19:00 h.