

Campus Recife

Tecnologia em Análise e Desenvolvimento de Sistemas

**Desenvolvimento de Aplicativo para Análise de Padrões de
Vibração de Compressores**

Trabalho de Conclusão de Curso de Graduação

por

João Vitor Gouveia de Lima

Orientador(a): Prof. Meuse Nogueira de Oliveira Júnior, DSc

Recife, 2025

João Vitor Gouveia de Lima

**Desenvolvimento de Aplicativo para Análise de Padrões de Vibração de
Compressores**

Trabalho de Conclusão de Curso apresentado
como requisito parcial para obtenção
do título de Tecnólogo em Análise e
Desenvolvimento de Sistemas, no Instituto
Federal de Educação, Ciência e Tecnologia
de Pernambuco (IFPE), Campus Recife.

Orientador(a): Prof Meuse Nogueira de Oliveira Júnior, DSc

Recife

2025

Ficha catalográfica elaborada pela bibliotecária Danielle Castro da Silva CRB4/1457

L732d
2026

Lima, João Vitor Gouveia de

Desenvolvimento de Aplicativo para Análise de Padrões de Vibração de Compressores / João Vitor Gouveia de Lima. --- Recife: O autor, 2026.

32f. il. Color.

Trabalho de Conclusão (Curso Superior Tecnológico em Análise e Desenvolvimento de Sistemas) – Instituto Federal de Pernambuco, Recife, 2025.

Inclui Referências.

Orientador: Prof. Dr. Meuse Nogueira de Oliveira Júnior.

1. Análise de vibrações. 2. Processamento de sinais. 3. Manutenção preditiva. I. Título. II. Oliveira Júnior, Meuse Nogueira de (orientador). III. Instituto Federal de Pernambuco.

CDD 658.404 (23 ed.)

João Vitor Gouveia de Lima

Desenvolvimento de Aplicativo para Análise de Padrões de Vibração de Compressores

Trabalho de Conclusão de Curso
apresentado, como requisito
parcial para obtenção do título de Tecnólogo
em Análise e Desenvolvimento de Sistemas,
do Instituto Federal de Educação, Ciência e
Tecnologia de Pernambuco – Campus Recife.

Aprovado em: 13 de novembro de 2025.

Banca Examinadora:

Prof. Dr. Meuse Nogueira de Oliveira Júnior (Orientador)
Instituto Federal de Pernambuco (IFPE)

Prof. Me. Tiago Lins Falcão
Instituto Federal de Pernambuco (IFPE)

Prof. Dr. Eduardo Antonio Guimaraes Tavares
Universidade Federal de Pernambuco (UFPE)

Recife

2025

RESUMO

Este trabalho apresenta o desenvolvimento de um aplicativo para análise de padrões de vibração em compressores utilizados em sistemas de climatização. A proposta oferece uma solução acessível, utilizando sensores de baixo custo (na ordem de R\$30,00), para auxiliar na identificação de falhas e na manutenção preditiva desses equipamentos, promovendo maior eficiência operacional e redução de custos de manutenção. O aplicativo é capaz de capturar, processar e analisar os sinais de vibração gerados pelos compressores, fornecendo diagnósticos precisos e em tempo hábil sobre o estado de saúde dos dispositivos. Os sinais de vibração são apresentados por meio de uma interface que permitirá a visualização e a interpretação dos dados de forma intuitiva, assegurando que profissionais da área possam tomar decisões rápidas e informadas.

Palavras-chave: Análise de vibrações; Processamento de sinais; Manutenção preditiva; IoT.

ABSTRACT

This work presents an application for analyzing vibration patterns in compressors used in air conditioning systems. The proposal provides an affordable solution by utilizing low-cost sensors (on the order of BRL 30.00) to assist in fault detection and predictive maintenance of these devices, promoting greater operational efficiency and reduced maintenance costs. The application is capable of capturing, processing, and analyzing vibration signals generated by compressors, delivering accurate and timely diagnostics regarding the health status of the equipment. The vibration signals are displayed through an interface that enables intuitive data visualization and interpretation, ensuring that professionals in the field can make quick and informed decisions.

Keywords: Vibration analysis; Signal processing; Predictive maintenance; IoT.

LISTA DE FIGURAS

| | | |
|-----------|---|----|
| Figura 1 | Emulador de Vibração | 18 |
| Figura 2 | Motor Monofásico | 21 |
| Figura 3 | Gráficos de Análise (aba Todos os Eixos) | 22 |
| Figura 4 | Métricas na Interface (aba eixos individuais) | 23 |
| Figura 5 | Dashboard do ThingsBoard com a Telimetria | 24 |
| Figura 6 | Listagem de Imagens..... | 25 |
| Figura 7 | Mensagem de instrução | 25 |
| Figura 8 | Registro das imagens no banco de dados | 26 |
| Figura 9 | Registro das imagens no servidor | 26 |
| Figura 10 | Aba de Configurações do sistema | 27 |
| Figura 11 | Aba de Configurações do sistema | 28 |
| Figura 12 | Documentação da aplicação | 28 |

SUMÁRIO

| | | |
|-------|---|----|
| 1 | INTRODUÇÃO | 8 |
| 1.1 | Objetivos | 9 |
| 1.1.1 | Objetivo Geral | 9 |
| 1.1.2 | Objetivos Específicos | 10 |
| 2 | FUNDAMENTAÇÃO TEÓRICA | 11 |
| 2.1 | A Manutenção Preditiva na Era da Indústria 4.0 | 11 |
| 2.2 | Análise de Vibração como Ferramenta de Diagnóstico | 11 |
| 2.3 | Técnicas de Processamento de Sinais para Análise de Vibração..... | 12 |
| 3 | METODOLOGIA | 14 |
| 3.1 | Prospecção de Tecnologia | 14 |
| 3.2 | Etapas Realizadas..... | 15 |
| 4 | RESULTADOS E DISCUSSÃO | 17 |
| 4.1 | Emulador de vibrações..... | 18 |
| 4.2 | Validação com motor monofásico..... | 19 |
| 4.3 | Plotagem dos Gráficos | 22 |
| 4.4 | Integração com a plataforma ThingsBoard..... | 24 |
| 4.5 | Visualização e Gerenciamento de Imagens | 24 |
| 4.6 | Armazenamento das Imagens | 26 |
| 4.7 | Configurações | 27 |
| 4.8 | Documentação | 28 |
| 5 | CONSIDERAÇÕES FINAIS | 29 |
| | REFERÊNCIAS | 30 |
| | ANEXOS..... | 32 |
| | Anexo A - Vídeo de Demonstração do Aplicativo..... | 32 |

1 INTRODUÇÃO

Nas indústrias tradicionais, as práticas de manutenção são frequentemente classificadas como corretivas ou preventivas. Conforme aponta Baldissarelli and Fabro (2018), a manutenção corretiva é aplicada somente após a falha de um equipamento, pode levar a paradas não programadas, altos custos operacionais e prejuízos na produção. Já a manutenção preventiva busca evitar essas falhas por meio de inspeções e substituições programadas, muitas vezes baseadas no tempo de uso ou na experiência técnica. Embora essas abordagens sejam amplamente utilizadas, elas nem sempre conseguem antecipar falhas de maneira precisa, o que limita sua eficácia em ambientes industriais modernos.

Com o advento da Indústria 4.0, surgiram novas oportunidades para superar as limitações das práticas tradicionais de manutenção. A integração de tecnologias como Internet das Coisas (Internet of Things - IoT), Big Data e inteligência artificial possibilitou a implementação de sistemas de manutenção mais inteligentes e conectados. De acordo com Pereira and de Oliveira Simonetto (2018), a Indústria 4.0 visa transformar os processos industriais por meio de digitalização e conectividade, promovendo maior eficiência e tomada de decisões baseadas em dados. Nesse contexto, a manutenção preditiva ganha destaque ao utilizar sensores inteligentes e análise em tempo real para prever falhas antes que ocorram, garantindo maior confiabilidade e disponibilidade dos equipamentos.

