

Monitoramento de Dados via ESP e Visualização com Grafana

Cesar School

Curso: Ciência da computação

Disciplina: Conectividade de redes

Autores: Gabriel Reis, Marília Santos, Lucas Fernandes

Professor(a): Jymmy Barreto

Data: 04/06/2025

1. Introdução

Este projeto visa demonstrar a aplicação prática da Internet das Coisas (IoT) por meio do monitoramento de dados utilizando um microcontrolador ESP e a visualização desses dados em tempo real através do Grafana. A comunicação entre os dispositivos é feita utilizando o protocolo MQTT, com o broker Aedes implementado no Node-RED. A motivação surge da necessidade de soluções acessíveis e eficientes para coleta e análise de dados. O principal objetivo é criar um sistema funcional e de baixo custo que possibilite a coleta, transmissão e exibição de dados em tempo real, nesse caso para o monitoramento de CO2 em locais fechados que tenham riscos, colocando a vida em risco.

2. Metodologia

2.1 Diagrama do Sistema

Inserir diagrama: ESP → Wi-Fi → Node-RED (Aedes) → Grafana

2.2 Lista de Hardware e Software

Hardware:

- ESP32
- Computador(MV fazendo a parte da raspbarry)
- Usando o botão para simular a troca de nível do CO2
- Leds para fazer o registro do nível do CO2(Verde(Normal), Amarelo(Cuidado) e Vermelho(Perigo)).

Software:

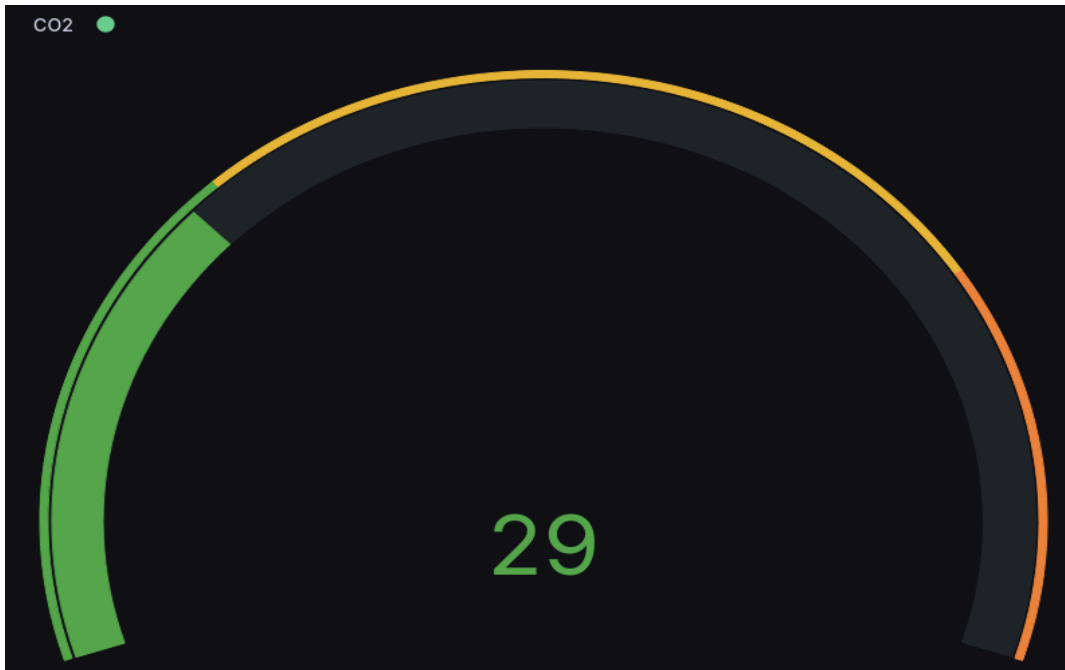
- Node-RED
- Broker MQTT Aedes
- Grafana
- PlatformIO

2.3 Fluxo de Comunicação

A ESP conecta-se ao Wi-Fi e envia dados via protocolo MQTT para o broker Aedes no Node-RED. O Node-RED processa os dados e os disponibiliza para visualização no Grafana. O fluxo de dados ocorre continuamente e pode ser monitorado em tempo real.

