

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS

ESCOLA POLITÉCNICA / ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Trabalho Final de Curso

FLAVIO BONIFACIO DA SILVA

INTEGRAÇÃO DE SISTEMAS OT/IT: UMA ABORDAGEM PRÁTICA UTILIZANDO NODE-RED, MYSQL E GRAFANA NA INDÚSTRIA 4.0

Trabalho Final de Curso como parte dos requisitos para obtenção do título de bacharel em Engenharia de Controle e Automação apresentado à Pontifícia Universidade Católica de Goiás.

BANCA EXAMINADORA:

Prof. Dr. CASSIO HIDEKI FUJISAWA – Orientador.

Prof. Dr. WANDERSON RAINER HILARIO ARAUJO – Banca examinadora.

Prof. Dr. ANTONIO MARCOS MELO MEDEIROS – Banca examinadora

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS-PUC GOIÁS.

Goiânia, 28 de novembro de 2025.

INTEGRAÇÃO DE SISTEMAS OT/IT: UMA ABORDAGEM PRÁTICA UTILIZANDO NODE-RED, MYSQL E GRAFANA NA INDÚSTRIA 4.0

1st Flavio Bonifacio da Silva

Engenharia de Controle e Automação

Pontificia Universidade Catolica de

Goiás (PUC-GO)

Goiânia-GO, Brasil

flaviobonifaciadasilva123@gmail.com

2st Cassio Hideki Fujisawa

Engenharia de Controle e Automação

Pontificia Universidade Catolica de

Goiás (PUC-GO)

Goiânia-GO, Brasil

hideki@pucgoias.edu.br

I. ABSTRACT

Due to the wide variety of assets in the industry, this project developed a system to monitor electric motors that are already integrated into the automation network. The solution was built entirely with free and open-source software: Node-RED, MySQL, and Grafana. All these tools were centralized on a single physical server, responsible for collecting and analyzing data from the factory floor.

The system communicates with PLCs (Programmable Logic Controllers) and leverages data structures already standardized by manufacturers, such as Rockwell's PlantPAX and Siemens' PCS 7. The focus of the monitoring is the information that these controllers exchange with the frequency inverters in the field. In this way, it is possible to collect all the data and generate historical data on the server without the need to change the PLC programming and without software license costs.

II. KEYWORDS

Industrial Monitoring, Node-RED, MySQL, Grafana

III. RESUMO

Devido à grande diversidade de ativos na indústria, este projeto desenvolveu um sistema para monitorar motores elétricos que já estão integrados à rede de automação. A solução foi construída inteiramente com softwares de código aberto e gratuitos: Node-RED, MySQL e Grafana. Todas essas ferramentas foram centralizadas em um único servidor físico, responsável por coletar e analisar os dados do chão de fábrica.

O sistema se comunica com os PLCs (Controladores Lógicos Programáveis) e aproveita as estruturas de dados já padronizadas pelos fabricantes, como o PlantPAX da Rockwell e o PCS 7 da Siemens. O foco do monitoramento são as informações que esses controladores trocam com os inversores de frequência em campo. Dessa forma, é possível coletar todos os dados e gerar históricos no servidor sem a necessidade de alterar a programação dos PLCs e sem custos com licenças de software.

IV. PALAVRAS-CHAVE

Monitoramento Industrial, Node-RED, MySQL, Grafana.

V. INTRODUÇÃO

O monitoramento de ativos industriais, como motores e equipamentos de produção, é um pilar estratégico para a excelência operacional. A capacidade de acompanhar em tempo real as horas de operação e produção não apenas viabiliza a manutenção preditiva, mas também fornece insights valiosos para a otimização de processos e o aumento da eficiência geral dos equipamentos (OEE). O objetivo deste documento é introduzir uma arquitetura de sistema robusta, flexível e de baixo custo para o monitoramento de motores, fundamentado em tecnologias de código aberto amplamente adotadas pela indústria.

A solução proposta integra três componentes principais: Node-RED, uma plataforma de programação visual que atua como o elo de comunicação entre o chão de fábrica e a infraestrutura de TI; MySQL, um sistema de gerenciamento de banco de dados relacional que serve como repositório central para os dados históricos; e Grafana, uma plataforma de visualização que transforma dados brutos em dashboards interativos e acionáveis. Juntos, esses componentes formam um ecossistema coeso capaz de coletar dados operacionais de PLC (Programmable Logic Controller) Rockwell ou Siemens, armazená-los de forma segura e apresentá-los de maneira intuitiva aos usuários finais.

VI. ARQUITETURA DO SISTEMA PROPOSTO

Com o aumento da conectividade no chão de fábrica (IIoT), a identificação correta dos dispositivos e o monitoramento do tráfego de rede tornam-se cruciais. Segundo Sheng *et al.* [14], o *fingerprinting* de tráfego de rede é uma técnica essencial para garantir a segurança e a integridade dos dados em ambientes industriais modernos, justificando a

importância de monitorar variáveis críticas como os dados de motores.

A seguir, será detalhada a arquitetura do sistema e o papel de cada tecnologia na construção desta solução de monitoramento industrial.