A manutenção preditiva é uma abordagem essencial para garantir a confiabilidade e a eficiência dos equipamentos industriais, sendo amplamente aplicada para monitorar e diagnosticar falhas em sistemas mecânicos. De acordo com Baldissarelli and Fabro (2018), "a manutenção preditiva também é conhecida como manutenção sob condição ou manutenção com base no estado do equipamento. Ela é baseada na tentativa de definir o estado futuro de um equipamento ou sistema, por meio dos dados coletados ao longo do tempo por uma instrumentação específica, verificando e analisando a tendência de variáveis do equipamento."

A análise de vibrações se destaca como uma das técnicas cruciais da manutenção preditiva no monitoramento de máquinas rotativas, como os compressores de sistemas de climatização. Isso ocorre porque os compressores emitem padrões de vibrações característicos quando estão em bom estado de funcionamento. Como destaca Cyrino (2017), ao longo do tempo, essas vibrações resultam em desgaste e deterioração, o que modifica

a distribuição da energia vibratória entre os componentes que constituem a máquina ou o equipamento. Consequentemente, o aparelho foge de seus padrões, o que se caracteriza pelo aumento na intensidade de suas vibrações.

Diante do contexto apresentado, este trabalho propõe o desenvolvimento de uma aplicação para análise de padrões de vibração de compressores. O aplicativo é um componente de um sistema de monitoramento de condicionadores de ar concebido no âmbito do GPSERS (Grupo de Pesquisa em Sistemas Embutidos e Redes de Sensores), localizado no Instituto Federal de Pernambuco - Campus Recife. A proposta se motiva pela busca de preencher a lacuna entre a necessidade de soluções acessíveis e os altos custos das ferramentas comerciais, por meio da utilização de sensores de baixo custo, como o acelerômetro MPU-6050 (com custo médio de R\$ 30,00), e técnicas eficazes de análise preditiva. O objetivo principal é criar uma ferramenta prática e acessível, capaz de identificar precocemente falhas, otimizando a manutenção de sistemas de climatização.

A abordagem proposta inclui três perspectivas principais: (i) gráficos no domínio do tempo, para análise inicial e detecção de padrões anômalos; (ii) transformada rápida de Fourier, para identificar as frequências predominantes associadas a possíveis falhas; e (iii) transformada de Fourier de curto prazo, para analisar a evolução dessas frequências ao longo do tempo. Esses métodos serão implementados e validados no aplicativo proposto, com o objetivo de criar uma ferramenta eficaz para diagnósticos e manutenção preditiva.

1.1 Objetivos

Nesta seção, será apresentado o principal objetivo do trabalho, bem como os objetivos específicos a serem alcançados.

1.1.1 Objetivo Geral

Desenvolver um aplicativo para análise de padrões de vibração de compressores, utilizando métodos de processamento de sinais como gráficos no domínio do tempo, transformada rápida de Fourier (FFT) e transformada de Fourier de curto prazo (STFT), a fim de identificar anomalias que possam indicar falhas mecânicas e promover a manutenção preditiva com sensores de baixo custo.

1.1.2 Objetivos Específicos

- Implementar métodos de coleta e processamento de dados de vibração captados por sensores acessíveis, otimizando o custo do monitoramento;
- Desenvolver gráficos no domínio do tempo, domínio da frequência e gráfico de intensidade de cores para identificar padrões e anomalias nos sinais de vibração;
- Criar uma interface intuitiva que permita a visualização e interpretação dos dados por usuários;
- Validar o aplicativo em compressores reais, verificando sua eficácia na identificação de falhas mecânicas de forma prática;
- Estruturar base de dados para futuras avaliações baseadas em IA.

2 FUNDAMENTAÇÃO TEÓRICA

Esta seção apresenta os conceitos fundamentais que sustentam o desenvolvimento do aplicativo, abordando o paradigma da Manutenção Preditiva na Indústria 4.0, a relevância da análise de vibrações como ferramenta de diagnóstico e as técnicas de processamento de sinais empregadas para a extração de informações úteis sobre a condição dos compressores.

2.1 A Manutenção Preditiva na Era da Indústria 4.0

A Quarta Revolução Industrial, ou Indústria 4.0, representa uma mudança de paradigma nos processos de manufatura, impulsionada pela fusão dos mundos físico e digital. Seu alicerce tecnológico é construído sobre Sistemas Ciber-Físicos (CPS), que integram sensores, poder computacional e conectividade em rede para monitorar e controlar processos físicos em tempo real Lee et al. (2015). Nesse contexto, a Internet das Coisas (IoT) emerge como uma tecnologia habilitadora essencial, permitindo que máquinas e equipamentos coletem e transmitam continuamente um volume massivo de dados operacionais.

É a capacidade de analisar esses dados que transforma as estratégias de manutenção. A manutenção preditiva (PdM), um tipo de Manutenção Baseada na Condição (CBM), beneficia-se diretamente desse ecossistema, utilizando dados coletados pelo monitoramento de condições para guiar as decisões. Diferentemente da manutenção preventiva, que opera em intervalos fixos, a PdM permite que ações de manutenção sejam tomadas apenas quando há evidência de comportamento anormal em um ativo físico. Essa abordagem busca evitar intervenções desnecessárias, o que pode reduzir significativamente os custos de manutenção e maximizar a disponibilidade dos equipamentos Jardine et al. (2006).

2.2 Análise de Vibração como Ferramenta de Diagnóstico

Toda máquina rotativa, incluindo compressores, gera vibração como parte natural de sua operação. Essa vibração, quando o equipamento está em condição normal de funcionamento, compõe uma "assinatura" característica estável. No entanto, o surgimento e a progressão de defeitos mecânicos alteram inevitavelmente as forças dinâmicas internas

da máquina, modificando essa assinatura de vibração de maneira detectável. Por essa razão, a análise de vibrações é amplamente reconhecida como uma das técnicas mais eficazes para o monitoramento da saúde de equipamentos rotativos Jardine et al. (2006).

Diferentes modos de falha geram padrões de vibração distintos que podem ser identificados na análise de frequência. Por exemplo, uma distribuição de massa irregular, característica do desbalanceamento, tende a excitar a vibração predominantemente na frequência fundamental de rotação do equipamento. Já o desalinhamento entre componentes costuma gerar um padrão mais complexo, com picos de vibração em múltiplos dessa frequência principal. Defeitos em rolamentos, por sua vez, são frequentemente identificados por sinais de impacto que se manifestam em altas frequências, enquanto folgas mecânicas podem produzir um espectro de vibração mais distribuído, com diversos componentes relacionados à rotação. A capacidade de monitorar e interpretar essas mudanças vibracionais é, portanto, a chave para um diagnóstico precoce e preciso de falhas, sendo um pilar da manutenção preditiva.

2.3 Técnicas de Processamento de Sinais para Análise de Vibração

O sinal de vibração bruto capturado por um acelerômetro é uma série temporal complexa. Para extrair informações diagnósticas, é necessário aplicar técnicas de processamento de sinais, que transformam os dados em representações mais informativas. Este trabalho emprega uma abordagem multiperspectiva, utilizando análises no domínio do tempo, da frequência e tempo-frequência.

