

Projeto de IoT com Cloud na Disciplina de APLIC. DE CLOUD, IOT E INDÚSTRIA 4.0 EM PYTHON

Projeto de cafeteria IoT, conectado a dispositivos e sensores à nuvem, com o telefone e monitoramento de parâmetros ambientais. Uma colaboração de 4 membros, utilizando tecnologias como Arduino e um broker MQTT na cloud.

[Aplic de cloud iot industria 4.0 python \(github.com\)](https://github.com)



3 Contributors



Equipe



João Daniel

Responsável pelo relatório, auxiliou na codificação para integração com a Alexa, organizou os diretórios e arquivos, e ficou encarregado da documentação do projeto ([Home · Aplic-de-cloud-iot-industria-4-0-python/projeto-final-iot-cloud Wiki \(github.com\)](#)).



Estevam Souza

Entusiasta em IoT e financiador do projeto, responsável pela implementação da parte física, do broker MQTT e da integração com a Alexa, bem como pela lógica de controle dos dispositivos e pela organização das tarefas no GitHub.



Leonardo Daniel

Responsável pela coleta e o controle preciso dos dados da cafeteria, auxiliou também na implementação da parte física dos componentes elétricos e arduino.



Gabriel Fucci

Responsável pelo desenvolvimento do código embarcado no arduino para a integração com a nuvem.

Preparação do Café e Apresentação do Projeto

Enquanto preparamos o café para os participantes, aproveitamos para iniciar a apresentação do nosso projeto de IoT para a disciplina de Aplicações de Cloud, IoT e Indústria 4.0 em Python.

O projeto é uma cafeteria IoT que conecta diversos dispositivos e sensores à nuvem, permitindo o monitoramento remoto de parâmetros ambientais e o controle dos processos de preparo do café.



O que Significa MQTT?

MQTT (Message Queuing Telemetry Transport) é um protocolo de comunicação leve e eficiente, amplamente utilizado em aplicações IoT (Internet das Coisas). Ele permite a troca de mensagens entre dispositivos e serviços na nuvem de forma simples e escalável.

O MQTT se baseia em um modelo de publicação/assinatura, onde os clientes (dispositivos IoT) podem publicar dados em tópicos específicos e outros clientes podem se inscrever nesses tópicos para receber as informações.



Cafeteria IoT com Controle de Alexa e Broker MQTT na Cloud (Cloud MQTT ONE)

A Cafeteria IoT desenvolvida no projeto utiliza um Broker MQTT na nuvem (Cloud MQTT ONE) para conectar e controlar diversos dispositivos IoT na cafeteria. Com esse sistema, é possível ligar e desligar equipamentos, medir a temperatura e umidade do ambiente, além de controlar o nível de água de forma remota e automatizada.

[MQTT One | Home](#)

Além disso, o sistema está integrado com a Alexa, permitindo que os usuários controlem a cafeteria por meio de comandos de voz. Essa integração foi feita utilizando a plataforma Sinric Pro e uma placa esp8266, que é responsável pela conexão com a Alexa.

[Sinric Pro - Connect Amazon Alexa, Google Home with esp8266, esp32, raspberry pi, RP2040](#)



Utilização de Broker MQTT na Cloud



Arquitetura MQTT

O broker MQTT na nuvem atua como um intermediário central, conectando os diversos dispositivos IoT (clientes MQTT) e permitindo a troca de dados de forma rápida e eficiente.



Modelo Pub/Sub

O broker MQTT utiliza o modelo de publicação e assinatura (Publish/Subscribe), onde os dispositivos publicam dados em tópicos específicos e outros assinam esses tópicos para receber as informações.



Vantagens da Nuvem

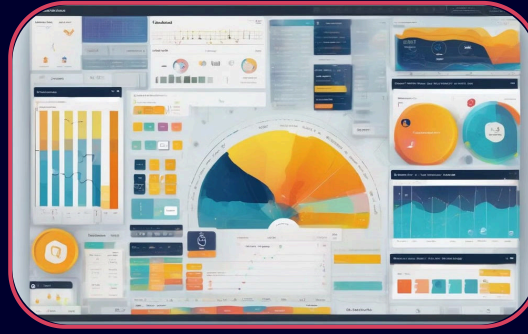
Ao utilizar um broker MQTT na nuvem, os desenvolvedores podem aproveitar a escalabilidade, confiabilidade e recursos avançados oferecidos pela infraestrutura de computação em nuvem.

