IoT Platform to monitor Industrial Machines and perform PdM at the edge: A case study

Lahis Almeida*, Erimar Farias*, Vicente Farias*, Camilo Sidou*, Brunna Talita*, Rafaela Lima*, Suelem Valadão‡, Lívia Araújo*
**R&D Projects Department, INDT - Instituto de Desenvolvimento Tecnológico, Manaus, Brazil*
‡R&D Technical coordination of projects, ADATA Eletronics, Manaus, Brazil
{lahis.almeida, erimar.farias, vicente.farias, camilo.sidou, brunna.talita, rafaela.lima, livia.araujo}@indt.org.br
suelem valadao@adata.com

***Abstract*— Predictive maintenance plays a crucial role in Industry 4.0 ensuring efficiency, product quality and a safe working environment. Edge computing is another trend that optimizes Smart Factories applications, bringing processing and decision making closer to the data collection end node points, reducing network traffic and latency. This paper proposed an IoT platform to monitor industrial machines and perform predictive maintenance at the network edge. The effectiveness of the proposed platform is validated in a real industrial application project. Among the models considered in this study, the Decision Tree model emerged as the most effective (reaching 96% of accuracy) for the industrial machines analyzed.**

Keywords—predictive maintenance, pdm, machine learning, edge computing, industry 4.0, esp32 board.

I. Introduction

This Industry 4.0 revolutionizes traditional industrial automation by introducing advancements that lead to the creation of smart factories [1]. These facilities incorporate emergent technologies such as advanced robotics, high computing power and connectivity [2]. Notably, remarkable advancements in the fields of Artificial Intelligence (AI) and Internet of Things (IoT) are fueling unprecedented technological and scientific developments [1]. By harnessing the potential of Industry 4.0, smart factories are able to offer customers a range of new services and products that exhibit higher levels of efficiency and quality standards compared to previous approaches [2].

Predictive maintenance (PdM) stands out as one of the most prominent opportunities discussed in the realm of Industry 4.0 [2]. It relies on ongoing monitoring of machine or process integrity, enabling maintenance to be carried out solely when necessary. Furthermore, PdM facilitates early detection of failures through the application of predictive tools that leverage historical data (i.e., machine learning techniques), statistical inference methods and engineering strategies[3]. In an industrial environment, the implementation of PdM applications has the potential to optimize production profits while minimizing costs and losses, including those associated with assets [4].

In PdM applications, the challenges of handling big data are magnified by the need for real time monitoring and processing of a significant volume of data generated by sensors [1]. This poses difficulties in ensuring satisfactory metrics such as latency, scalability, and network bandwidth, particularly when immediate action is required to prevent failures caused by predicted events [4]. To address this, another significant trend emerges within the Industry 4.0 paradigm: Edge computing. By bringing processing closer to the data collection point and delegating non-time-sensitive tasks to the cloud, edge computing effectively reduces the strain on the network [5]. This approach optimizes the distribution of processing workloads across edge nodes, alleviating core network traffic and enhancing overall application performance [6].

Within all this context of Industry 4.0 and smart applications, the company ADATA Eletronics[1] (Amazonas - Manaus' site), unfortunately did not have an automated and integrated control of its facilities equipment (e.g., Generator, Compressor, QTA, Trafo), leading to some unwanted losses and assets costs. In contrast to this scenario, the Utility Automation Research and Development (R&D) project was created, which consists of a partnership between INDT (*Instituto de Desenvolvimento Tecnológico*) and ADATA team, whose objective is to improve ADATA's maintenance process through the development of a monitoring and control platform for industrial utility equipment (i.e., Generator, UPS, QTA, Chiller). Thus, the main contribution of this work is to (i) propose an IoT platform for monitoring ADATA industrial machines and perform PdM at the edge of the network; (ii) testing and validate the proposed platform together with the client; (iii) presents all steps to performs PdM at the edge and (iv) evaluates PdM models through machine learning and performance metrics.