- **Análise no Domínio do Tempo:** A representação mais direta do sinal é no domínio do tempo. A partir desta forma de onda, é possível extrair métricas estatísticas que fornecem uma visão geral da condição da máquina, como valor de pico, valor RMS (Root Mean Square) e Fator de Crista. Embora úteis, esses indicadores são geralmente limitados para diagnosticar a causa específica da falha e servem como um primeiro nível de alerta Jardine et al. (2006).
- **Análise no Domínio da Frequência (FFT):** Para identificar as causas subjacentes de uma vibração excessiva, é essencial decompor o sinal em suas frequências constituintes. A ferramenta matemática para essa tarefa é a Transformada de Fourier. Na prática, utiliza-se o algoritmo da Transformada Rápida de Fourier (FFT), que

é computacionalmente eficiente para calcular o espectro de frequência de um sinal digital Smith (1997). O espectro resultante mostra a amplitude da vibração em cada frequência, permitindo associar picos de energia a fenômenos físicos específicos, como as falhas de desbalanceamento e desalinhamento. A análise de frequência é a técnica de diagnóstico mais poderosa na manutenção preditiva.

- **Análise no Domínio Tempo-Frequência (STFT):** Os métodos de análise nos domínios do tempo e da frequência assumem que o sinal de vibração é estacionário, uma suposição que nem sempre é verdadeira, especialmente quando defeitos começam a surgir. Para sinais não-estacionários, as análises de tempo-frequência são mais adequadas. A Transformada de Fourier de Curto Prazo (STFT) é uma das principais técnicas dessa categoria. Ela funciona dividindo o sinal não-estacionário em segmentos menores, que podem ser considerados localmente estacionários, e aplicando a FFT a cada um desses segmentos. O resultado, geralmente visualizado como um espectrograma, revela como o conteúdo de frequência do sinal evolui com o tempo, sendo crucial para capturar eventos transitórios e entender a dinâmica de falhas em desenvolvimento Safizadeh et al. (2005).

A combinação dessas três abordagens de análise oferece uma visão completa do comportamento vibratório do compressor, permitindo um diagnóstico informado sobre a natureza e evolução de possíveis falhas.

3 METODOLOGIA

Neste capítulo, serão apresentadas as principais abordagens metodológicas utilizadas neste trabalho.

3.1 Prospecção de Tecnologia

Inicialmente, o desenvolvimento do aplicativo foi realizado em Java. Contudo, a linguagem foi posteriormente substituída por Python devido à sua ampla biblioteca de recursos voltados à análise. Entretanto, como o Python é naturalmente mais lento que o Java, serão realizadas análises de desempenho à medida que as funcionalidades forem sendo finalizadas. Abaixo estão representados os recursos de desenvolvimento:

- Visual Studio Code (VSCODE, 2025): IDE utilizada para a implementação das funcionalidades e implantação das bibliotecas;
- NumPy (NUMPY, 2025): Biblioteca utilizada para simular sinais de vibração com senoides e calcular a transformada rápida de Fourier (FFT);
- Librosa (LIBROSA, 2025): Biblioteca utilizada para calcular a transformada de Fourier de curto prazo (STFT) e gerar os espectrogramas;
- Matplotlib (MATPLOTLIB, 2025): Biblioteca utilizada para fazer a plotagem dos sinais e apresentá-los;
- PostgreSQL (POSTGRESQL, 2025): Banco de dados relacional utilizado para armazenar os sinais analisados.
- ThingsBoard (THINGSBOARD, 2025): Plataforma open-source de IoT utilizada para gerenciar dispositivos conectados, coletar e processar dados dos sensores, além de disponibilizar dashboards interativos para visualização e análise em tempo real.
- Pdoc (PDOCLIB, 2025): Biblioteca utilizada para gerar a documentação do sistema em formato html.

3.2 Etapas Realizadas

O desenvolvimento do aplicativo progrediu utilizando as metodologias ágeis Scrum SCRUM (2025) e Kanban KANBAN (2025), para entregas contínuas divididas em *sprints*. As *sprints* foram definidas com duração de um mês e, ao final de cada uma, foram realizadas reuniões para discutir e fazer revisões sobre o que foi entregue. Abaixo estão representadas as etapas realizadas:

1. **Estruturação básica do aplicativo:** Foi desenvolvida a estrutura inicial do aplicativo, que incluiu a integração e os testes das bibliotecas para geração de sinais. Foram geradas senóides, aplicadas a Transformada Rápida de Fourier (FFT) e criados gráficos tanto no domínio do tempo quanto no domínio da frequência, além de implementar a plotagem desses gráficos em uma interface para visualização.
2. **Implementação de registro dos dados:** Foi desenvolvido o banco de dados, com a implementação de métodos de acesso, bem como de leitura e escrita dos sinais.
3. **Implementação de gráfico de intensidade com cores:** Foi feita a implementação e plotagem de um gráfico no formato de espectrograma, que representa as frequências presentes no sinal ao longo do tempo.
4. **Implementação de Servidor Socket TCP:** Foi feita a implementação de serviço de recepção e processamento do stream json enviado pelo sensor de vibração.
5. **Implementação de funcionalidades para gerar e inserir as imagens no banco de dados:** Foram implementadas funcionalidades para salvar as imagens geradas pelos sensores no servidor e, conseqüentemente, no banco de dados.
6. **Implementação de funcionalidades para buscar e deletar imagens:** Foram adicionadas funções para buscar imagens salvas no servidor e para deletar imagens que estão locais ou remotas (servidor e banco de dados).
7. **Implementação da interface do usuário:** Foi implementada a interface de usuário com uma estrutura de abas, contendo: (a) visualização de gráficos gerais; (b) visualização dos gráficos de forma individualizada com painéis para exibição de métricas de diagnóstico (ex: Frequência de Pico, Amplitude da Frequência de Pico);

(c) funcionalidade para listar, baixar e deletar as imagens salvas; e (d) uma aba de configurações para todos os parâmetros de conexão.

8. **Implementação de Broker mosquitto MQTT:** Foi feita a implementação de um container docker com broker mosquitto para recepção de dados do sensor via MQTT.
9. **Implementação da Integração com a Plataforma ThingsBoard:** Foi implementada a funcionalidade de telemetria, responsável pelo envio em tempo real das métricas de diagnóstico para a plataforma de IoT ThingsBoard.
10. **Implementação da Documentação:** Foi implementada a documentação da aplicação no formato html utilizando a biblioteca pdoc.

4 RESULTADOS E DISCUSSÃO

A manutenção preditiva tem se destacado como uma solução promissora para evitar falhas e otimizar o desempenho de sistemas industriais. Contudo, apesar de sua relevância, a implementação dessa abordagem ainda enfrenta barreiras significativas. Ferramentas comerciais de análise de vibração, amplamente utilizadas para monitoramento e diagnóstico de equipamentos, frequentemente apresentam custos elevados, tanto em relação aos sensores quanto aos sistemas de processamento e análise de dados. Essa realidade restringe o acesso a essas tecnologias, especialmente em ambientes que buscam soluções mais acessíveis e escaláveis.

Adicionalmente, sensores sofisticados, como acelerômetros de alta precisão, embora forneçam dados detalhados, podem ser superdimensionados para aplicações menos críticas ou específicas, como o monitoramento de compressores em sistemas de climatização. Essa lacuna entre custo e necessidade limita a adoção de soluções preditivas em diversos ambientes industriais e comerciais.

Um exemplo dessa disparidade é apresentado por Vavassori et al. (2022), que desenvolveu um protótipo de análise de vibração com custo de **R\$ 295,10**, comparando-o com sensores comerciais mais sofisticados. Os resultados demonstraram que, embora o protótipo seja capaz de realizar leituras precisas em determinadas faixas de frequência, diferenças nas amplitudes de aceleração foram observadas em outras faixas devido à menor precisão do sensor e à variação na taxa de amostragem. Isso evidencia tanto o potencial quanto os desafios da adoção de soluções de baixo custo.

Neste contexto, a solução aqui apresentada se destaca por levar a acessibilidade a um novo patamar. Enquanto o protótipo citado representa um avanço, esta abordagem emprega o acelerômetro MPU-6050, cujo custo médio é de aproximadamente R\$30,00, quase dez vezes inferior ao do protótipo de referência e uma fração mínima do valor de sensores comerciais. Essa escolha estratégica de hardware é fundamental para a viabilidade da proposta, reforçando o objetivo de democratizar o acesso à manutenção preditiva.