Uma arquitetura de sistema bem definida é fundamental para garantir a interoperabilidade entre as tecnologias de automação (OT) e as tecnologias da informação (IT), criando uma ponte eficaz entre o chão de fábrica e os sistemas de gestão. Esta seção detalha o papel de cada componente tecnológico — Node-RED, MySQL e Grafana — e descreve como eles se integram para formar uma solução coesa, escalável e eficiente para o monitoramento de motores.

A. Visão Geral do Fluxo de Dados

O fluxo de dados do sistema proposto segue uma sequência lógica e otimizada, desde a coleta no chão de fábrica até a visualização pelo usuário final. O processo pode ser resumido nas seguintes etapas:

- a) Os dados operacionais do motor, como status de funcionamento, corrente e outras variáveis relevantes, residem em tags de memória específicas de um PLC Rockwell, que opera na rede industrial.
- b) O Node-RED, atuando como um gateway de dados, estabelece comunicação com o PLC e monitora continuamente os valores dessas tags.
- c) A cada intervalo de tempo predefinido (por exemplo, 15 segundos), o Node-RED coleta os dados mais recentes do PLC e os organiza em um formato estruturado.
- d) Em seguida, o Node-RED constrói e executa uma instrução SQL (Structured Query Language) para inserir os dados coletados em uma tabela específica dentro de um banco de dados MySQL, criando um registro histórico com carimbo de data e hora.
- e) A plataforma Grafana, que está configurada para utilizar o MySQL como sua fonte de dados (Data Source), consulta periodicamente o banco de dados para buscar as informações mais recentes e exibi-las em dashboards interativos e de fácil compreensão para os usuários.

Este fluxo garante que os dados do processo sejam capturados, armazenados e visualizados de forma automatizada e confiável, transformando informações operacionais brutas em inteligência de negócio. O *Fluxograma 1* apresenta de forma clara as etapas de comunicação do projeto, sendo as informações saindo do PLC e indo direto para o Node-RED que vai tratar os dados e enviar para o MySQL que guarda os dados do PLC e por fim se tem o grafana que é responsável por mostrar os dados coletados de forma gráfica e visual para o usuário.

Fluxograma 1 Processo de coleta dos dados ate a apresentação grafica



Fonte: elaborada pelo próprio autor.

VII. NODE-RED

Node-RED desempenha a função estratégica de orquestrador do fluxo de dados. Trata-se de uma ferramenta de programação visual de baixo código (low-code) projetada para aplicações orientadas a eventos, o que a torna ideal para cenários de Internet Industrial das Coisas (IIoT). Neste sistema, sua importância reside em sua capacidade de atuar como o elo que conecta o mundo da automação industrial (OT), representado pelo PLC, à infraestrutura de tecnologia da informação (IT), representada pelo banco de dados, uma função que o estabelece como um "gestor de protocolos" ideal para cenários de IIoT [4].

Sua arquitetura baseada em nós que permite interagir nativamente com diversos protocolos industriais, como PCCC ou EtherNet/IP para PLCs da família Rockwell, e, ao mesmo tempo, comunicar-se com sistemas de TI, como bancos de dados e APIs. Além disso, o Node-RED é multiplataforma e consome poucos recursos computacionais, podendo ser executado de forma eficiente em dispositivos de baixo custo, como um Raspberry Pi, o que o torna uma solução versátil e econômica para a criação de gateways de dados industriais.

Para iniciar no Node-RED foi instalado diretamente pela página do "Node.js" que fornece suporte gratuito para instalação [5].

Observa-se *Figura 1* a página de instalação do Node-RED, podendo escolher até para qual sistema operacional vai ser executado, no caso do projeto proposto foi escolhido para o Windows. As instruções de instalação estão na mesma página Web que esta o instalador, o procedimento consiste em o que está sendo instalado e seguir os passo a passo.

Figura 1 Página de instalação NODE-RED



Fonte: <https://nodejs.org/pt/download>

Após a instalação, pode-se verificar a versão que foi instalada ou se já tiver instalado anteriormente se tem a consultar da versão, tudo isso através do “Prompt de Comando” presente dentro do Windows como na.

Executando o comando “node-RED” diretamente no prompt como na vai inicializar o Node RED, para se ter acesso a ele tem que se observar a janela do prompt que vai mostrar um endereço IP da página executado dentro da própria máquina, sendo rodada diretamente no “http://localhost:1880/” com porta de comunicação padrão sendo a “:1880”, para ter acesso é só digitar o IP que retorna do próprio prompt no navegador “https://xxx.xxx.xxx.xxx:1880/” na o IP que aparece é “https://127.0.0.1:1880/”, observa-se na Figura 2 como é feito o acesso usando o navegador digitando “localhost:1880/”.

Figura 2 localhost:1880

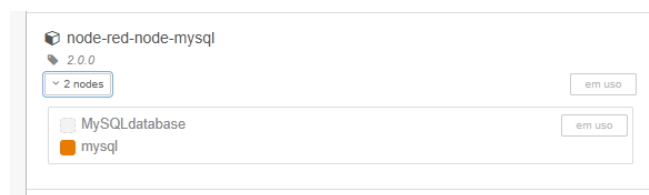


Fonte: elaborada pelo próprio autor.