The rest of the paper is organized as follows: Section II provides an overview of related work that involve PdM application at the edge, Section III describes the proposed approach employed in our research, Section IV presents the platform experiments, deploy and the discussion of the results obtained; and, finally, Section VI presents the conclusions and future works directions.

[1]https://www.adata.com/br

ISSN 2447-8326. V.1 © 2023 BTSYM

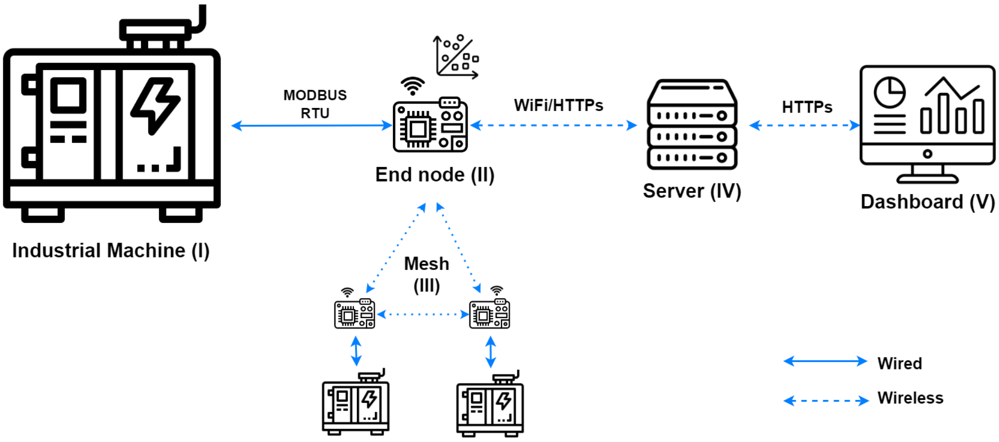

Fig. 1. Utility Project architecture.

II. Related Work

The continuous advancement of the AI and IoT technologies has garnered significant attention in the research community towards the PdM field [7]. In [8], the authors found that an IoT PdM system had the potential to enhance production throughput, minimize maintenance costs, and facilitate effective decision-making. Their research showed that several design considerations are taken into account when this kind of IoT system is proposed, including the types of analysis available, communication technology options, and the sensors required for data collection.

In [5] study, a data driven PdM system was developed for production lines in manufacturing. They utilize real-time data from IoT sensors and employ machine learning methods to identify signals indicative of potential failures. The evaluation results demonstrate the effectiveness of the proposed PdM system and indicate that boosting and bagging ensemble models perform well when the output of the model represents the estimated useful time remaining before a failure.

In [9], the authors proposed a PdM architecture based on containerized micro services on intelligent edge devices together with a hybrid model which fuses generalized fault trees (GFTs) and anomaly detection. Their results revealed a significant estimated reduction of over 63% in current maintenance costs. In [10], authors focused on estimating the remaining useful life (RUL) of electronic components through a three-step process: selecting fault features using variance analysis, measuring fault severity with relative entropy distance, and conducting fault prognosis using the gradient boosting decision tree (GBDT) model. Their results demonstrated superior prediction accuracy compared to other distance-measuring methods for determining the RUL of electronic elements.

The literature covers a wide variety of ML techniques for PdM (i.e., supervised classification, anomaly detection, regression in high dimensional data), each with specific characteristics and applications [1][7]. The authors of [11] used ANN predictive maintenance in machine centers. However, results showed that the error correction and compensation process is mainly dependent on the types of machines and processes. In [12] work, a total of six distinct supervised learning techniques were utilized for the PdM approach. These methods encompass k-Nearest Neighbour (kNN), Back-propagation Feed-forward Neural Network (FFNN), Decision Tree (DT), and Naïve Bayesian. The outcomes obtained from these approaches demonstrated significant advantages in terms of economic benefits when compared to conventional reactive maintenance and time-based maintenance practices. The findings revealed an estimated cost reduction of 40%, coupled with a remarkable six-fold decrease in the number of breakdowns.