Para validar a proposta, inicialmente, foi criado um emulador de vibrações (Figura 1) com um acelerômetro que envia os dados via protocolo TCP e MQTT para o aplicativo, que processa as informações e exibe gráficos na tela.

4.1 Emulador de vibrações

Figura 1: Emulador de Vibração



Fonte: O Autor, 2025.

O emulador representado é composto pelo sensor acelerômetro MPU-6050 que capta as vibrações geradas por um cooler. O acelerômetro foi escolhido por sua capacidade de detectar movimentos e vibrações de maneira eficiente, além de ser acessível em termos de custo.

O objetivo do emulador é replicar o comportamento vibracional típico de um compressor em funcionamento. O acelerômetro mede a aceleração em três eixos (X, Y e Z), proporcionando uma visão tridimensional das vibrações geradas pelo compressor durante seu funcionamento. Esses dados são transmitidos em tempo hábil para o aplicativo de análise via protocolo TCP/IP ou MQTT.

4.2 Validação com motor monofásico

Com o ciclo de desenvolvimento do aplicativo finalizado e todas as suas funcionalidades, desde a análise de sinais até a interface do usuário e o armazenamento de dados, implementadas e testadas em ambiente simulado, a etapa subsequente foi a validação final do sistema. O objetivo desta fase foi comprovar a eficácia e a robustez da solução completa ao ser submetida a dados de vibração provenientes de um equipamento físico real.

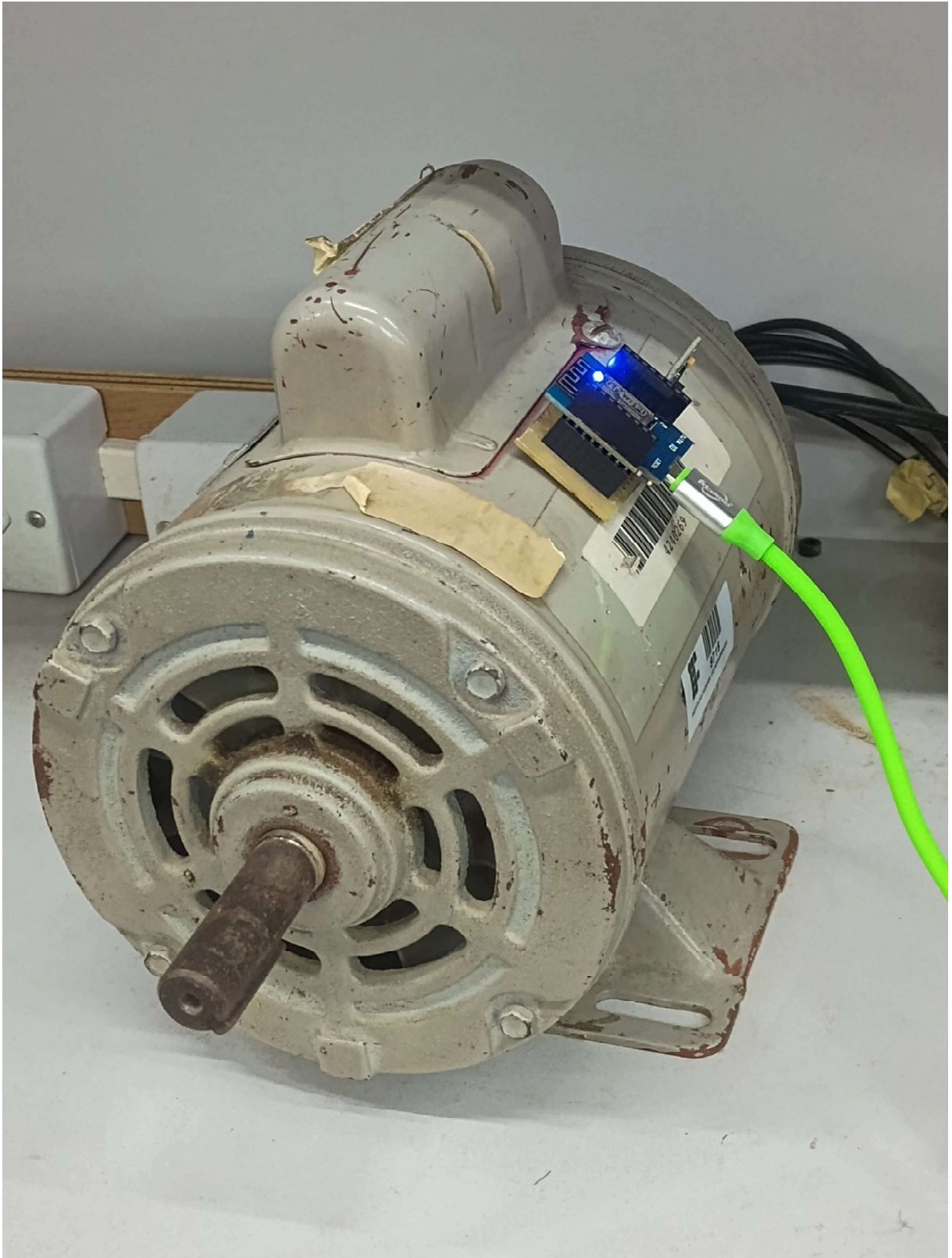
Para este propósito, foi utilizada uma bancada de testes com um motor elétrico industrial, conforme ilustrado na Figura 2. O equipamento é um motor de indução monofásico do tipo TFVE (Totalmente Fechado com Ventilação Externa), escolhido por ser representativo dos compressores e ventiladores encontrados em sistemas de climatização, o alvo principal deste trabalho.

Para a condução do teste, o sensor acelerômetro MPU-6050 foi acoplado diretamente à carcaça do motor. O sucesso deste teste prático confirmou que o sistema, já integralmente desenvolvido, é capaz de operar de forma eficaz com sinais do mundo real, validando a proposta deste trabalho.

A análise espectral dos dados coletados pelo aplicativo, por meio da Transformada Rápida de Fourier (FFT), revelou um pico de frequência dominante e claramente definido. Conforme mostram os gráficos da Figura 3, esse pico foi identificado em aproximadamente 123 Hz. Este resultado é plenamente condizente com a assinatura de vibração esperada para um motor de indução monofásico operando em uma rede elétrica de 60 Hz. O componente em 120 Hz, correspondente a duas vezes a frequência da linha (2x FL), é o

mais característico e resulta da pulsação do torque eletromagnético intrínseco a esse tipo de motor (Finley. et al., 1999). A pequena diferença entre o valor teórico (120 Hz) e o medido (123 Hz) é atribuída à resolução da análise espectral, uma consequência do uso de uma janela de 256 amostras para garantir o processamento em tempo hábil dos dados provenientes do sensor de baixo custo.

