

15º Congresso de Inovação, Ciência e Tecnologia do IFSP - 2024

DESENVOLVIMENTO DE UM PIMS PARA MONITORAMENTO EM TEMPO REAL DOS INDICADORES DE PRODUÇÃO COM ESP32 E PYTHON

GABRIEL VICTORIO DITOMASO¹, JOSÉ ROBERTO DE LIMA², ALISON JOSÉ PEREIRA NERI³, ALLAN MICHAEL CORREA AMARAL⁴, JAIR DE MARTIIN JUNIOR⁵

¹ Graduando em Eng. de Controle e Automação, IFSP, Campus Catanduva, ditomaso.gabriel@aluno.ifsp.edu.br

² Graduando em Eng. de Controle e Automação, IFSP, Campus Catanduva, lima.jose@aluno.ifsp.edu.br

³ Graduando em Eng. de Controle e Automação, IFSP, Campus Catanduva, alison.n@aluno.ifsp.edu.br

⁴ Graduando em Eng. de Controle e Automação, IFSP, Campus Catanduva, a.michael@aluno.ifsp.edu.br

⁵ Professor Doutor de Eletrônica, IFSP, Campus Catanduva, jairdemartinjr@ifsp.edu.br

Área de conhecimento (Tabela CNPq): 3.08.01.02-8: Planejamento, Projeto e Controle de Sistemas de Produção.

RESUMO: Ao permitir a coleta, armazenamento e análise de dados de produção em tempo real, o PIMS é uma ferramenta vital para a otimização de processos industriais na atual Indústria 4.0. Propõe-se o desenvolvimento de um sistema PIMS para digitalizar e monitorar dados do chão de fábrica em tempo real. Foi elaborado um sistema PIMS, onde os dados coletados são armazenados em um arquivo CSV, por meio de um microcontrolador ESP32 e o protocolo de comunicação MQTT sendo assim analisados por um *script* em *Python*. A interface foi desenvolvida com o *Streamlit* que possibilita uma apresentação visual. Os resultados obtidos apresentam uma visualização intuitiva dos indicadores de produção o que torna as decisões estratégicas na indústria pautadas em análise de dados obtidos em tempo real. Conclui-se que o sistema PIMS desenvolvido concatena os dados do chão de fábrica com análises e o exibe de maneira clara aos gestores, com potencial futuro para integração com MES e ERP.

PALAVRAS-CHAVE: *python*, MQTT, indústria 4.0, ESP32, análise de dados.

DEVELOPMENT OF A PIMS FOR REAL-TIME MONITORING OF PRODUCTION INDICATORS WITH ESP32 AND PYTHON

ABSTRACT: By enabling the collection, storage, and analysis of real-time production data, the PIMS is a vital tool for optimizing industrial processes in today's Industry 4.0. The development of a PIMS system is proposed to digitize and monitor shop floor data in real-time. A PIMS system was developed, where the collected data is stored in a CSV file, using an ESP32 microcontroller and the MQTT communication protocol, and then analyzed by a Python script. The interface was developed with Streamlit, which enables a visual presentation. The results provide an intuitive visualization of production indicators, which makes strategic decisions in the industry based on real-time data analysis. It is concluded that the developed PIMS system connects shop floor data with analysis and displays it clearly to managers, with future potential for integration with MES and ERP.

KEYWORDS: *python*, MQTT, industry 4.0, ESP32, data analysis.

INTRODUÇÃO

Na Indústria 4.0, a digitalização no ambiente industrial é essencial para otimizar processos produtivos e reduzir custos operacionais. Com a crescente competitividade, o monitoramento de indicadores-chave como disponibilidade, performance e qualidade de máquinas tornou-se fundamental para garantir a eficiência e continuidade das operações. Nesse cenário, os sistemas PIMS (*Plant Information Management System*) ou sistemas de gestão de informações da planta industrial emergem como ferramentas cruciais para a automação, possibilitando a coleta, análise e visualização de dados em tempo real ou de dados armazenados (Pamplona, 2024; Carvalho *et al.*, 2005).