Após acessar a página do Node-RED diretamente pelo navegador foi instalado algumas bibliotecas para comunicação com o banco de dados e o PLC.

Para se comunicar o Node-RED com o banco de dados foi utilizado a biblioteca “node-RED-node-mysql” como mostrado na Figura 3.

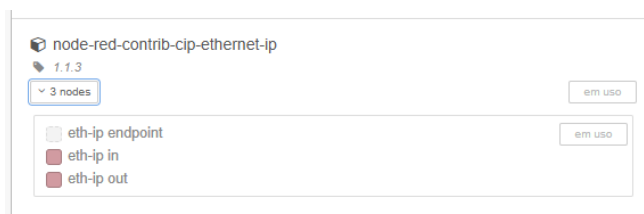
Figura 3 node-red-node-mysql



Fonte: elaborada pelo próprio autor.

Para se comunicar com o PLC Rockwell que foi usando como instrumento de estudos foi utilizado a biblioteca “node-red-contrib-cip-ethernet-ip” como mostrado na Figura 4. Essa biblioteca se tem a possibilidade de monitorar variáveis dentro do PLC Rockwell e escrever valores dentro de variáveis.

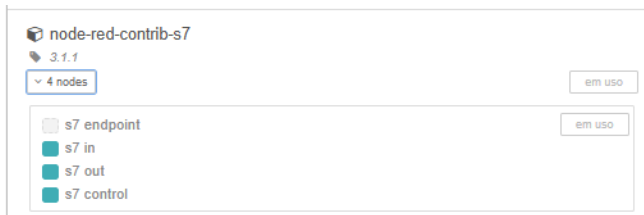
Figura 4 node-red-contrib-cip-ethernet-ip



Fonte: elaborada pelo próprio autor.

Para se comunicar com o PLC Siemens que foi o usando como instrumento de estudos foi utilizado a biblioteca “node-red-contrib-s7” como mostrado na Figura 5. Essa biblioteca se tem a possibilidade de monitorar variáveis dentro do PLC e escrever valores dentro de variáveis.

Figura 5 node-red-contrib-s7

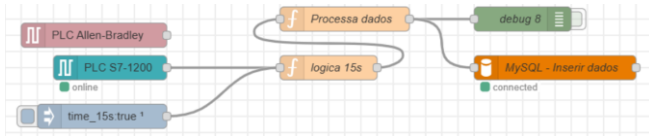


Fonte: elaborada pelo próprio autor.

Com essas bibliotecas foi possível se comunicar com o PLC na rede e o banco de dados, tendo as bibliotecas de comunicação de cada um já configurado foi montado a estrutura dentro do node-RED. Na Figura 6 se tem a entrada de dados dos blocos “PLC”, podendo alternar entre um ou outro a depender do modelo em campo, também se tem um bloco que é para registrar valores a cada 15s e envia para a

próxima etapa que trata os dados e formata para só no final enviar para o banco de dados.

Figura 6 Estrutura de blocos dentro do Node-RED



Fonte: elaborada pelo próprio autor.

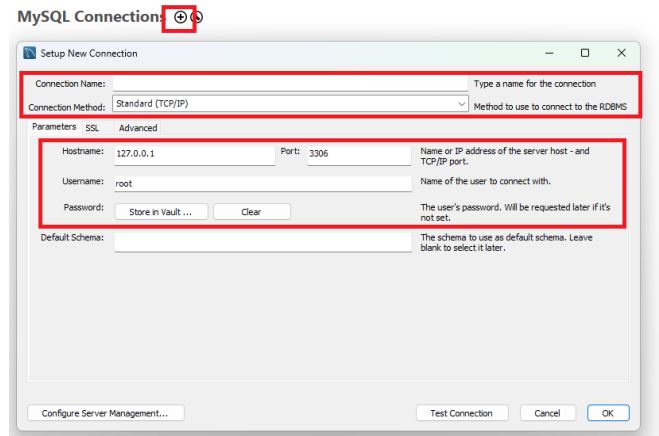
VIII. MYSQL

O MySQL foi escolhido como o sistema de gerenciamento de banco de dados relacional (RDBMS) de código aberto para este projeto, cumprindo o papel de repositório central e persistente para os dados de horímetros. Sua adequação para esta aplicação é sustentada por sua comprovada confiabilidade, escalabilidade e desempenho, sendo a tecnologia por trás de aplicações de alto tráfego como as do Facebook e Uber. Nesta arquitetura, o MySQL armazena os dados coletados pelo Node-RED de forma estruturada em tabelas, garantindo a integridade e a consistência das informações através de seu suporte a transações ACID (Atomicidade, Consistência, Isolamento e Durabilidade). Ao criar um repositório histórico confiável e organizado, o MySQL não apenas assegura a Figura 1 Página de instalação NODE-RED rastreabilidade das operações, mas também fornece a base de dados essencial para análises futuras, relatórios de desempenho e algoritmos de manutenção preventiva.

O MySQL atendeu a demanda do projeto para tráfego de dados por ser um projeto simples com poucas requisições simultâneas. Também tendo um ótimo suporte por ser muito utilizado e bem conhecido, os problemas encontrados foram solucionados de forma rápida e simples.