In this PdM context, the present work proposes to contribute with insights and discussions about the PdM solutions through an IoT platform to monitor industrial machines and perform PdM at the edge of the network. Additional contributions concern in showing all challenges and steps to perform PdM at constrained devices (32 bit architecture) in a real world industrial application.

III. Proposed approach

This paper proposes a methodology for monitoring electrical equipment in industrial environments, enabling end users, in this case, ADATA's facilities team, to remotely monitor the behavior of this equipment, identifying anomalies present in them. In addition to monitoring the variables of these devices (e.g. voltage, current, power factor), a PdM module is also proposed at the edge, i.e., at the end nodes, in order to mitigate network latency and enable faster decision making. The following sections describe the monitoring architecture and the project architecture is presented in Fig. 1.

The first stage of the proposed approach consists in establishing communication between the industrial machine (I) and the end node (II). The industrial machine component represents the equipment that will be monitored. In ADATA, there are equipments located in outdoor and indoor environments.

The end node (II) component of the Utility Control project is the component responsible for wired communication with the machines and inductive sensors installed in the ADATA plant; and for wireless communication with the Dashboard

Server (IV) through the WiFi/WiFi Mesh Network (III). It is also responsible for inferring the PdM models in the respective selected equipment, sending the respective alert notifications if any anomalous behavior is identified.

Most equipment such as QTA, UPS, Generator, Compressor, have HMIs (Human Machine Interface) with communication interface via MODBUS RTU, enabling communication with the end node of the solution. However, there are some devices that do not have this interface such as the Diesel Tank, Exhausters and Insulators; therefore, for these cases, isolated sensors were installed and communication with them was also done through the end node, by protocols (e.g., I2C, 4-20mA).

After the connection established between industrial machine and end node, the collection of the variables of interest of the machine is carried out and the data are transmitted through the WiFi Mesh network (III) to a central end node, which follows the transmission, via traditional WiFi and the HTTPs protocol, sending them to the Server (IV) and the developed Dashboard software (V), This last component aims to present the collected variables in various visualization formats (e.g., widgets, pop-ups, graphs) for better monitoring of the facilities team. In this work, WiFi Mesh network will not be described in detail given the complexity of its arrangement. However, in future opportunities, mesh network coverage results can be detailed and disseminated.

IV. Experiments and Results

The experiments of this work aimed to validate the proposed IoT monitoring platform as well as to evaluate the predictive maintenance AI module at the edge. The following subsections describe the experiments and obtained results.

A. *End node*

The end node PCB design of the Utility Automation project was developed using Altium Designer[2] software in version 22.1.2. The System on a Chip (SoC) used in the project was the ESP32-WROOM-32UE, manufactured by Espressif, due its main features are the support for WiFi 2.4GHz, 32 bit microprocessor with two cores (Xtensa), 26 GPIOs and connector for external antenna, thus meeting the hardware requirements of the our project. The main circuits designed for the end node were: recording circuit (USB/TTL), interface with digital (e.g., temperature, water flow) and analog sensors (e.g., current, pressure) and the communication interfaces (i.e, I2C, RS485 (MODBUS RTU), ADC). The fabricated and assembled PCB can be visualized in Fig. 2.

The implementation of the firmware modules in the end node was done using the official *Espressif* framework, ESP-IDF (version 4.4). The languages used for implementation were C/C++. In addition to the communication modules with the sensors and machines, WiFi/WiFi Mesh communication was implemented and, in the application layer, the HTTPS protocol was implemented. In the firmware, a data persistence module was also implemented, using SPIFFS (SPI Flash File System), so that even in an eventual reset caused by an energy peak, the board does not lose registration information such as ID and commands coming from the server.

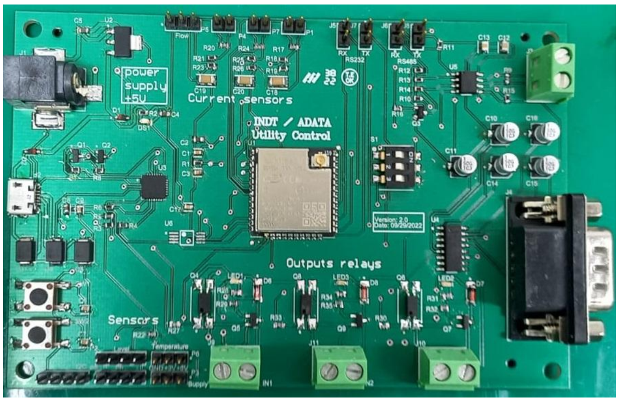

Fig. 2. End node.

Finally, the firmware also had the PdM module, which is responsible for performing the inferences of the PdM models in real time. To manage all firmware modules (i.e., sensor interfaces, HTTPS communication, PdM inferences), the code was built based in FreeRTOS[3] tasks approach.

B. *Dashboard*

The implementation of the Dashboard software screens was done using the AngularJS web framework, with the Angular Material UI library to implement most of the screen components, following the *Material Design* standard. For the development of the API, the ASP.NET Core framework was chosen, following the REST standards for web application development. The choice of ASP.NET Core is due to some of the existing systems in the ADATA company having been previously developed in the same language, C#, in addition to being a very robust web application development platform, offering a series of tools and development libraries that facilitate the delivery of functionalities [13].

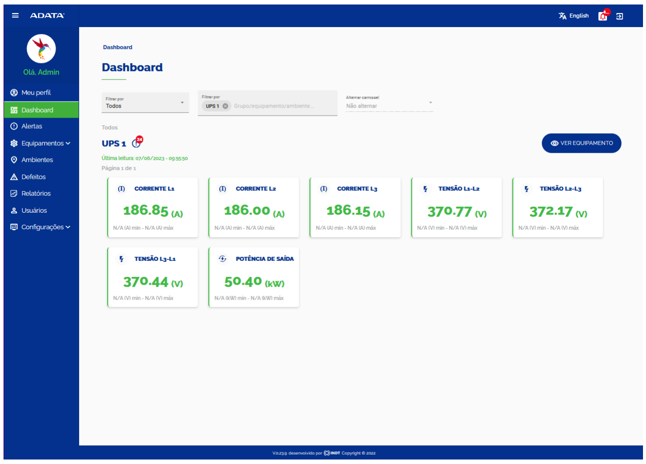

Fig. 3. UPS monitoring screen.

The open source relational PostgreSQL system was chosen as a database due extensive documentation and support on different platforms and libraries [14]. For integration between the web application and its database was used the Entity Framework Core, which is a native integration solution for web applications developed in ASP.NET Core. Fig. 3 shows one of the dashboard screens, presenting one of the monitoring screens (UPS machine Dashboard).

C. Predictive maintenance at the edge

After analysis in the literature (Section II) and alignments with the client, it was decided to treat the PdM module of the Utility Automation project as a type of binary classification problem (Supervised Learning), that is, the output of the model would be 0, characterizing a normal behavior in the respective machine (not anomalous); or 1, characterizing an abnormal problem in the machine (anomalous).

The predictive maintenance models embedded in the end nodes will perform anomaly or non-anomaly inferences according to the information collected from the respective equipment selected to have the PdM module. Thus, for each UPS type equipment, i.e., the model that was trained with the UPS equipment information will be embedded in the end node for monitoring this equipment. The steps for the construction of the PdM module are:

- Definition of the target machines of the PdM module;
- Data collection;
- Data preprocessing;
- Training of the models;
- Deployment of models in the end nodes;
- Evaluation of PdM models.

Following subsections detail the completion of these steps.

1) PDM module target machines: the equipment chosen to have the PdM module were the QTA and UPS. The QTA (Automatic Transfer Board) equipment is responsible for triggering the start of equipment (e.g., generator sets) immediately after the interruption/failure of the Energy Provider's power, without the manual exposure of employees. Thus, in situations where the main power, i.e. the power supplying ADATA, is interrupted, the QTA automatically triggers the company's internal generators so that no factory environment is preadjudicated [15]. UPS (Uninterruptible Power Supply) equipment is used to protect data centers and electrical equipment that have a high level of criticality, such as those operating on production lines. The main function of the UPS is to supply power to critical loads when the main power fails. The UPS differs from the QTA in that its response time is instantaneous, as it delivers energy stored in its internal batteries, and is sufficiently designed to withstand periods of time until the main power is switched to a secondary power source (Generator) [16]. In this context, it is possible to observe the great relevance of these two functions for ADATA, because without them, an eventual power failure could cause great losses in its manufacturing processes and industrial tests. Fig. 3 shows the UPS industrial equipment.

2) Data collection: the implementation of the PdM module at ADATA faced a number of challenges since its conception. ADATA's site in Manaus is recent (~ 2 years) and, since the installation of the industrial machines (e.g., Generator, QTA, Compressor, UPS) there was no history of

Fig. 4. UPS machine.

failures in this equipment. The monitoring logs of the equipment were also not accessible and the bureaucracy between ADATA and the manufacturers of these equipment was hampered by the costs of technical visits and availability of the technicians outsourced by them; making it impossible to use the logs as dataset in our project without causing losses in delivery time.

In this context, data field collection was needed. The data collection performed through MODBUS RTU communication between the endnode and the respective equipment lasted approximately 1 month. This period was defined according to the planning and delivery deadlines of the Utility Automation project. After this period, the collected data were exported through the Dashboard software itself in the *csv* format. For the QTA equipment, the variables collected were the three phases of the input voltage (V), three phases of the input current (A), power factor, frequency (Hz) and instantaneous consumption (kWh), while for the UPS equipment, the variables collected were the three phases of the output voltage (V) and three phases of the output current (A). Data collection frequency for both equipment was 20Hz. For QTA, a total of 197.337 samples were purchased, and for UPS, 196.453.

3) Data preprocessing: in this phase, the raw data collected undergo treatments that allow them to be more robust for analysis. The first treatment employed on the data was the separation of samples into binary classes: abnormal and normal. For samples that presented values within the expected ranges, the data were labeled as normal. Otherwise, they were labeled as abnormal. After the separation into classes, the missing data check was performed, i.e., if there was missing data between the sensor readings of the collected samples; and if the classes were balanced. After the missing data check, it was possible to observe that there was no occurrence of missing data. However, the two datasets were unbalanced. The QTA dataset had 189.455 of normal behavior and 7.882 of abnormal behavior. The UPS dataset had 196.322 of normal behavior and 131 of abnormal behavior. To mitigate this problem, the oversampling technique (90%) was performed on the data. The anomaly data that were generated were based on the maximum and minimum ranges of each of the variables collected in the samples. As a consequence, the abnormal class of the QTA dataset became 189.430 and the UPS, 196.586.

4) Training of the models: The PdM models were trained on a Dell Vostro 15 model 5510 notebook, with the 11th Generation Intel Core I7-11390H processor with a frequency of 3.4GHz; and 16GB memory and 512GB SSD. The algorithms chosen to train the models were Decision Tree, Random Forest and Naive Bayes as they are widely used in PdM classification problem perspective.

5) Deployment of the respective models in the end nodes: in this step, PdM models are ported to 32-bit architecture (Xtensa chip). The models are compressed into a .h file, using the *MicroML generator*[4] library, and from there it is possible to insert it into the firmware project and perform the model inference calls. To perform the inference correctly, the same vector of variables per sample (feature vector) considered in the training must be used, so, for UPS, for example, the input vector must always be the three phases of current and output voltage of the equipment. Table 1 shows a comparison between the size of the compressed model and the characteristics of its operation in the end node.

6) Evaluation of PdM models: the metric chosen to evaluate the classification algorithms was Accuracy, as it is responsible for measuring the percentage of correct predictions that an AI model made [17]. Regarding the performance of the model on the end node, some interesting features were listed such as compressed model size and inference time. All these evaluated methods and data insights can assist to choose the most indicated model to be deployed in which end node.

Evaluating Table I and II, the algorithm that had the best performance on UPS end node was Decision Tree, since its accuracy result (98 %) was superior to the others. Another differential of this model, was inference time that leads to a better latency and very compact model size, facilitating its execution on firmware project.

TABLE I. UPS PdM compressed models Results

	UPS		
Algorithm	**Accuracy**	***Compressed Model Size***	***Mean inference time***
Naive Bayes	87 %	2.4 kB	34 us
Decision Tree	**98 %**	**1.2 kB**	**6 us**
Random Forest	93 %	228 kB	-

For the QTA end node, Decision Tree and Random Forest models were the best results (both with 96%). Although, Random Forest compressed model are larger than others, besides its port process to a *cpp* lib was more complex due code size generated. Thus, for QTA equipment it was also chosen to be the Decision Tree model to be used in the PdM module.

TABLE II. QTA PdM compressed models Results

	QTA		
Algorithm	**Accuracy**	***Compressed Model Size***	***Mean inference time***
Naive Bayes	84 %	1.6 kB	10 us
Decision Tree	**96 %**	**1.9 kB**	**13 us**
Random Forest	96 %	3.4 kB	-

Once these steps were completed, the IoT project to monitor Industrial Machines and perform Predictive Maintenance at the edge was implemented at ADATA, which led to the automation and integration of Adata's industrial equipment. ADATA's facilities team validated the developed software, as well as the installed sensors and end nodes. Regarding the PdM module some occurrences have already been identified. All related to power surges or any possible machine shutdown, proving that the deployed model can detect anomalous behaviors and notify, through alarm notifications on the developed Dashboard. Fig. 5 shows the anomaly notification screen.

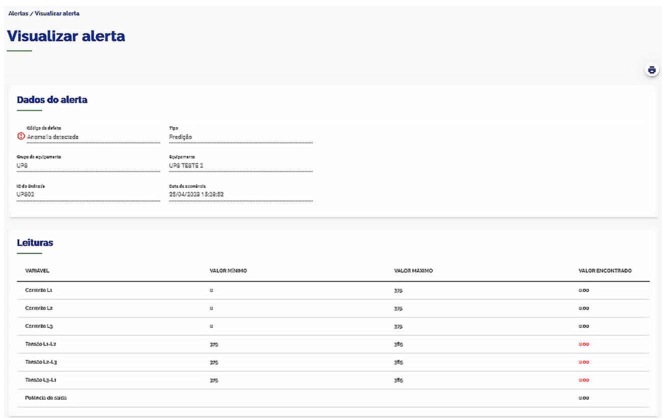

Fig. 5. UPS detected an anomaly alarm screen.

V. Conclusion

In this research, an IoT platform was developed to monitor and control the equipment of ADATA's industrial plant, improving their maintenance process. In addition, a PdM module on the edge was also developed, helping in the decision making of maintenance schedules.

The results of the research were successful as they showed that the proposed flow approach for the IoT platform worked well in the industrial environment considered. The INDT and ADATA team conducted thorough testing of the end node and Dashboard components, validating the widget views and CRUD operations in the developed software.

Despite the challenges faced in the implementation of the AI module, such as difficulty in obtaining the equipment operation logs for use as dataset; it was possible to trace an effective flow of steps from data collection in the field to the porting of the models to the end nodes of the selected equipment (QTA and UPS). The two trained models showed accuracy above 96% and the adopted flow can be used as a basis for more equipment to have PdM module in future works. Thus, we hope that our research will contribute to discussions and insights around PdM and Edge computing areas.

Future opportunities for the PdM module include: dealing with the PdM problem from the Regression perspective, as the Remain Time Useful information may also be interesting for the facilities team. In the data pre-processing stage, employ feature extraction and/or dimensionality reduction in the samples collected from the equipment and study the computational cost of employing these techniques in 32 bits devices.

From the firmware improvements perspective, the implementation of an OTA module would be of great value as it would optimize the time spent on firmware update in the field. As for the hardware module, the adoption of *borne kre* connector terminals and a hardware reset circuit (watchdog) would increase the reliability and robustness of the developed prototypes. Another relevant study would be a coverage analysis of the Mesh network adopted in the project, as well as analysis of metrics such as Throughput, Latency and Packet Delivery Rate. Thus enabling optimization in the choice of end nodes that will play the role of root node and child node in mesh topology.

Acknowledgment

The experiments and results presented in this publication were obtained through the research and development activities of the Utility Automation project, a partnership between INDT - Instituto de Desenvolvimento Tecnológico and ADATA Eletronics BRAZIL S/A, supported by SUFRAMA under the terms of Federal Law No. 8248/91.

References

[1] J. Dalzochio, R. Kunst, E. Pignaton, A. Binotto, S. Sanyal, J. Favilla, and J. Barbosa, "Machine learning and reasoning for predictive maintenance in industry 4.0: Current status and challenges," Computers in Industry, vol. 123, p. 103298, 2020.

[2] M. Compare, P. Baraldi, and E. Zio, "Challenges to iot-enabled predictive maintenance for industry 4.0," IEEE Internet of Things Journal, vol. 7, no. 5, pp. 4585–4597, 2019.

[3] T. P. Carvalho, F. A. Soares, R. Vita, R. d. P. Francisco, J. P. Basto, and S. G. Alcal ́a, "A systematic literature review of machine learning methods applied to predictive maintenance," Computers & Industrial Engineering, vol. 137, p. 106024, 2019.

[4] H. Hua, Y. Li, T. Wang, N. Dong, W. Li, and J. Cao, "Edge computing with artificial intelligence: A machine learning perspective," ACM Computing Surveys, vol. 55, no. 9, pp. 1–35, 2023.

[5] Z. Liu, C. Jin, W. Jin, J. Lee, Z. Zhang, C. Peng, and G. Xu, "Industrial ai enabled prognostics for high-speed railway systems," in 2018 IEEE international conference on prognostics and health management (ICPHM). IEEE, 2018, pp. 1–8

[6] "Predictive maintenance and the smart factory," https://www2.deloitte.com/content/dam/Deloitte/us/Documents/process-and-operations/us-cons-predictive-maintenance.pdf, 2023, accessed: 2023-07-17.

[7] S. Ayvaz and K. Alpay, "Predictive maintenance system for production lines in manufacturing: A machine learning approach using iot data in real-time," Expert Systems with Applications, vol. 173, p. 114598, 2021.

[8] B. Sicard, N. Alsadi, P. Spachos, Y. Ziada, and S. A. Gadsden, "Predictive maintenance and condition monitoring in machine tools: An iot approach," in 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS). IEEE, 2022, pp. 1–9.

[9] P. Nunes, E. Rocha, and J. P. Santos, "Using intelligent edge devices for predictive maintenance on injection molds," Applied Sciences, vol. 13, no. 12, p. 7131, 2023.

[10] L. Wang, D. Zhou, H. Zhang, W. Zhang, and J. Chen, "Application of relative entropy and gradient boosting decision tree to fault prognosis in electronic circuits," Symmetry, vol. 10, no. 10, p. 495, 2018.

[11] Z. Li, Y. Wang, and K.-S. Wang, "Intelligent predictive maintenance for fault diagnosis and prognosis in machine centers: Industry 4.0 scenario," Advances in Manufacturing, vol. 5, pp. 377–387, 2017.

[12] B. Schmidt and L. Wang, "Predictive maintenance of machine tool linear axes: A case from manufacturing industry," Procedia manufacturing, vol. 17, pp. 118–125, 2018.

[13] "Documentação do ASP.NET - Desenvolver aplicativos ASP.NET core," https://learn.microsoft.com/pt-br/aspnet/core/, 2023, accessed: 2023-06-18.

[14] "Postgresql: The world's most advanced open source relational database," https://www.postgresql.org/, 2023, accessed: 2023-06-18.

[15] "O que é o qta?" https://www.ageradora.com.br/o-que-e-qta/, 2023, accessed: 2023-06-11.

[16] "Transferência de energia elétrica," https://apgenergia.com.br, 2023, accessed: 2023-06-11.

[17] "Metrics to evaluate your machine learning algorithm," https://towardsdatascience.com, 2023, accessed: 2023-06-18.