O avanço das tecnologias de comunicação e a acessibilidade a dispositivos de baixo custo, como o microcontrolador ESP32, com sua capacidade de conectividade sem fio e suporte a diversos sensores, representa uma solução poderosa na coleta de dados em ambientes industriais (Aghenta; Iqbal, 2019).

Paralelamente, a linguagem de programação *Python* tem se consolidado no desenvolvimento de sistemas de análise de dados e visualização. Com suas bibliotecas robustas e versatilidade, o *Python* permite a criação de *dashboards* interativos que facilitam a interpretação dos dados coletados, o que proporciona suporte para decisões e agilidade. (Neri; Pereira, 2023).

Propõe-se o desenvolvimento de um sistema PIMS voltado para a digitalização dos dados do chão de fábrica. Através de sensores, os dados são coletados e enviados para o ESP32, que os armazena em um banco de dados, acessível via web. O *Python* é a linguagem utilizada para gerenciar essa comunicação entre os elementos e para a análise dos dados. O sistema proposto visa melhorar a acessibilidade às informações críticas de produção, permitindo um monitoramento mais eficiente e contribuindo para a melhoria contínua dos processos produtivos.

FUNDAMENTAÇÃO TEÓRICA

A pirâmide de automação é uma representação hierárquica dos níveis de automação industrial, além disso ela organiza os diferentes níveis de sistemas e tecnologias utilizados para automatizar e gerenciar processos industriais que vai desde os dispositivos de campo até os sistemas de gestão corporativa (Bezerra *et al.*, 2020; Joaquim, 2006). A pirâmide é composta principalmente por cinco níveis, como é possível ver na Figura 1, a seguir.

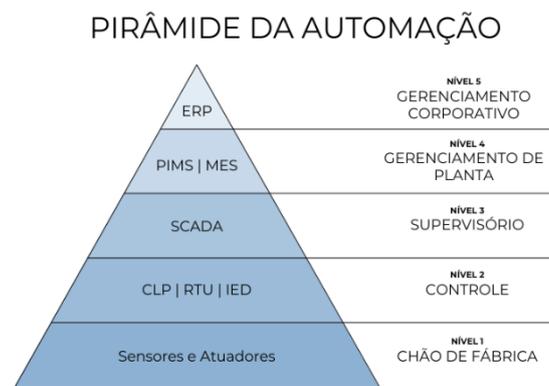


FIGURA 1. Pirâmide da automação industrial.

No nível 1 da pirâmide da automação, conhecido como chão de fábrica, estão os sensores e atuadores. Esses dispositivos são responsáveis pela coleta de dados (pulsos, tempo, contagem etc.) e execução de comandos em tempo real. Para este projeto, foi selecionado um sensor óptico devido à sua alta precisão e capacidade de operar em ambientes industriais. Esse sensor é essencial para capturar dados de produção, o que garante a coleta seja realizada com precisão e rapidez.

No nível 2, encontram-se os controladores, como CLPs (Controladores Lógicos Programáveis), RTUs (Unidades Remotas de Telemetria) e IEDs (Dispositivos Eletrônicos Inteligentes). Esses controladores processam os dados coletados pelos sensores e coordenam os atuadores, permitindo o funcionamento das máquinas e processos conforme as lógicas predefinidas.

Localizado no nível 3, está o sistema SCADA (*Supervisory Control and Data Acquisition*), que monitora e controla os processos em tempo real, que possibilita intervenções e ajustes para assegurar que a operação ocorra dentro dos parâmetros estabelecidos.

O nível 4 da pirâmide é composto pelos sistemas de gestão da planta, como o PIMS (*Plant Information Management System*) e o MES (*Manufacturing Execution System*) que integram as operações do chão de fábrica com os sistemas de gestão empresarial. Esse é o nível de operação do sistema criado, em que os dados coletados são enviados para uma base de dados central, para garantir flexibilidade e acesso rápido às informações da produção.

No topo da pirâmide, o nível 5 é representado pelo ERP (*Enterprise Resource Planning*), que utiliza as informações geradas pelos níveis inferiores para otimizar processos empresariais e gerenciar

recursos de forma integrada, oferecendo uma visão abrangente e em tempo real das operações da empresa.