Após instalação e pré configuração do MySQL foi iniciado o processo de criar um banco, configurar o “Hostname” que é o endereço do servidor e sua porta que já vem como padrão “:3306”. Foi utilizado as configurações padrões do MySQL e estabelecido só uma nova conexão como mostrado na Figura 7, mantendo o Hostname e Porta como “127.0.0.1:3306”.

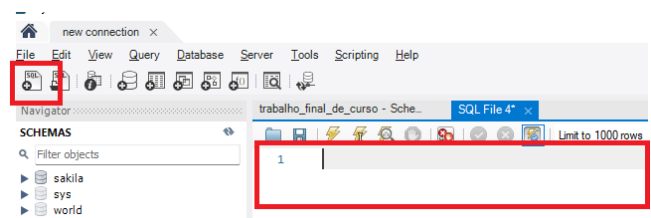
Figura 7 Nova Conexão no MySQL



Fonte: elaborada pelo próprio autor.

Para usuário/senha foi mantido os padrões como root/admin, que vai ser utilizado na hora de fazer requisições de dados pelo grafana e escrever dados nas tabelas pelo Node-RED. Para navegação interna dentro do banco de dados foi escolhido utilizar o interface de comando SQL disponível dentro do software por ser mais prático, tendo só de executar os comandos em linguagem SQL, na Figura 8 se tem por onde abre uma nova janela que executa comando em SQL.

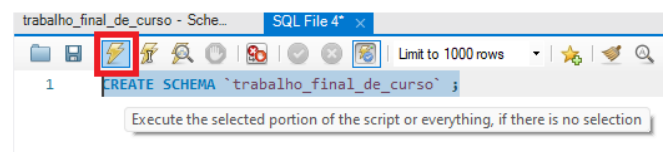
Figura 8 Abrindo uma nova janela em para executar em SQL



Fonte: elaborada pelo próprio autor.

Para se criar um Banco de dados dedicado ao projeto foi utilizado os comandos “CREATE SCHEMA ‘trabalho_final_de_curso’ ;” sendo “CREATE SCHEMA” para criar o banco com o nome “ ‘trabalho_final_de_curso’ ” Após escrever os comandos em SQL se tem de executar como mostrado na Figura 9 que se tem a criação do banco. Foi selecionado o comando e executado.

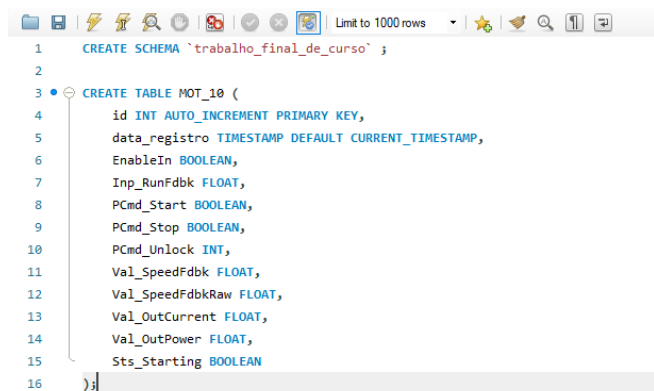
Figura 9 executando comandos dentro do SQL



Fonte: elaborada pelo próprio autor.

Após criar o banco se tem a criação de suas tabelas onde se tem as variáveis para cada motor como mostrado na *Figura 10* que é referente ao motor com identificação em campo “MOT_10”.

Figura 10 Criando Tabela de motor para o "Motor 10"



Fonte: elaborada pelo próprio autor.

Para se definir quais tags para monitorar esse motor foi analisado sua estrutura em lógica que esta em execução dentro do PLC. Quando se adicionar um motor dentro de um PLC independente se for de fabricante Siemens ou Rockwell se tem alguns padrões de qualidades que os próprios fabricantes dão suporte. O desenvolvimento da lógica de controle nos CLPs requer conhecimento dos padrões, por se uma boa prática de programação, pois já foram consolidados com o tempo e tendo suporte direto do fabricante e fornecimento pelo mesmo.

IX. GRAFANA

Grafana é a camada de apresentação da solução, funcionando como a plataforma de código aberto para consulta, visualização e alerta sobre dados. Sua função no sistema é transformar os dados brutos e históricos armazenados no MySQL em insights visuais e acionáveis para as equipes de operação e engenharia.

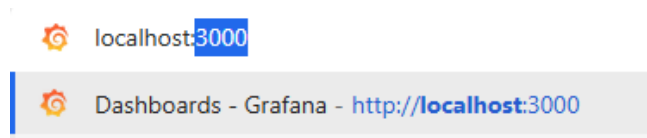
A principal vantagem do Grafana é sua capacidade de se conectar diretamente a diversas fontes de dados, incluindo o MySQL. Através de uma interface intuitiva, os usuários podem criar dashboards personalizados com uma vasta gama de painéis de visualização, como gráficos de séries temporais (Time series) para analisar tendências, medidores (Gauge) para monitorar valores instantâneos e tabelas (Table) para exibir dados detalhados. Essa flexibilidade permite que os usuários monitorem o status dos motores em tempo real, analisem o comportamento histórico dos equipamentos e tomem decisões mais informadas para otimizar a performance da planta.