3. Resultados

3.1 Prints do Dashboard



3.2 Dados Coletados

●

⌚ Time	📊 Value
2025-06-04 01:15:36	28
2025-06-04 01:15:42	25
2025-06-04 01:15:42	25
2025-06-04 01:15:56	30
2025-06-04 01:15:57	28
2025-06-04 01:15:57	29
2025-06-04 01:15:57	30
2025-06-04 01:15:57	29
2025-06-04 01:15:58	30
2025-06-04 01:15:58	21
2025-06-04 01:15:58	20
2025-06-04 01:15:59	26
2025-06-04 01:15:59	28
2025-06-04 01:15:59	22

4. Conclusão

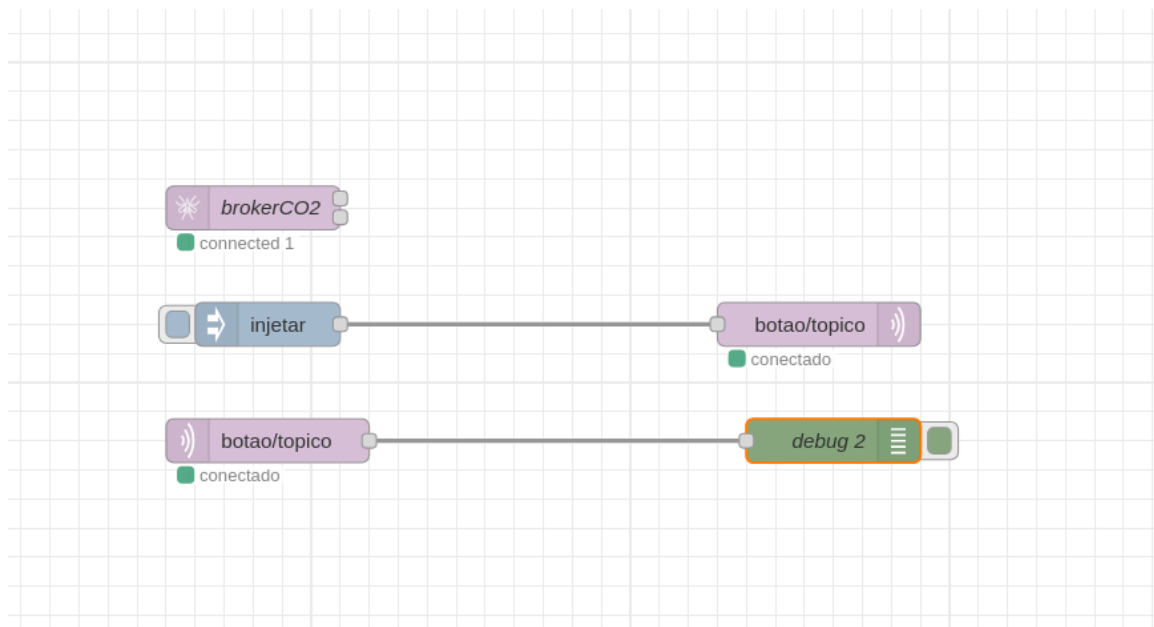
Durante o desenvolvimento do projeto, enfrentamos desafios como a configuração do broker MQTT e a integração com o Grafana. Entretanto, conseguimos implementar um sistema funcional, entendendo na prática o funcionamento do protocolo MQTT e a construção de dashboards. Como melhorias futuras, sugerimos a implementação de banco de dados para armazenar os dados historicamente, uso de múltiplos sensores e adição de notificações automatizadas.