MATERIAL E MÉTODOS

O sistema desenvolvido neste trabalho e que pode ser descrito dentro do contexto de um projeto PIMS, ou seja, no nível 4 da pirâmide de automação, auxiliando o MES para calcular e apresentar os indicadores-chave e seus históricos. No primeiro estágio, tem-se a aquisição de dados, onde sensores desempenham um papel crucial na captura de informações de campo para realizar os cálculos dos indicadores-chave. Especificamente, um sensor óptico é utilizado para coletar dados que são fundamentais para o processo. Esses dados são então encaminhados para o próximo componente, o ESP32, que atua como um controlador, processando as leituras coletadas para calcular informações críticas como disponibilidade (Equação 1), que mede o tempo em que o equipamento está realmente disponível para produção; performance (Equação 2), que avalia a eficiência da máquina em relação à sua capacidade máxima de produção e qualidade (Equação 3), que determina a proporção de produtos livres de defeitos. O OEE (*Overall Equipment Effectiveness*) tratando-se de um indicador de desempenho geral para as máquinas de produção (Equação 4). Além disso, o ESP32 também interage com botões para registro de paradas de máquina e LEDs para fornecer um *feedback* visual, auxiliando o operador na sinalização e apontamento dessas paradas.

$$Disponibilidade = \frac{Tempo\ de\ máquina\ produzindo}{Tempo\ programado\ para\ máquina\ produzir} \cdot 100 \quad (1)$$

$$Performance = \frac{Quantidade\ de\ produção\ realizada}{Quantidade\ de\ produção\ meta} \cdot 100 \quad (2)$$

$$Qualidade = \frac{Quantidade\ de\ itens\ bons\ produzidos}{Quantidade\ total\ que\ deveria\ ser\ produzida} \cdot 100 \quad (3)$$

$$OEE = Disponibilidade \cdot Performance \cdot Qualidade \quad (4)$$

No segundo estágio, o foco está no armazenamento e processamento dos dados. Após serem processados pelo ESP32 e transmitidos via MQTT (*Message Queuing Telemetry Transport*) para uma base de dados central, organizada em um arquivo CSV (*Comma-Separated Values*). Aqui o Python é responsável por realizar análises adicionais sobre os dados coletados, extraindo indicadores e mostrando as paradas de produção e seus respectivos tempos. Esses dados processados são então apresentados no terceiro estágio, que consiste em uma interface de usuário intuitiva, permitindo que os usuários visualizem informações críticas de maneira clara e rápida. O diagrama ilustrado pela Figura 2 a seguir, trata-se da representação do desenvolvimento de sistema PIMS.

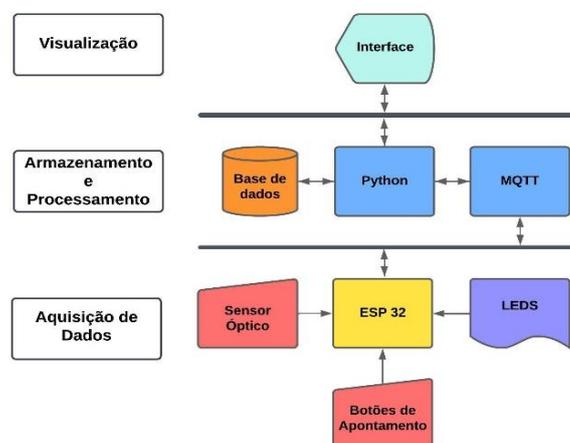


FIGURA 2. Diagrama de desenvolvimento do sistema PIMS.

Na Figura 3, é possível observar o circuito físico do sistema PIMS, composto por componentes essenciais para a coleta e processamento de dados. O sensor óptico, em conjunto com um optoacoplador, captura os dados iniciais e os transmite para o ESP32, que processa essas informações. O sistema também gerencia a interação com botões dedicados ao registro de diferentes tipos de paradas, como manutenção corretiva, preventiva, falta de matéria-prima, entre outras. Além disso, LEDs são utilizados para fornecer *feedback* imediato de cada registro de parada realizado pela operação, o que possibilita o monitoramento em tempo real de todas as intervenções e seus respectivos tempos.