O grafana é um software dedicado a parte de análise de dados por meio de gráficos, tabelas, linhas do tempo dentre muitas outras formas e até mesmo utilizando APIs para melhorar essa exibição de dados.

Sua instalação e execução é semelhante ao Node-RED, basta seguir para o [link](#) de instalação [4].

Após instalar o grafana é só executado diretamente no navegador igual o Node-RED, só mudando a porta que já vem padrão “:3000”, como mostrado na *Figura 11* basta digitar “localhost:3000” que já vai abrir o grafana.

Figura 11 abrindo o grafana no navegador

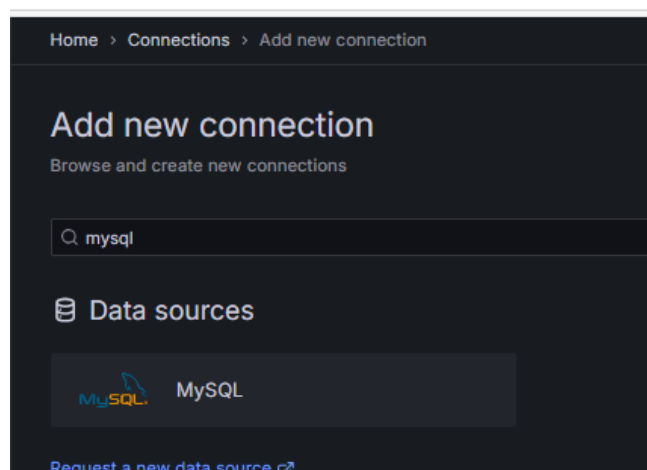


Fonte: elaborada pelo próprio autor.

No primeiro acesso ele vai pedir login com nome de usuário e senha que vem padrão como usuário “admin” e senha “admin”, após isso vai pedir para criar uma senha.

Essas primeiras pré configurações tem de se configurar a conexão com o banco de dados MySQL, dentro do grafana essas ferramentas são chamadas de “Plugins” e foi utilizado o plugin do MySQL como mostrado na *Figura 12* para se conectar diretamente nele.

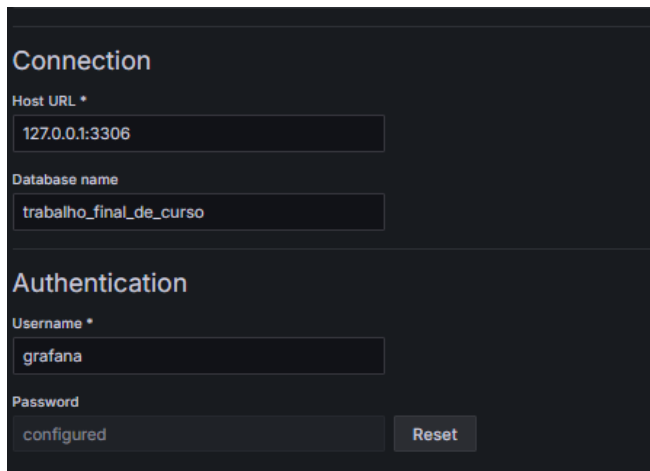
Figura 12 Plugin MySQL



Fonte: elaborada pelo próprio autor.

Para configurar esse plugin tem de se utilizar as informações do novo banco de dados criado com nome “trabalho_final_de_curso” como mostrado na *Figura 13* se tem os carros a serem preenchidos com informações do MySQL. Para o usuário foi criado um novo pois o user “root” não foi possível usar mesmo estando na mesma máquina, o banco de dados e o grafana. Foi criado um usuário dentro do banco de dados via SQL com todas as permissões para esse novo usuário.

Figura 13 Configurando conexão com MySQL



Fonte: elaborada pelo próprio autor.

X. DETALHAMENTO DO FLUXO DE IMPLEMENTAÇÃO

A utilização de arquiteturas baseadas em web para monitoramento industrial tem crescido significativamente. Estudos recentes, como os de Christopher *et al.* [12], demonstram a eficácia da integração entre Node-RED e bancos de dados SQL para coleta de dados em tempo real. Similarmente, Chen *et al.* [13] validam o uso de Node-RED e protocolos industriais (como Modbus e MQTT) em plataformas Linux embarcadas para controle ambiental e de iluminação, reforçando a flexibilidade dessas ferramentas *open source*.

Esta seção detalha as etapas práticas de configuração e comunicação entre os componentes da arquitetura, demonstrando a viabilidade técnica e a simplicidade relativa da implementação da solução proposta.