Figura 2: Motor Monofásico

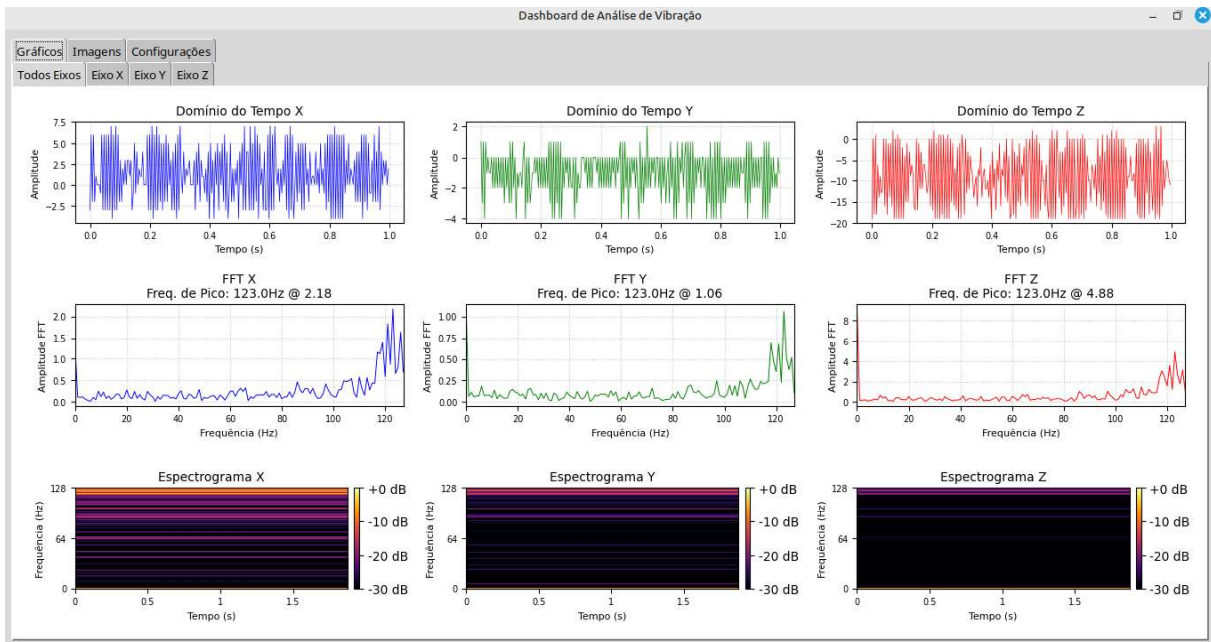


Fonte: O Autor, 2025.

4.3 Plotagem dos Gráficos

A aba principal da interface do aplicativo, conforme exibida na Figura 3 permite a visualização dos gráficos no domínio do tempo, FFT e STFT, facilitando a identificação de possíveis anormalidades no comportamento do equipamento.

Figura 3: Gráficos de Análise (aba Todos os Eixos)



Fonte: O Autor, 2025.

Os gráficos na primeira linha da interface referem-se ao domínio do tempo, representando a amplitude da vibração em função do tempo. Picos de amplitude podem estar relacionados a alterações no comportamento do equipamento monitorado, como variações no funcionamento do compressor.

Os gráficos na segunda linha da interface resultam da aplicação da Transformada Rápida de Fourier (FFT) nos sinais que estão no domínio do tempo. A transformada foi aplicada para identificar as principais frequências (em Hz) associadas ao funcionamento do equipamento.

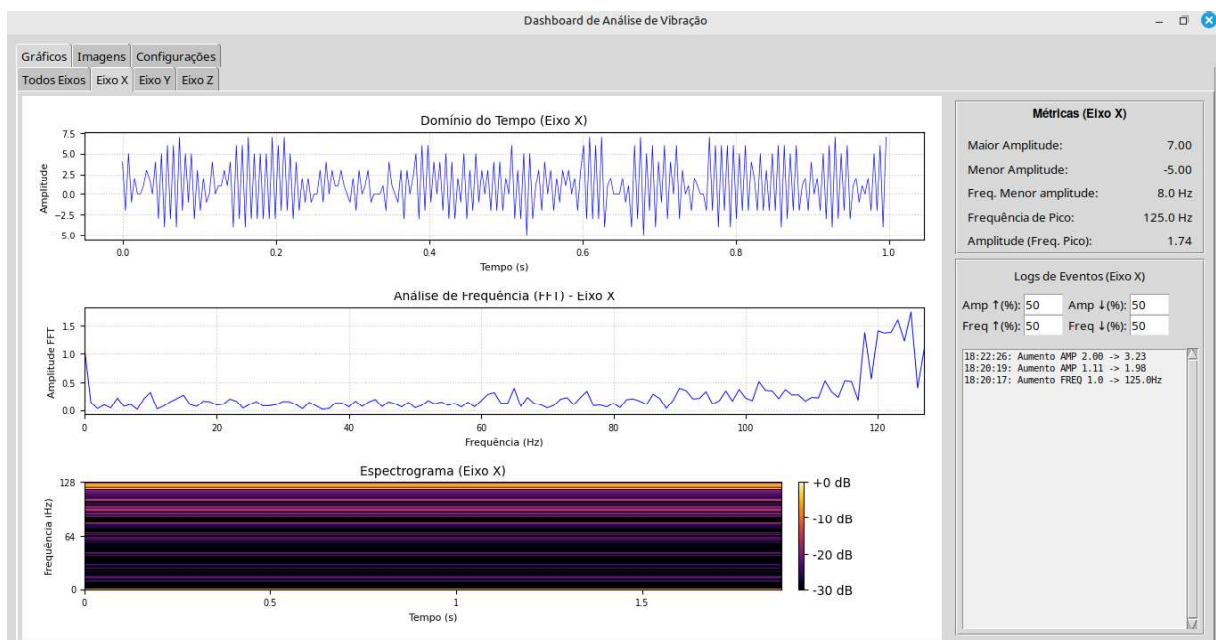
Os gráficos na terceira linha da interface representam a Transformada de Fourier de Curto Prazo (STFT). Esses gráficos possuem uma escala de cores que indica as diferentes frequências ao longo do tempo. Cores mais escuras representam menor intensidade das frequências, enquanto cores mais claras representam frequências de maior intensidade.

Um dos resultados mais significativos da plataforma é a sua capacidade de traduzir

a análise visual para o domínio quantitativo, por meio da extração automática de métricas-chave. Conforme demonstrado na Figura 4, para cada eixo de análise, o sistema calcula e exibe em tempo real um painel com os seguintes indicadores diagnósticos:

1. **Maior e Menor Amplitude:** Indicam os valores extremos da vibração no domínio do tempo, úteis para detectar impactos ou eventos transitórios.
2. **Frequência de Pico:** Identifica a frequência dominante no espectro, sendo a métrica mais crítica para associar a vibração a falhas mecânicas específicas.
3. **Amplitude da Frequência de Pico:** Quantifica a severidade da vibração na frequência dominante, permitindo acompanhar a evolução de um defeito.
4. **Frequência de Menor Amplitude:** Fornece uma referência do nível de vibração de base ou ruído do sistema.
5. **Logs de Aumentos e Quedas:** Registra eventos de aumento ou quedas bruscas de amplitude e frequência de pico.

Figura 4: Métricas na Interface (aba eixos individuais)



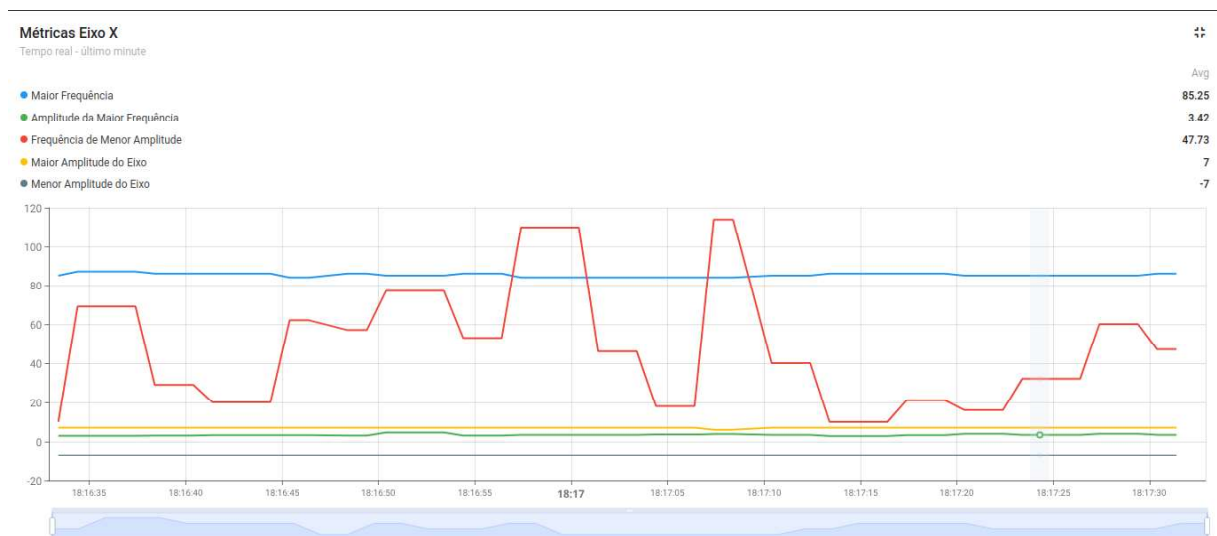
Fonte: O Autor, 2025.

4.4 Integração com a plataforma ThingsBoard

Visando a escalabilidade e a conformidade com os princípios da Indústria 4.0, a plataforma foi integrada a um sistema externo de Internet das Coisas (IoT). As métricas quantitativas mais críticas, extraídas durante a análise, são enviadas em tempo real para um dashboard na nuvem.

Essa integração expande as fronteiras do monitoramento para além da estação de análise local. Ela permite que gestores e equipes de manutenção supervisionem a condição do equipamento remotamente, configurem alertas automáticos por e-mail ou SMS caso os níveis de vibração excedam limiares de segurança e centralizem dados de múltiplos ativos em um único local. A Figura 5 apresenta o recebimento e a plotagem desses dados telemétricos na plataforma ThingsBoard, validando o fluxo de comunicação de ponta a ponta.

Figura 5: Dashboard do ThingsBoard com a Telimetria



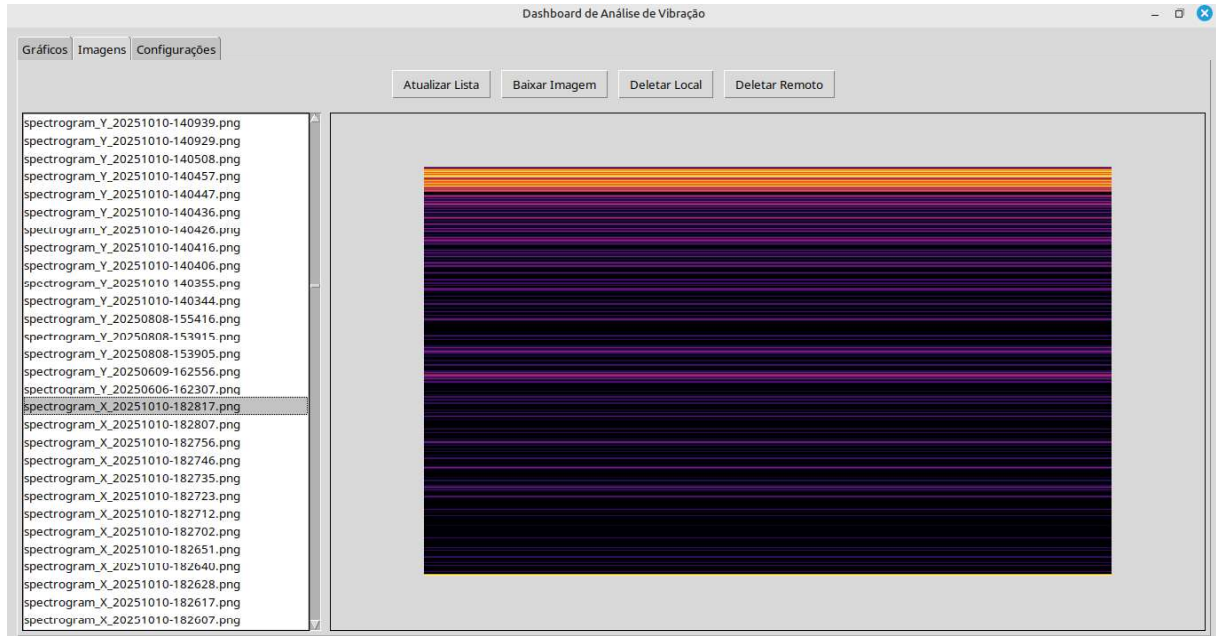
Fonte: O Autor, 2025.

4.5 Visualização e Gerenciamento de Imagens

O aplicativo também oferece uma funcionalidade para exibir as imagens armazenadas no servidor. A Figura 6 ilustra a interface do aplicativo na aba imagens, que permite ao usuário visualizar e gerenciar as imagens salvas. Todas as imagens salvas no servidor são listadas e o usuário pode baixá-las para um diretório local para análise posterior ou deletá-las quando necessário. Essa funcionalidade é crucial para facilitar o gerenciamento

das informações coletadas, garantindo que o usuário tenha acesso rápido e organizado às imagens associadas ao monitoramento dos sinais de vibração.

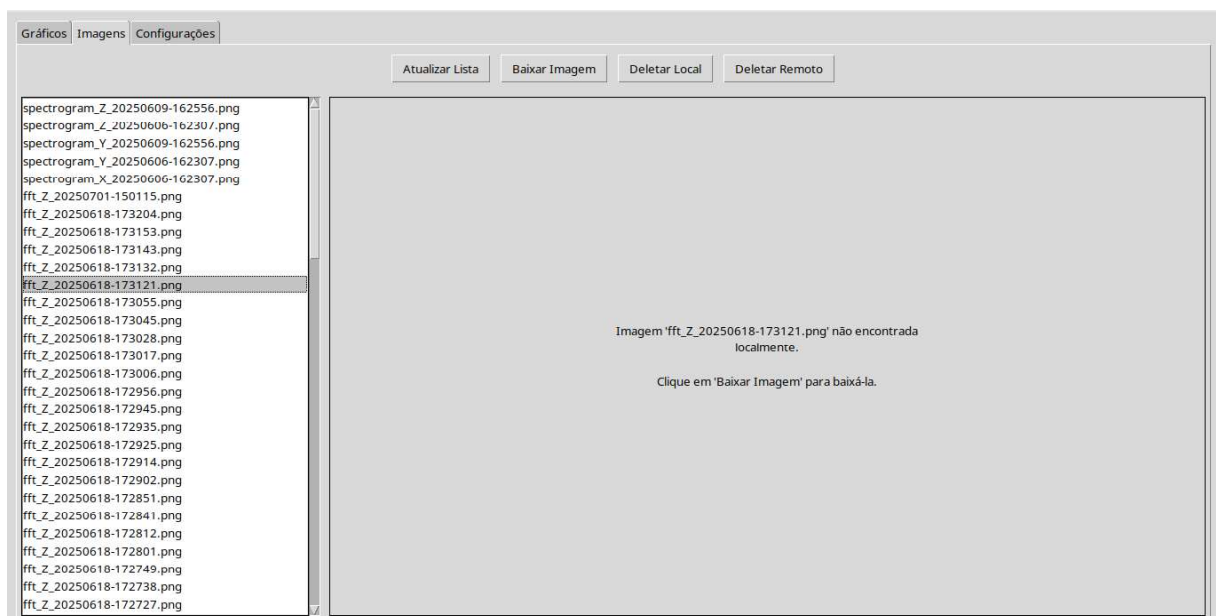
Figura 6: Listagem de Imagens



Fonte: O Autor, 2025.

Além disso, caso a imagem não esteja baixada localmente para visualização é exibido uma mensagem indicativa como mostra a figura abaixo (Figura 7):

Figura 7: Mensagem de instrução



Fonte: O Autor, 2025.