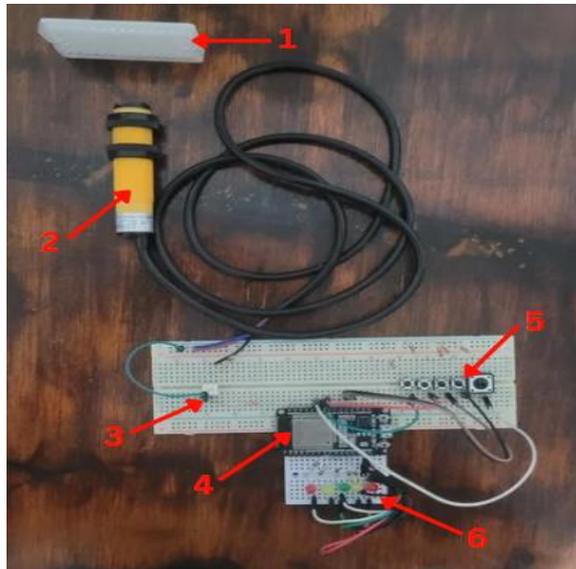


FIGURA 3. Circuito físico do PIMS.

Os seguintes componentes e suas numerações foram utilizados no circuito PIMS:

- | | | |
|---------------------------|------------------------|--|
| 1. Espelho reflexivo; | 3. Optoacoplador 4N25; | 5. Botões do tipo <i>push-button</i> ; |
| 2. Sensor óptico E3F-RN1; | 4. ESP32 Dev kit; | 6. LEDs de <i>feedback</i> . |

O sistema PIMS desenvolvido em *Python* é organizado em cinco componentes principais: comunicação, tratamento de dados, geração de tabelas e armazenamento, criação de interface de usuário e geração de gráficos.

A comunicação é realizada pela biblioteca *Paho*, especializada no gerenciamento do protocolo MQTT. Essa biblioteca cumpre a função de *publisher*, enviando dados por meio de canais (tópicos) para o *broker* (servidor intermediário entre *publishers* e *subscribers*), sem necessidade de conhecer quem receberá as informações. Ela também atua como *subscriber*, inscrevendo-se em tópicos específicos para receber as mensagens enviadas por outros dispositivos conectados ao sistema (apenas recebe as informações dos tópicos que se inscreveu).

O tratamento de dados é efetuado com o auxílio das bibliotecas *Numpy* e *Pandas*. A *Numpy* facilita a manipulação de vetores e a simulação de dados complexos, enquanto a *Pandas* é amplamente utilizada para a análise de dados, oferecendo ferramentas robustas para lidar com dados faltantes, duplicados e para a manipulação de arquivos CSV e outros formatos de maneira eficiente.

A interface de usuário foi criada utilizando a biblioteca *Streamlit*, que possibilita o desenvolvimento rápido e intuitivo de interfaces web interativas. A geração de gráficos e visualizações dos dados é feita através das bibliotecas *Matplotlib* e *Plotly*, que oferecem visualizações interativas e detalhadas de indicadores-chave de produção, como disponibilidade, performance, qualidade e OEE.

Essas visualizações facilitam a interpretação dos dados, o que inclui as ocorrências de paradas de máquina e tempos perdidos, fornecendo suporte essencial para a tomada de decisões estratégicas com base nas informações geradas.

RESULTADOS E DISCUSSÃO

O desenvolvimento do sistema PIMS apresentou uma interface intuitiva, a qual facilita o acesso e o monitoramento dos principais indicadores de produção. A interface construída proporciona uma

navegação simples e eficiente, que permite os usuários acessarem tanto métricas em tempo real quanto dados históricos de maneira ágil e clara. O *layout* foi projetado com foco na simplicidade, que garante as informações mais relevantes estejam sempre visíveis para tomada rápida de decisões.

Um dos principais diferenciais do sistema é a facilidade de programar metas de produção por máquina, como data de produção, tempo programado e a meta de produção (peças desejadas). A interface permite que o usuário defina esses parâmetros de forma prática, integrando automaticamente esses dados ao cálculo dos indicadores-chave de desempenho. A visualização desses parâmetros, juntamente com o *layout* e o exemplo de uma máquina para seleção, pode ser observada na Figura 4.

FIGURA 4. Página de configuração de produção e seleção de máquina.

O sistema PIMS gera *dashboards* com gráficos interativos e dinâmicos que facilitam a interpretação dos principais indicadores de produção geral ou de máquinas específicas, como disponibilidade, performance, qualidade e OEE, conforme ilustrado na Figura 5. Esses gráficos fornecem uma visualização clara dos períodos selecionados, que permite os usuários identificarem rapidamente padrões de desempenho e áreas críticas que necessitam de melhorias no processo produtivo. Além dos indicadores de desempenho, a figura 5 também apresenta a análise detalhada das ordens de parada de máquinas, classificando-as por tipo e registrando o tempo de inatividade.

Essa funcionalidade permite uma compreensão aprofundada das causas e durações das paradas, fornecendo *insights* valiosos para o planejamento de manutenções corretivas e preventivas.

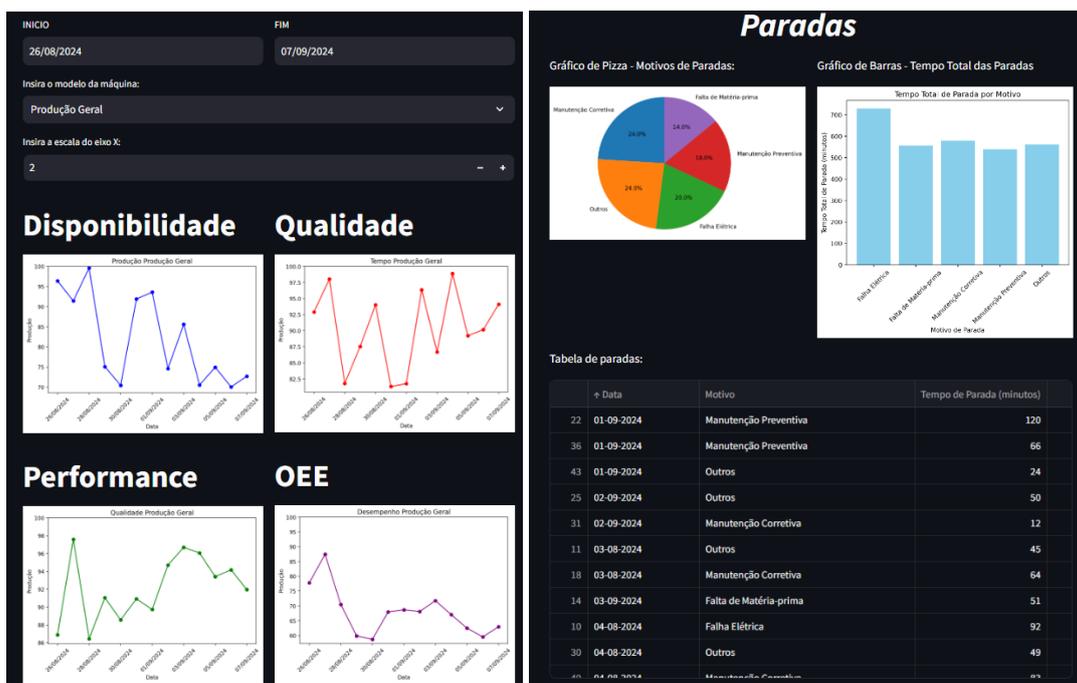


FIGURA 5. Dashboards de análise dos indicadores e paradas de máquinas.

A capacidade de monitorar e analisar de forma precisa os indicadores-chave, como disponibilidade, performance e qualidade, é fundamental para otimizar o desempenho geral da produção. Ao integrar essas métricas com a categorização detalhada das paradas de máquinas, o sistema PIMS proporciona uma visão abrangente e completa da operação, permitindo que as decisões sejam tomadas com base em dados confiáveis e precisos. Essa abordagem é crucial para otimizar os processos, reduzir os tempos de inatividade e aumentar a eficiência das máquinas, contribuindo diretamente para a melhoria contínua da eficiência operacional da indústria como um todo.