A. Coleta de Dados do PLC com Node-RED

O primeiro passo é estabelecer a comunicação entre o Node-RED e o PLC Rockwell. Para isso, utiliza-se um nó de contribuição específico para o protocolo de comunicação da Rockwell (como o `node-red-contrib-ethernet-IP`). Este nó é configurado com as informações de rede do PLC, como seu endereço IP e o slot da CPU no chassis (por exemplo, slot 3). Dentro da configuração do nó, são especificadas as tags de memória que devem ser lidas. Embora a lógica interna programada no PLC não seja o foco deste documento, é crucial entender que os valores de interesse estão dentro de uma estrutura já pré-definida pela própria Rockwell quando se tem motores no escopo do projeto, já estão disponíveis nessas tags de memória como resultado do programa do controlador. O Node-RED é configurado para ler essas tags em intervalos regulares, trazendo os dados do chão de fábrica para o ambiente de processamento.

B. Processamento e Armazenamento no MySQL

Uma vez que os dados são lidos do PLC, o Node-RED os processa e os envia para o banco de dados MySQL. Esse processo é realizado em uma sequência de nós interligados:

- 1) **Formatação dos Dados:** Os dados brutos provenientes do PLC são recebidos como uma mensagem (msg) no fluxo do Node-RED. Nós como o “Change” ou “Function” são utilizados para extrair os valores relevantes e prepará-los para a inserção no banco de dados.
- 2) **Criação da Query SQL:** para escrever de uma forma que o banco de dados consiga interpretar as informações tem de se fazer uma estrutura com as tags de forma que se interprete em linguagem SQL. O nó Template é empregado para construir a instrução SQL parametrizada, que é armazenada na propriedade `msg.topic`. Por exemplo: “INSERT INTO tabela_horimetros (timestamp, valor_corrente) VALUES (“+HORA+”, “+CORRENTE+)”. Sendo a variável “HORA” o instante que foi feita a leitura e “CORRENTE” a variável que armazena o valor de corrente instantânea do MOT_10.
- 3) **Execução no Banco de Dados:** Um nó Function prepara a propriedade `msg.payload` como um array contendo os valores na ordem correta para substituir os marcadores de posição (ex: [timestamp, valor_corrente]). Em seguida, o nó `node-red-node-mysql`, pré-configurado com os detalhes da conexão ao servidor MySQL, espera um intervalo a cada 15 segundos e recebe a mensagem, associa essa mesma mensagem a uma query do “msg.topic” aos valores do “msg.payload” e executa a transação de forma segura, persistindo os dados do horímetro na tabela correspondente.

C. Visualização dos Dados no Grafana

A etapa final consiste em configurar o Grafana para se conectar ao MySQL e exibir os dados em um dashboard. O processo é direto e realizado através da interface web do Grafana:

- 1) **Configuração da Fonte de Dados:** O primeiro passo é adicionar o MySQL como uma nova fonte de dados (Data Source). No painel de configuração do Grafana, seleciona-se MySQL e preenche-se as informações de conexão necessárias, como o endereço do host, o nome do banco de dados, o usuário e a senha. Um teste de conexão confirma que o Grafana consegue se comunicar com o banco de dados.
- 2) **Criação do Dashboard:** Com a fonte de dados estabelecida, um novo dashboard é criado. Dentro do dashboard, o usuário pode adicionar múltiplos painéis, cada um destinado a uma visualização específica (um gráfico, um medidor, uma tabela, etc.).
- 3) **Consulta e Exibição:** Para cada painel adicionado, uma consulta SQL é escrita diretamente na interface do editor de painel do Grafana. Esta consulta seleciona os dados que devem ser exibidos. Por exemplo, uma consulta como `SELECT timestamp AS "time", valor_corrente`

FROM tabela_horímetros WHERE
 _timeFilter(timestamp) buscará os dados de corrente. O
 macro \$__timeFilter(timestamp) é um recurso poderoso
 do Grafana que ajusta dinamicamente a
 cláusula WHERE da consulta com base no intervalo de
 tempo selecionado pelo usuário na interface do
 dashboard. O Grafana então executa essa consulta no
 banco de dados MySQL e renderiza o resultado no tipo
 de visualização escolhido.

XI. BENEFÍCIOS E APLICAÇÕES DA SOLUÇÃO

A implementação do sistema proposto oferece um valor agregado significativo para as operações industriais, traduzindo-se em benefícios tangíveis.

A. Visibilidade operacional em tempo real

Os dashboards dinâmicos no Grafana fornecem uma visão instantânea e clara do status dos equipamentos. As equipes de operação podem acompanhar o desempenho dos motores, identificar paradas não programadas e monitorar variáveis críticas em tempo real, permitindo uma resposta mais rápida a eventos no chão de fábrica.

B. Base de Dados para Análise Histórica

O armazenamento persistente dos dados no MySQL cria um repositório histórico confiável. Essa base de dados é fundamental para a análise de tendências de longo prazo, identificação de padrões de falha, otimização de ciclos de produção e para auditorias de processo, fornecendo o subsídio necessário para uma gestão baseada em evidências.

C. Baixo Custo e Flexibilidade

A utilização exclusiva de tecnologias de código aberto (Node-RED, MySQL e Grafana) elimina os custos associados ao licenciamento de software proprietário. Além disso, essa abordagem oferece alta flexibilidade para customização e expansão do sistema, permitindo que a solução seja adaptada às necessidades específicas da planta sem depender de um único fornecedor.