5. Apêndice

5.1 Código da ESP

<https://github.com/grmp13/redes/tree/main/Firmware/src/main.c>

5.2 Configuração do Node-RED



5.3 Configuração do Broker

Editar aedes broker nó

Deletar Cancelar Feito

Propriedades

Nome brokerCO2

Connection Security

MQTT port 1883

WS bind port

WS port Enter Websocket port. Leave blank to disable Websocket support

☐ Enable secure (SSL/TLS) connection

Persistence Memory

Habilitar

5.4 Configuração do MQTT IN

Editar mqtt in nó

Deletar Cancelar Feito

Propriedades

Servidor mqttinCO2

Ação Assinar um tópico único

Tópico botao/topico

QoS 0

Saída auto-deteccção(objeto JSON, cadeia de caracteres ou armazenamento temporário anali: v

Nome Nome

Habilitar

5.5 Configuração do MQTT OUT

The screenshot shows the Node-RED interface with the 'Fluxo 1' workspace. The left sidebar contains a palette with nodes categorized under 'comum' (inject, debug, complete, catch, status, link in, link call, link out, comment) and 'função' (function, switch, change). The workspace contains three nodes: 'brokerCO2' (MQTT Broker), 'injetar' (Inject), and 'botao/topico' (MQTT Out). The 'botao/topico' node is selected, and the 'Editar mqtt out nó' dialog is open. The dialog has tabs for 'Propriedades' (selected), 'Mensagem', and 'Depurar'. The 'Propriedades' tab shows the following configuration:

- Servidor: mqttinCO2
- Tópico: botao/topico
- QoS: 0
- Reter: ☐
- Nome: Nome

A yellow tip box states: 'Dica: deixe o tópico, qos ou retenha em branco se quiser defini-los por meio das propriedades da mensagem.' At the bottom of the dialog is a 'Habilitar' checkbox.

5.6 Configuração do Debug vindo do output

This screenshot shows the same Node-RED workspace as the previous one, but with the 'Editar debug nó' dialog open for the 'debug 2' node. The 'Propriedades' tab shows the following configuration:

- Saída: msg.payload
- Para: ☒ janela de depuração, ☐ console do sistema, ☐ estado do nó (32 caracteres)
- Nome: debug 2

At the bottom is a 'Habilitar' checkbox. To the right of the dialog, the 'depurar' (debug) console is visible, showing a list of debug messages. Each message includes a timestamp, the node name, and the payload. The messages are as follows:

Timestamp	Node	Payload
04/06/2025, 01:24:44	debug 2	
04/06/2025, 01:24:45	debug 2	msg.payload : number
04/06/2025, 01:24:45	debug 2	msg.payload : number
04/06/2025, 01:24:45	debug 2	msg.payload : number
04/06/2025, 01:24:45	debug 2	msg.payload : number
04/06/2025, 01:24:45	debug 2	msg.payload : number
04/06/2025, 01:24:46	debug 2	msg.payload : number
04/06/2025, 01:24:46	debug 2	msg.payload : number
04/06/2025, 01:24:46	debug 2	msg.payload : number
04/06/2025, 01:24:46	debug 2	msg.payload : number

5.7 Configuração do MQTT no Grafana

Name ⓘ grafana-mqtt-datasource Default ☒

Before you can use the MQTT data source, you must configure it below or in the config file. For detailed instructions, [view the documentation](#).

Fields marked with * are required

Connection

URI *

Authentication

Username

Password


 Reset





Use TLS Client Auth
Enables TLS authentication using client cert configured in secure json data.
☐

Skip TLS Verification
When enabled, skips verification of the MQTT server's TLS certificate chain and host name.
☒

With CA Cert
Needed for verifying servers with self-signed TLS Certs.
☐

5.7 Configuração do Grafana

Data source  grafana-mqtt-datasol ⓘ > Q... MD = auto = 550 Interval = 30s Query inspector

▼ A (grafana-mqtt-datasource)    

Topic

+ Add query + Expression