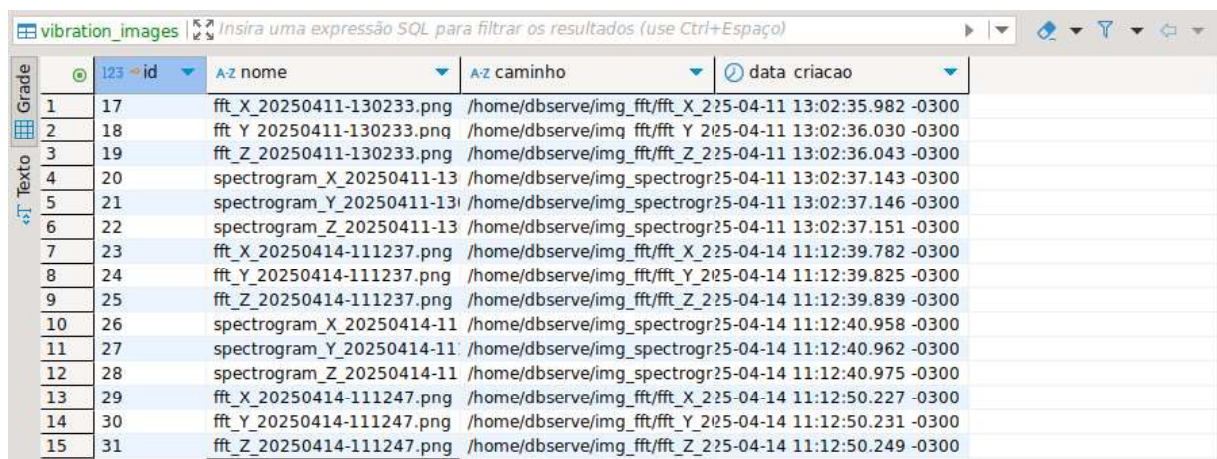
4.6 Armazenamento das Imagens

As imagens listadas estão sendo armazenadas em pastas em um servidor, organizadas em diretórios específicos conforme o tipo de análise (img_fft e img_espectrograma). O banco de dados PostgreSQL não armazena os arquivos de imagem diretamente, mas atua como um índice de metadados, registrando em uma tabela o caminho (path) exato para cada imagem no servidor.

Um componente fundamental deste formato é a implementação de uma convenção de nomenclatura padronizada e autodescritiva para cada arquivo gerado, seguindo o formato: tipoDaAnálise_eixo_data-hora.png (ex: fft_X_20250411-130233.png). Este padrão garante não apenas a unicidade de cada registro, evitando sobreposições, mas também oferece rastreabilidade completa, permitindo que cada imagem seja facilmente associada a um tipo de análise, a um eixo específico e ao momento exato em que foi gerada, apenas pela leitura de seu nome.

A abordagem é ilustrada na Figura 8 e Figura 9, que apresenta os registros no banco de dados e os arquivos no servidor.

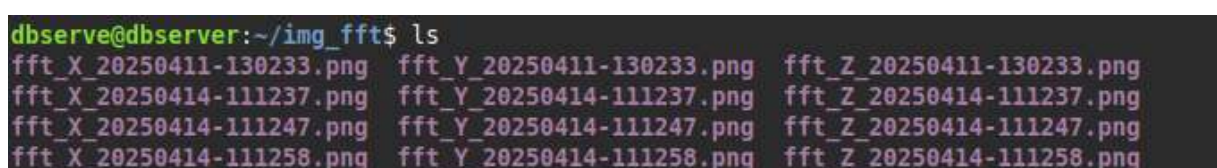
Figura 8: Registro das imagens no banco de dados



| | id | nome | caminho | data_criacao |
|----|----|---------------------------|--|--------------|
| 1 | 17 | fft_X_20250411-130233.png | /home/dbserve/img_fft/fft_X_225-04-11 13:02:35.982 -0300 | |
| 2 | 18 | fft_Y_20250411-130233.png | /home/dbserve/img_fft/fft_Y_225-04-11 13:02:36.030 -0300 | |
| 3 | 19 | fft_Z_20250411-130233.png | /home/dbserve/img_fft/fft_Z_225-04-11 13:02:36.043 -0300 | |
| 4 | 20 | spectrogram_X_20250411-13 | /home/dbserve/img_spectrogr25-04-11 13:02:37.143 -0300 | |
| 5 | 21 | spectrogram_Y_20250411-13 | /home/dbserve/img_spectrogr25-04-11 13:02:37.146 -0300 | |
| 6 | 22 | spectrogram_Z_20250411-13 | /home/dbserve/img_spectrogr25-04-11 13:02:37.151 -0300 | |
| 7 | 23 | fft_X_20250414-111237.png | /home/dbserve/img_fft/fft_X_225-04-14 11:12:39.782 -0300 | |
| 8 | 24 | fft_Y_20250414-111237.png | /home/dbserve/img_fft/fft_Y_225-04-14 11:12:39.825 -0300 | |
| 9 | 25 | fft_Z_20250414-111237.png | /home/dbserve/img_fft/fft_Z_225-04-14 11:12:39.839 -0300 | |
| 10 | 26 | spectrogram_X_20250414-11 | /home/dbserve/img_spectrogr25-04-14 11:12:40.958 -0300 | |
| 11 | 27 | spectrogram_Y_20250414-11 | /home/dbserve/img_spectrogr25-04-14 11:12:40.962 -0300 | |
| 12 | 28 | spectrogram_Z_20250414-11 | /home/dbserve/img_spectrogr25-04-14 11:12:40.975 -0300 | |
| 13 | 29 | fft_X_20250414-111247.png | /home/dbserve/img_fft/fft_X_225-04-14 11:12:50.227 -0300 | |
| 14 | 30 | fft_Y_20250414-111247.png | /home/dbserve/img_fft/fft_Y_225-04-14 11:12:50.231 -0300 | |
| 15 | 31 | fft_Z_20250414-111247.png | /home/dbserve/img_fft/fft_Z_225-04-14 11:12:50.249 -0300 | |

Fonte: O Autor, 2025.

Figura 9: Registro das imagens no servidor



```
dbserve@dbserver:~/img_fft$ ls
fft_X_20250411-130233.png  fft_Y_20250411-130233.png  fft_Z_20250411-130233.png
fft_X_20250414-111237.png  fft_Y_20250414-111237.png  fft_Z_20250414-111237.png
fft_X_20250414-111247.png  fft_Y_20250414-111247.png  fft_Z_20250414-111247.png
fft_X_20250414-111258.png  fft_Y_20250414-111258.png  fft_Z_20250414-111258.png
```

Fonte: O Autor, 2025.

4.7 Configurações

Para garantir a máxima flexibilidade, adaptabilidade e a fácil implantação da plataforma em diferentes cenários de hardware e rede, foi implementado um painel de configuração com campos para especificar os atributos de conexão. Este módulo desacopla os parâmetros operacionais do código-fonte, permitindo que o usuário final modifique todas as variáveis críticas da aplicação sem a necessidade de intervenção técnica no software.

Conforme apresentado nas Figura 10 e Figura 11, a interface agrupa de forma lógica todas as configurações essenciais, incluindo os detalhes de conexão com o sensor (via TCP ou MQTT), as credenciais de acesso ao servidor remoto (SSH) e ao banco de dados, os endereços da plataforma de IoT (ThingsBoard) e os caminhos para o armazenamento local e remoto dos arquivos de imagem. A capacidade de salvar e carregar essas configurações a partir de um arquivo externo assegura a consistência e a reprodutibilidade das análises, além de facilitar a migração do sistema entre diferentes ambientes de teste ou produção. Esta abordagem modular reforça a robustez e a usabilidade da ferramenta, tornando-a prontamente adaptável a novas infraestruturas.

Figura 10: Aba de Configurações do sistema

The screenshot shows the 'Configurações' (Settings) tab of a software interface. The tab is selected, and the settings are organized into four sections:

- Conexão do Sensor (TCP):** Host do Sensor: 10.5.0.76, Porta do Sensor: 9090.
- Conexão do Broker (MQTT):** Host: 10.5.0.17, Porta: 1883, User: dexter, Senha MQTT: ***** (masked), Topic: esp8266/data.
- Tipo de conexão:** A checkbox labeled 'Usar MQTT (desmarcado = TCP)' is checked.
- Conexão do Servidor (SSH):** Host SSH: 10.5.0.68, Usuário SSH: dbserve, Senha SSH: ***** (masked).

Fonte: O Autor, 2025.

Figura 11: Aba de Configurações do sistema

The screenshot shows a configuration window with three tabs: 'Gráficos', 'Imagens', and 'Configurações'. The 'Configurações' tab is active. It contains three sections:

- Conexão do Banco de Dados:** Fields for Host do DB (10.5.0.68), Porta do DB (5434), Nome do DB (airpower), Usuário do DB (postgres), and Senha do DB (masked with ****).
- Caminhos de Pastas:** Fields for Pasta Local de Imagens (./img), Pasta Remota (FFT) (/home/dbserve/img_fft/), and Pasta Remota (Espectrograma) (/home/dbserve/img_spectrograms/).
- Conexão com o ThingsBoard:** Fields for Host (10.5.0.66), Porta (8080), and Token (FM4KnhtAAZjvSY0t8n9P).

At the bottom, there are two buttons: 'Carregar do Arquivo' and 'Salvar Alterações'.

Fonte: O Autor, 2025.

4.8 Documentação

Por fim, foi realizada a documentação da aplicação, onde todo o código foi comentado e a documentação foi gerada em formato html, utilizando a biblioteca pdoc. A Figura 12 ilustra a página gerada; nela, é possível percorrer por todos os scripts da aplicação, visualizando a explicação de cada linha de código implementado.

Figura 12: Documentação da aplicação

The screenshot shows a web browser displaying the API documentation for 'util.db'. The page has a sidebar on the left with a search bar and a list of API elements. The main content area shows the documentation for the 'DBManager' class.

util.db

Módulo para gerenciamento de conexão e operações com banco de dados PostgreSQL.

Este módulo implementa a classe DBManager que fornece uma interface simplificada para:

- Conexão com banco de dados PostgreSQL
- Inserção de registros de imagens
- Gerenciamento do ciclo de vida da conexão

class DBManager:

Classe para gerenciamento de operações com banco de dados PostgreSQL.

Responsável por estabelecer, manter e encerrar conexões com o banco de dados, além de prover métodos para operações básicas de inserção de dados.

Atributos:

```
dbname(str)
user(str)
password(str)
host(str)
port(str)
conn(usrconn? connection)
```

Fonte: O Autor, 2025.

5 CONSIDERAÇÕES FINAIS

Este trabalho se propôs a enfrentar o desafio de desenvolver uma ferramenta acessível e eficaz para a manutenção preditiva de compressores. Partindo do contexto da Indústria 4.0, o projeto foi motivado pela necessidade de democratizar o acesso a tecnologias de monitoramento, superando as barreiras de custo e complexidade frequentemente associadas às soluções comerciais. O objetivo central foi criar e validar uma aplicação de software capaz de analisar padrões de vibração utilizando sensores de baixo custo, fornecendo um diagnóstico preciso sobre a condição operacional do equipamento.

A aplicação foi desenvolvida para desktop e ela gerencia todo o ciclo de vida dos dados: desde a aquisição em tempo hábil por meio de protocolos de rede como TCP e MQTT, passando pelo processamento e análise com técnicas consagradas como a Transformada Rápida de Fourier (FFT) e a Transformada de Fourier de Curto Prazo (STFT), até a visualização em um dashboard interativo. O sistema é capaz de extrair métricas quantitativas de forma automática, persistir os resultados em um banco de dados para análise histórica e integrar-se com plataformas de Internet das Coisas (IoT) para monitoramento remoto.

Apesar dos resultados promissores, reconhece-se que este desenvolvimento representa uma etapa inicial. Como trabalhos futuros, sugere-se a validação da ferramenta em campo, com a instalação de sensores em condicionadores de ar em operação real para coletar dados sob diversas condições de carga e ambiente. Adicionalmente, o vasto conjunto de dados que pode ser coletado abre uma importante avenida para a implementação de modelos de inteligência artificial e aprendizado de máquina, que poderiam automatizar a classificação de diferentes tipos de falhas com base nos padrões de vibração. A evolução da plataforma com essas novas funcionalidades tem o potencial de transformá-la em uma ferramenta de diagnóstico ainda mais poderosa e autônoma, consolidando sua relevância no cenário da manutenção 4.0.

REFERÊNCIAS

- Luciano Baldissarelli and Elton Fabro. Manutenção preditiva na indústria 4.0. *Scientia Cum Industria*, 4(1):1–11, 2018.
- Luis Cyrino. Diagnóstico de falhas das vibrações, 2017. URL <https://www.manutencaoemfoco.com.br/diagnostico-de-falhas-das-vibracoes/>.
- W.R Finley., M.M. Hodowanec, and W.G. Holter. An analytical approach to solving motor vibration problems. In *Industry Applications Society 46th Annual Petroleum and Chemical Technical Conference (Cat.No. 99CH37000)*, pages 217–232, 1999.
- Andrew K. S. Jardine, D. Lin, and D. Banjevic. A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical Systems and Signal Processing*, 20(7):1483–1510, 2006.
- KANBAN. *Kanban Guide*, 2025. URL <https://kanban.university/kanban-guide/>.
- Jay Lee, Behrad Bagheri, and Hung an Kao. A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems. *Manufacturing Letters*, 3:18–23, 2015.
- LIBROSA. *Librosa Documentation*, 2025. URL <https://librosa.org/>.
- MATPLOTLIB. *Matplotlib Documentation*, 2025. URL <https://matplotlib.org/>.
- NUMPY. *Numpy Documentation*, 2025. URL <https://numpy.org/>.
- PDOC. *Pdoc Documentation*, 2025. URL <https://pdoc.dev/docs/pdoc.html>.
- Adriano Pereira and Eugênio de Oliveira Simonetto. Indústria 4.0: conceitos e perspectivas para o brasil. *Revista da Universidade Vale do Rio Verde*, 16(1), 2018.
- POSTGRESQL. *PostgreSQL Documentation*, 2025. URL <https://www.postgresql.org/>.
- M. S. Safizadeh, A. A. Lakis, and M. Thomas. Time-frequency analysis of a cracked beam in bending vibration using the Wigner-Ville distribution. *Journal of Vibration and Control*, 11(7):891–911, 2005.

SCRUM. *Scrum Guide*, 2025. URL <https://scrumguides.org/>.

Steven W. Smith. *The Scientist and Engineer's Guide to Digital Signal Processing*. California Technical Publishing, San Diego, CA, 1997.

THINGSBOARD. *ThingsBoard Docs*, 2025. URL <https://thingsboard.io/docs/>.

Matheus Mattei Vavassori, Elvys Isaias Mercado Curi, Richard de Medeiros Castro, Douglas de Medeiros Deolindo, and Guilherme Bampi Righeto. Análise de falhas em equipamentos utilizando sensores de baixo custo, como método de manutenção preditiva. *Periódico Científico da UniSATC*, 7(2):136–194, 2022.

VSCODE. *Virtual Studio Code*, 2025. URL <https://code.visualstudio.com>.

ANEXOS

Anexo A - Vídeo de Demonstração do Aplicativo

Um vídeo de demonstração do aplicativo, desenvolvido neste trabalho, está disponível na plataforma YouTube.

O vídeo pode ser acessado através do seguinte link:

<https://www.youtube.com/watch?v=eHptCSAEI58>