CONCLUSÕES

Conclui-se que o sistema desenvolvido apresenta uma solução para o monitoramento em tempo real dos principais indicadores de produção, como disponibilidade, performance, qualidade e OEE, dentro do contexto da Indústria 4.0. A digitalização dos dados de chão de fábrica permite a otimização dos processos produtivos, a redução dos tempos de parada e uma melhoria significativa na tomada de decisões, todas baseadas em dados tratados e filtrados. A interface intuitiva, acessível a operadores e gestores, facilita a visualização dos dados e proporciona uma ação rápida frente a eventos críticos e oportunidades de melhoria no processo produtivo. Essa acessibilidade em tempo real garante maior controle operacional, resultando em maior eficiência e agilidade na resposta a desafios diários.

Além disso, o sistema destaca-se pela sua flexibilidade e potencial de expansão. A possibilidade de integrar a solução a sistemas de gestão corporativa, como ERP e MES, abre caminho para futuras melhorias, ampliando o fluxo de informações e o controle entre o chão de fábrica e os níveis estratégicos da organização e até mesmo a implementação do sistema em uma linha de produção realizando uma comparação direta entre produção sem e com o PIMS desenvolvido.

CONTRIBUIÇÕES DOS AUTORES

Gabriel Victorio Ditomaso: Desenvolvimento do *script* e interface em *Python*. José Roberto de Lima: Desenvolvimento de código do ESP32 para aquisição e análise de dados. Alison José Pereira Neri: Redação do manuscrito e auxílio na criação de interface. Allan Michael Correa Amaral: Pesquisa bibliográfica e auxílio no manuscrito. Jair de Martin Junior: Orientação e revisão do manuscrito.

REFERÊNCIAS

AGHENTA, L. O.; IQBAL, M. T. Low-cost, open source IoT-based SCADA system design using thinger. IO and ESP32 thing. **Electronics**, v. 8, n. 8, p. 822, 2019. Disponível em: <https://www.mdpi.com/2079-9292/8/8/822>. Acesso em: 18 ago. 2024.

BEZERRA, A.; GREATI, V.; CAMPOS, V.; SILVA, I.; GUEDES, L. A.; LEITÃO, G.; SILVA, D. Enabling interactive visualizations in industrial big data. **IFAC PapersOnLine**, v. 53, n. 2, 2020. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2405896320305723>. Acesso em: 18 ago. 2024.

CARVALHO, F. B.; TORRES, B. S.; FONSECA, M. O.; SOARES FILHO, C. Sistemas pims-conceituação, usos e benefícios. **Tecnologia em Metalurgia, Materiais e Mineração**, v. 1, n. 4, p. 1-5, 2005. Disponível em: <https://tecnologiammm.com.br/doi/10.4322/tmm.00104001>. Acesso em: 18 ago. 2024.

JOAQUIM, R. C. **Novas Tecnologias para Comunicação entre Chão de Fábrica e o Sistema Corporativo**. 2006. 75f. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2006. Disponível em: <https://www.teses.usp.br/teses/disponiveis/18/18135/tde-26112006-232833/pt-br.php>. Acesso em: 18 ago. 2024.

NERI, A.; PEREIRA, V. Linguagem Python no laboratório didático de física: uma alternativa para análise de dados. **CONICT - Congresso de Inovação, Ciência e Tecnologia**, 2023. Disponível em: <https://ocs.ifsp.edu.br/conict/xivconict/paper/view/9478/3669>. Acesso em: 18 ago. 2024.

PAMPLONA, H. **Desenvolvimento de uma ferramenta para extração, comparação e validação de conformidades entre dados históricos armazenados em um PIMS**. Orientador: Dr. Marcelo Stemmer. 2024. p. 24-36. TCC (Graduação) - Eng. de Controle e Automação, Universidade Federal de Santa Catarina, Florianópolis, 2024. Disponível em: <https://repositorio.ufsc.br/handle/123456789/254669>. Acesso em: 18 ago. 2024.