D. Interoperabilidade e Escalabilidade

A arquitetura modular e desacoplada promove a interoperabilidade entre diferentes sistemas e equipamentos. O Node-RED pode ser facilmente expandido para se conectar a outras fontes de dados enquanto o MySQL e o Grafana são capazes de escalar para lidar com grandes volumes de dados, permitindo que o sistema cresça junto com a operação.

XII. RESULTADOS

Foi possível testar em campo com o ambiente prático, o projeto tendo várias frentes e uma delas foi o registro de funcionamento e monitoramento de motores através da rede industrial. Como mostrado na *Figura 14* se tem a imagem do servidor instalado diretamente em campo para realizar testes de comunicação em campo.

Figura 14 Servidor do Node-red, MySQL e Grafana.



Fonte: elaborada pelo próprio autor.

Se teve todo um cuidado com a instalação do servidor para que não houvesse nenhum conflito com a rede presente dentro da indústria como conflitos de endereços IP no dia da instalação e para isso foi alinhado diretamente com a equipe de TI da própria indústria.

Na *Figura 15* se tem os dados coletados previamente de um motor sendo o “MOT_10” só para fins de validação de toda a estrutura em campo.

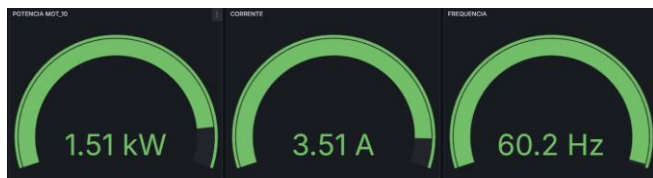
Figura 15 Dashboards do grafana



Fonte: elaborada pelo próprio autor.

É possível ver os valores que estão sendo ligados diretamente do banco de dados, os valores do gráfico são mostrados na linha do tempo e os da parte superior sendo medidores referentes aos últimos valores lidos a cada 15 segundos. Se tem a possibilidade de acompanhar em tempo real os últimos valores coletados através dos “Gauges” mostrados na *Figura 16* com informações em tempo real de potência, Corrente e Frequência da última leitura que foi registrada no banco de dados.

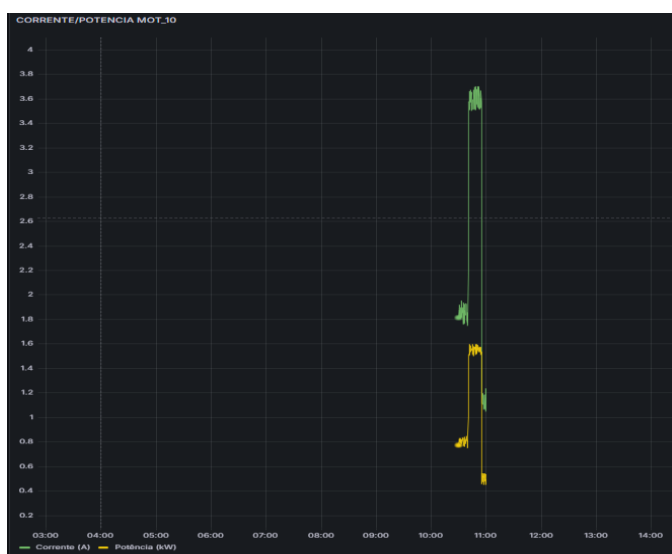
Figura 16 Gauges.



Fonte: elaborada pelo próprio autor.

Para o monitoramento de algumas variáveis estratégicas feita com relação ao tempo se tem as análises gráficas na *Figura 17*. Com a análise gráfica foi possível monitorar picos de aceleração e desaceleração, os dados coletados para a análise gráfica foram com aceleração em 50%, 100% e 30% feitos em campo e monitorado pela análise gráfica.

Figura 17 Gráfico da Corrente e Potência no tempo.



Fonte: elaborada pelo próprio autor.

O servidor monitora toda a estrutura do motor apontado pelo Node-red e armazena no banco, por monitorar toda a estrutura se tem varios dados que se tem um grau de importância entre eles. Como estratégia para facilitar o usuário na hora de analisar esses dados não foram colocados todos os dados nos gráficos ou nos *gauges*, mas foi colocado uma tabela no final do projeto de forma que seja mostrado todos os dados coletados da estrutura presente dentro do PLC e sendo utilizado para análises mais detalhadas, com um foco para a equipe da manutenção.

CONCLUSÃO

A integração sinérgica entre Node-RED, MySQL e Grafana constitui uma solução moderna, eficaz e economicamente acessível para o desafio do monitoramento de horímetros em ambientes industriais. A arquitetura proposta demonstra como tecnologias de código aberto podem ser combinadas para criar um sistema robusto que preenche a lacuna entre a tecnologia de automação (OT) e a tecnologia da informação (IT). Ao automatizar a coleta, o armazenamento e a visualização de

dados operacionais, a implementação deste sistema capacita as organizações a transitarem para uma tomada de decisões verdadeiramente baseada em dados, resultando em melhorias diretas na eficiência operacional, na confiabilidade dos ativos e na competitividade do negócio.