Porque não utilizamos o Arduino Cloud?



Escolha do Cloud MQTT ONE

Optamos por utilizar o Cloud MQTT ONE, uma plataforma cloud com um broker MQTT, pois atende melhor às necessidades do nosso projeto de IoT na cafeteria.



Limitações do Arduino Cloud

Diferente do Arduino Cloud, o Cloud MQTT ONE nos permite maior flexibilidade na integração de diversos tipos de dispositivos IoT, além de ser uma opção gratuita e escalável.



Dificuldade com outras plataformas

A dificuldade de implementação com outras plataformas pela limitação de recursos influenciou muito em nossa decisão.



Componentes

Arduino Giga R1 Wifi

O Arduino Giga R1 Wifi é o controlador principal do nosso projeto de Cafeteria IoT. Ele é responsável pela comunicação MQTT com o broker na nuvem e pelo gerenciamento de todos os sensores e atuadores do sistema.

Sensor de Temperatura e Umidade

Esse sensor fornece dados em tempo real sobre as condições ambientais da cafeteria, permitindo que o sistema monitore e ajuste as configurações conforme necessário.

Atuadores de Energia

Os atuadores controlam a energia elétrica, permitindo que o sistema ligue, desligue os recursos da cafeteria de forma automatizada.

Funcionalidades da Cafeteria IoT

Liga/Desliga

O sistema permite ligar e desligar a cafeteira remotamente, com o toque de um botão no aplicativo ou comando de voz.

Monitoramento Ambiental

Sensores medem a temperatura e umidade do ambiente, garantindo as condições ideais para a preparação do café.

Controle de Água

O sistema monitora e controla automaticamente o nível de água, mantendo a cafeteira sempre abastecida e pronta para uso.

Código do Projeto IoT em C

O código do nosso projeto de Cafeteria IoT, desde a comunicação com o broker na nuvem até as integrações com os sensores e atuadores foi desenvolvido em C, uma linguagem de programação amplamente utilizada em aplicações embarcadas e de alto desempenho. Essa escolha nos permitiu criar um sistema robusto e eficiente, aproveitando ao máximo os recursos do nosso Arduino Giga R1 Wifi.

[cafeiteira-mqtt/src/main/main.ino at main ·](#)

[Aplic-de-cloud-iot-industria-4-0-python/cafeiteira-mqtt \(github.com\)](#)

[Sinric Pro \(github.com\)](#)



[cafeiteira-mqtt/src/alexa/alexa.ino at main ·](#)

[Aplic-de-cloud-iot-industria-4-0-python/cafeiteira-mqtt \(github.com\)](#)

Exemplos de códigos utilizando esp8266

[esp8266-esp32-sdk/examples at master ·](#)
[sinricpro/esp8266-esp32-sdk \(github.com\)](#)

Dificuldades Enfrentadas

As principais dificuldades enfrentadas durante o desenvolvimento do projeto de Cafeteria IoT foram encontrar um protocolo de comunicação IoT que permitisse a integração perfeita entre o projeto físico pré-pronto e a programação desenvolvida, bem como sua conexão com a plataforma de nuvem.

Isso exigiu um esforço redobrado da equipe para pesquisar, testar e implementar a solução mais adequada, garantindo o funcionamento completo e integrado da Cafeteria IoT.



Conclusão e próximos passos

O projeto da cafeteira IoT integrada a um broker MQTT na nuvem demonstrou seu potencial em trazer maior eficiência, conveniência e insights valiosos para os usuários. Próximos passos incluem a expansão das funcionalidades, a integração com assistentes virtuais e a análise preditiva para uma experiência ainda mais personalizada.



Coleta de Dados

Sensores monitoram temperatura, umidade e pressão durante o preparo do café, enviando esses dados em tempo real para a nuvem.



Ajuste Automático

O sistema ajusta automaticamente os parâmetros da máquina de café com base nas informações coletadas, garantindo consistência e qualidade.



Notificação para o Cliente

Quando a bebida está pronta, uma notificação é enviada diretamente para o aplicativo do cliente, informando que o pedido foi servido.