XIII. REFERÊNCIAS

BIBLIOGRAFIA

- [1] NODE-RED.: Node-RED Documentation. NODE-RED. Acesso 20 de outubro de 2025. Disponível em: <https://nodered.org/docs/>.
- [2] MYSQL.: MySQL Reference Manual. MYSQL. Acesso 20 de outubro de 2025. Disponível em: <https://dev.mysql.com/doc/>.
- [3] GRAFANA LABS.: Grafana Documentation. GRAFANA LABS. Acesso 20 de outubro de 2025. Disponível em: <https://grafana.com/docs/>.
- [4] GRAFANA LABS.: Download Grafana. GRAFANA LABS. Acesso 20 de outubro de 2025. Disponível em: <https://grafana.com/grafana/download?platform=windows>
- [5] Ventura, J.T., Puente, A.H.R., García, J.R.H.: Rendimiento para la interoperabilidad entre Raspberry pi, ESP8266 y PLC con Node-RED para el IIoT. Ingenius, Revista de Ciencia y Tecnología; 29, 90-97; (2023).
- [6] NODE.JS.: Node-RED instalador. NODE.JS. Acesso 20 de outubro de 2025. Disponível em: <https://nodejs.org/pt>.
- [7] Rockwell Automation.: PlantPAX Distributed Control System. Rockwell Automation. Acesso 20 de outubro de 2025. Disponível em: <https://www.rockwellautomation.com/en-us/capabilities/process-solutions/process-systems/plantpax-distributed-control-system.html>.
- [8] Siemens.: Libraries in the TIA Portal. Siemens Industry Online Support. Acesso 20 de outubro de 2025. Disponível em: <https://support.industry.siemens.com/cs/document/109738702/libraries-in-the-tia-portal?dti=0&lc=en-BR>.
- [9] Siemens.: STEP7 V1x MODBUS RTU. Siemens Industry Online Support. Acesso 20 de outubro de 2025. Disponível em: https://cache.industry.siemens.com/dl/files/272/109755272/att_940616/v1/STEP7_V1x_MODBUS_RTU.pdf.
- [10] Node-RED.: node-red-contrib-s7. Node-RED Flow Library. Acesso 20 de outubro de 2025. Disponível em: <https://flows.nodered.org/node/node-red-contrib-s7>.
- [11] Node-RED.: node-red-contrib-cip-ethernet-ip. Node-RED Flow Library. Acesso 20 de outubro de 2025. Disponível em: <https://flows.nodered.org/node/node-red-contrib-cip-ethernet-ip>.
- [12] N. Christoper, A. A. Hadi, S. Razali, e H. Manap, "Web-Based Data Monitoring for IIoT using Node-RED and Raspberry Pi," in *2025 IEEE 8th International Conference on Electrical, Control and Computer Engineering (InECCE)*, Kuantan, Malaysia, 2025, pp. 391-396.
- [13] C.-Y. Chen, S.-H. Wu, B.-W. Huang, C.-H. Huang, e C.-F. Yang, "Web-based Internet of Things on environmental and lighting control and monitoring system using node-RED, MQTT and Modbus communications within embedded Linux platform," *Internet of Things*, vol. 27, p. 101305, 2024.

- [14] C. Sheng *et al.*, "Network Traffic Fingerprinting for IIoT Device Identification: A Survey," *IEEE Transactions on Industrial Informatics*, vol. 21, no. 5, pp. 3541-3554, maio 2025.
- [15] M. Diaz-Cacho, P. Falcón, J. M. Acevedo, e A. P. Domínguez, "Educational project for remote access to industrial networks," in *2024 IEEE Global Engineering Education Conference (EDUCON)*, Kos, Grécia, 2024.

RESOLUÇÃO nº 038/2020 – CEPE

ANEXO I

APÊNDICE ao TCC

Termo de autorização de publicação de produção acadêmica

O estudante FLAVIO BONOIFACIO DA SILVA do Curso de ENGENHARIA DE CONTROLE E AUTOMAÇÃO matrícula 20211011800175, telefone: 62 996839649 e-mail flaviobonifaciодasilva123@gmail.com, na qualidade de titular dos direitos autorais, em consonância com a Lei nº 9.610/98 (Lei dos Direitos do Autor), autoriza a Pontifícia Universidade Católica de Goiás (PUC Goiás) a disponibilizar o Trabalho de Conclusão de Curso intitulado DESENVOLVIMENTO DE UM SISTEMA DE MONITORAMENTO DE HORÍMETROS UTILIZANDO NODE-RED, MYSQL E GRAFANA, gratuitamente, sem ressarcimento dos direitos autorais, por 5 (cinco) anos, conforme permissões do documento, em meio eletrônico, na rede mundial de computadores, no formato especificado (Texto(PDF); Imagem (GIF ou JPEG); Som (WAVE, MPEG, AIFF, SND); Vídeo (MPEG, MWV, AVI, QT); outros, específicos da área; para fins de leitura e/ou impressão pela internet, a título de divulgação da produção científica gerada nos cursos de graduação da PUC Goiás.

Goiânia, 30 de setembro de 2025.

Assinatura do autor: _____

Nome completo do autor: FLAVIO BONIFACIO DA SILVA

Assinatura do professor-orientador: _____

Nome completo do professor-orientador: