INSTITUTO POLITÉCNICO DE BRAGANÇA
Escola Superior de Tecnologia e Gestão

Intelligent and Predictive Maintenance in Manufacturing Systems

Ana Margarida Lima Cachada - 27174

Dissertation presented to the School of Technology and Management of Bragança to obtain the Master Degree in Engenharia Industrial.

Work oriented by:

Prof. Dr. Paulo Jorge Pinto Leitão

Prof. Dr. José Fernando Lopes Barbosa

Bragança

2018

INSTITUTO POLITÉCNICO DE BRAGANÇA
Escola Superior de Tecnologia e Gestão

Intelligent and Predictive Maintenance in Manufacturing Systems

Ana Margarida Lima Cachada - 27174

Dissertation presented to the School of Technology and Management of Bragança to obtain the Master Degree in Engenharia Industrial.

Work oriented by:

Prof. Dr. Paulo Jorge Pinto Leitão

Prof. Dr. José Fernando Lopes Barbosa

Bragança

2018

To my family, close friends and everyone who helped me get where I am today

Acknowledgement

Here I would like to express my gratitude towards everyone who helped me through the process of developing and writing this thesis.

First and foremost, I would like to thank my supervisors, Prof. Paulo Leitão and Prof. José Barbosa, for the mentoring, motivation and advices offered throughout the development of this work.

I would also like to express my gratitude to my fellow lab colleagues, José Barbosa, Adriano Ferreira, Diana Pinho, Jonas Queiroz, Flávia Pires, Vitor Sousa, Marcelo Alves, Luís Piardi, David Costa, Hasmik Badikyan and Nelson Rodrigues. Thank you for providing a great work environment, for your friendship and your patience. Also, to Américo Alves and Leandro Alves for contributing to the development of this work.

To all my friends who were a very important pillar, not only in the development of the thesis, but in the most diverse moments of my life, thank you. A special thanks to Cláudio, Mel and Flávia who have been there bearing with my good and bad moments.

Last but not least, my sincere thanks to my whole family for always being my safe harbour, especially to my parents and siblings for the constant concern and unconditional support. There are no words to express the importance of your presence in my life.

Without all of you none of this would be possible.

Forever grateful.

Abstract

In recent years manufacturing companies have been facing a major shift in the manufacturing requirements, for example the shift in demand for highly customized products resulting in a shorter product life cycle, rather than the traditional mass production of standardized products.

As a consequence of the change, the enterprises are facing the need to adapt, forcing all sectors of the manufacturing activity to move accordingly. Maintenance is one of the major activities in manufacturing as it highly influences production productivity and quality, and has a direct impact on production cost and customer satisfaction.

Nowadays, corrective and scheduled maintenance are widely implemented. However, the manufacturing world need to adapt to this new reality by implementing new, intelligent and innovative maintenance systems capable of predicting in advance possible failures. Lately, predictive maintenance systems and tools have been developed and continue to be studied and improved. However, companies do not have enough trust on these systems to fully rely on them.

Considering all these aspects, the work developed on this thesis introduces a system architecture for an intelligent predictive maintenance system based on the Condition-Based Maintenance (CBM) to be used in the Catraport case study, focusing particularly on the development of the monitoring module of the system architecture. This module comprises a tool developed by using Node-RED that displays the collected data alongside with the warnings triggered by cross-checking the incoming data with implemented decision rules, through the use of graphics and text. Additionally, an Android mobile application was also developed to allow consulting remotely the operating state of the assets.

Keywords: Intelligent maintenance; Internet of Things; Monitoring; Industry 4.0.

Resumo

Nos últimos anos, as empresas de manufatura têm enfrentado uma grande mudança nos requisitos de fabrico, nomeadamente, na procura por produtos altamente personalizados, resultando num ciclo de vida do produto mais curto, contrariamente à tradicional produção em massa de produtos padronizados.

Como consequência desta mudança, as empresas, bem como todos os setores da atividade de manufatura, enfrentam a necessidade de se adaptar. A manutenção é uma das principais atividades de fabrico, visto que influência fortemente a produtividade e a qualidade da produção, e tem um impacto direto no custo do produto e na satisfação do cliente.

Atualmente, as estratégias de manutenção corretiva e programada são amplamente implementadas. No entanto, o mundo da manufatura precisa de se adaptar à nova realidade, implementando sistemas de manutenção novos, inteligentes e inovadores, capazes de prever possíveis falhas. Ultimamente, os sistemas e ferramentas de manutenção preditiva têm sido desenvolvidos e continuam a ser estudados e melhorados. No entanto, as empresas não possuem confiança suficiente nesses sistemas para os implementar nas suas instalações.

Considerando todos esses aspetos, o trabalho desenvolvido nesta dissertação introduz uma arquitetura para um sistema inteligente de manutenção preditiva baseado na técnica Condition-Based Maintenance (CBM) a ser usado no estudo de caso da Catraport, focando-se particularmente no desenvolvimento do módulo de monitorização da arquitetura. Este módulo compreende uma ferramenta desenvolvida com recurso ao Node-RED que exibe os dados colecionados. Adicionalmente são apresentados avisos originados pelo cruzamento dos dados recebidos com as regras de decisão implementadas. Além disso, uma aplicação móvel Android também foi desenvolvida para permitir a consulta remota o estado operacional dos equipamentos.

Palavras-chave: Manutenção inteligente; Internet das Coisas; Monitorização; Industria 4.0.

Contents

List of Tables

List of Figures

Acronyms

AR Augmented Reality.

BPMN Business Process Model and Notation.

CBM Condition-Based Maintenance.

GRU Gated Recurrent Unit.

HMI Human-Machine Interface.

I2C Inter-Integrated Circuit.

ICT Information and Communication Technology.

IoT Internet of Things.

KPI Key Performance Indicators.

LSTM Long Short-Term Memory.

M2M Machine to Machine.

MQTT Message Queuing Telemetry Transport.

MTBF Mean Time Between Failure.

MTTR Mean Time To Repair.

OEE Overall Equipment Effectiveness.

OSA-CBM Open System Architecture for Condition-Based Maintenance.

PHM Prognosis and Health Management.

R2MPHM Real-time Remote Machinery Prognostics and Health Management.

RCM Reliability-Centered Maintenance.

RNN Recurrent Neural Networks.

RUL Remaining Useful Lifetime.

TPS Toyota Production System.

UI User Interface.

Chapter 1

Introduction

Throughout the years, with the constant evolution of society and the intensification of product demands, the competitiveness has increased, leading the organizations on a journey to try to find ways to improve their business. Ultimately what companies needed was to be more competitive and this brought an extremely high pressure upon the industrial sector, compelling companies to reduce the costs and increase the value of its assets and yet improve the quality of its products [1]. This pressure led all sectors related with the manufacturing world, such as marketing, production, engineering, among others, to improve themselves and to decentralize its functionalities.

One of the main concerns within the manufacturing world is associated to the preservation of the physical assets' optimal working condition, given the fact that all equipment, or asset, is unreliable since it degrades with age and/or usage and the failure occurs when it is no longer capable of delivering the products and services [2]. Consequently, one of the topics that has been highly discussed is the maintenance and all the functions inherent to this activity.

The existing maintenance concepts, strategies and techniques are applied in different sectors, such as construction, transportation, airline industry, power and manufacturing [1]. However, maintenance applied to the manufacturing sector is probably one of the most complex types of maintenance, when compared to the others, due to the number of variables related to the processes and to the volatility inherent to the industrial sector. Additionally, due to the impact that the manufacturing sector has in modern economy, business and society, the maintenance applied to this sector may be the most developed and matured when compared to the other sectors.

Historically, industrial maintenance started as a necessary evil, meaning that maintenance operations were executed only when strictly necessary. The primary goal of maintenance was to minimize the cost of the life cycle of a physical asset, and the physical assets only suffered maintenance interventions when a failure was detected, leading to sudden stops in the production and, consequently, unexpected loss of profit. For example, the ineffective maintenance shutdown of a petrochemical plant costs millions of dollars of production loss [1].

Over the years, the concepts, strategies and techniques related to maintenance evolved according to the needs and eventually the manufacturing sector realized that maintenance could play a major role in increasing the competitiveness of an organization in a globally competitive market [1] and this led to great changes in the maintenance concept. What was once seen as a necessary evil is now understood as a strategical factor and a profit contributor to ensure productivity in industrial systems [3], [4]. In other words, although maintenance has associated costs, often high, it is extremely important in order to guarantee the quality of production. Particularly in industrial environments, which are characterized by being stochastic, dynamic and chaotic, maintenance is essential to guarantee the stability and efficiency of production and, when performed correctly, is a great contributor to safety in the manufacturing environment as well as in the global industrial environment [1]. Also, maintenance is becoming a strategic decision instrument when it comes to asset acquisition, product design, customer satisfaction, and manufacturing sustainability [1].

During the course of industrial history several maintenance techniques were developed, such as schedule maintenance, condition based maintenance, reliability based maintenance and so forth [1]. Initially, maintenance was a production task and the prevalent technique was corrective maintenance. With the mechanization of processes and the increasing customer demands, organizations could not afford to wait until machines failed and maintenance became a technical matter and failure prevention was an issue, that empowered the creation and implementation of schedule maintenance which is based on the failure history of the assets and is implemented by the scheduling of maintenance interventions. Nowadays, due to process' automation maintenance techniques such as Condition-Based Maintenance (CBM) and Reliability-Centered Maintenance (RCM), are being developed and maturated [2], [5]. Meanwhile, organizations are seeking to take advantage of the most recent and intuitive technologies, such as Internet of Things (IoT), advanced data analysis, real-time monitoring and so forth, in order to facilitate their performance and gain safer, smarter and more sustainable environment [6]. All maintenance techniques mentioned previously, except the corrective maintenance, has reveal themselves very effective eliminating unexpected failures and unplanned unavailabilities [1].

The emergent interest in improve and develop maintenance concepts and techniques, combined with the advance of the Information and Communication Technology (ICT), has generated a wave of research that has contributed to the development of new, innovative and intelligent approaches.

The work presented in this thesis is part of the R&D project named Maintenance 4.0, which main goal is to develop an intelligent approach for the industrial maintenance, aligned with Industry 4.0 principles. This approach should consider advanced analysis of the data collected from the shop floor to monitor and detect earlier the occurrence of disturbances and consequently the need to implement maintenance interventions. Additionally, should be able to support the maintenance technician during the maintenance interventions by providing a guided intelligent decision support articulated by the use of human-machine interaction technologies. Given the nature of the project and case study the maintenance technique that respects all the requirement of the intended system is the CBM, therefore this work is fully aligned with this technique.

Thus, this thesis aims to develop an overall system architecture for an intelligent and innovative maintenance system and the development of the monitoring module that allows the monitoring of the condition of an asset and triggers alarms when a possible disturbance is detected or predicted. The designed tool should be composed by an engine that cross-checks the collected data with pre-defined rules and a visualization tool, being both developed using the Node-RED framework. Additionally, the development of an Android application is considered in this work, allowing the maintenance technician, or another person, to verify the system status out of the shop floor.

This thesis is organized as follows: Chapter 2 presents an overview on maintenance concepts and techniques, and presents the new tendencies in maintenance. The architecture of the design system is introduced in Chapter 3, which is followed by the presentation of the case study scenario in Chapter 4. The developed online monitoring solution is presented in Chapter 5. Finally, Chapter 6 rounds up this thesis with the conclusions.

The present work is inserted within the national project Maintenance 4.0, which has received funding from Norte 2020, Portugal 2020 and the União Europeia (FEDER).

Chapter 2

State of the Art in Maintenance Engineering

The definitions for this activity vary depending on the authors that study this subject. However, the majority of them agree that maintenance can be defined, in its narrow meaning, as a "*set of activities required to keep physical assets in the desired operating condition or to restore them to this condition"* [2]. In another words, the main purpose of industrial maintenance is to guarantee a manufacturing asset availability and reliability [1]. This was presented as one of the first definitions for this activity considering that, primarily, maintenance aimed to restore an asset to its operational state after a failure occurred. Nowadays, the objective of any maintenance program is to find a balance between the capability of a system and the costs [7].

The concept of maintenance has grown with the evolution of the manufacturing world, and has been through different stages as represented in Figure 2.1.

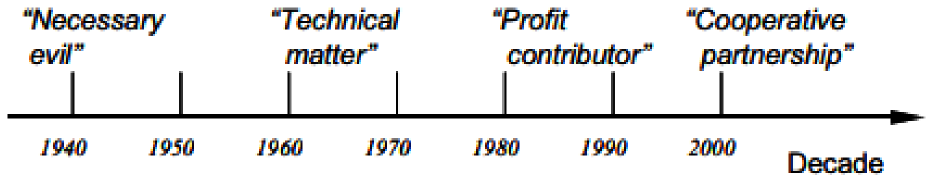


Figure 2.1: The maintenance function in a time perspective [2].

A representation of the evolution of manufacturing world is illustrated in Figure 2.2 where it is possible to establish a relationship among the type of manufacturing, its objectives and the correspondent enabling factors.

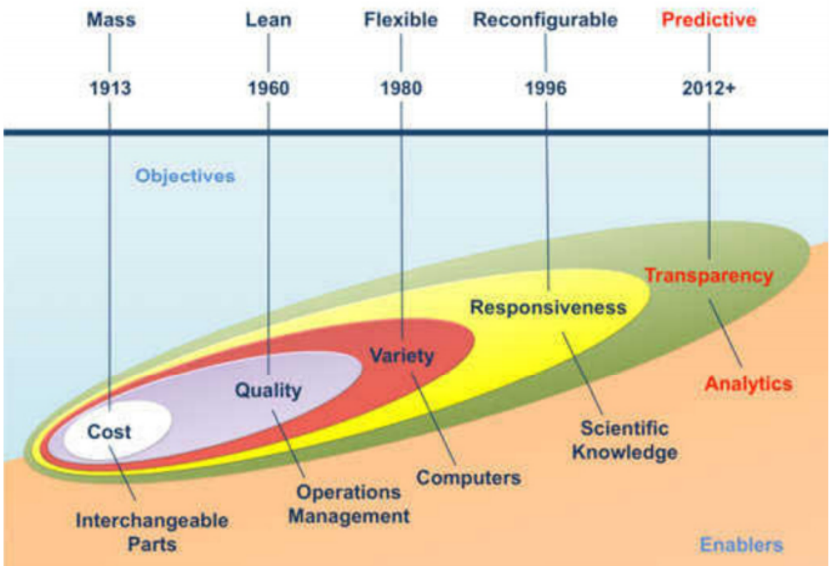


Figure 2.2: Evolution of the Manufacturing Systems [8].

After World War 1, due to the increasingly demand of goods and products, mass production assembly lines were established to produce large quantities of standardized units. This methodology, popularized by Henry Ford [9], minimized the production costs through the use of interchangeable parts where one part could promptly replace another rather then building an entire product from the beginning. The mass production strategy proved efficient for low-mix and high-volume applications [8]. Due to the fact that manufactures were concerned with only producing large quantities of products to satisfy custumers demands, maintenance was seen as a necessary evil that would only be performed when strictly necessary, meaning that maintenance tasks, i. e. repairs and replacements, were performed when a evident failure was detected and no optimization questions were raised [2].

Years later, in 1971, the Toyota Production System (TPS) was introduced, however this methodology started being developed around 1948 by Taiichi Ohno [10]. The TPS aimed to fill the gap between the western and the Japanese manufacturing industries through the application of methodologies that allowed to reduce the production costs by minimizing sources of waste, i. e. overproduction, waiting, transport, processing, inventory, motion and defects, while maintaining high quality standards that are expected of the product. In order to achieve the TPS goals, a sweeping of the operations management is involved. This methodology is based on two concepts, namely the Jidoka and Just-in-Time concepts, which intend to support the highlighting/visualization of problems and the productivity improvement [11]. As opposed to the mass production, the TPS is typically triggered by customer-pull or producing only when ordered, therefore the TPS is suitable for low-mix and low-volume applications [8]. In the western manufacturing world methodologies to enable the waste reduction were also developed, namely the Lean principles and six sigma techniques.

This change in the manufacturing paradigm helped enterprises to conceive that maintenance was an actual technical matter. This was translated in the optimization of technical maintenance solutions and it also involved attention of the organization on the maintenance work [2].

Then came the early forms of computers, such as numeric controllers, which allowed the automation of processes and provided more flexibility to equipment. This change enabled manufacturers to produce more variety and afford individual customization with costs that are comparable to standard goods and services.

Later on, industries have taken advantage of the information technology and social media networks which have influenced greatly consumers' perception on product innovation, quality, variety and speed of delivery [8]. This has created a competitive environment leading manufactures to change the manufacturing paradigm in order to fulfill customers demands. These conditions enabled the implementation of reconfigurable manufacturing approaches, which allow for a plant structure to change in a short period of time in order to increase production capacity. Due to these changes in the manufacturing reality, the maintenance concept suffered a changeover once again and was considered a profit contributor. Maintenance became then a mature function, instead of production sub-function [2].

Presently, the industry wants to integrate within the manufacturing systems another capability, namely transparency, which is the ability to discover uncertainties and quantify real manufacturing capability and readiness [12]. To achieve transparency the industry has to change its paradigm, transforming itself into predictive manufacturing [8]. Such transformation can be enabled by taking advantage of the high volume of data produced in the shop floor and the use of powerful predictive analytics so that data generated can be constantly processed and translated into valuable information, which allows making informed decisions.

In order to be harmonized with the industrial evolution, maintenance paradigm need to adapt once again, exploring the possibility of predicting failures to avoid unexpected shutdowns and products defects. This is executable through the use of the emergent technologies, such as IoT, ICT, among others. These technologies enable the possibility to monitor the assets' condition in real-time and to study which is the optimal set of work conditions that maximize the Overall Equipment Effectiveness (OEE) and improve performance.

Therefore, nowadays maintenance is not seen as an activity performed only to keep the physical assets operational but it has significantly evolved to the point where companies see the practice of maintenance as a strategical factor and its main goal is to contribute towards the organization's profit, arising the need for maintenance operations to be synchronized with the corporate objective [4]. Thus, as defined in [1] maintenance can be considered as a "set of activities, technical, administrative, and managerial, performed during the life cycle of an item, workplace or work equipment to preserve the value of an asset".

The value of an asset is evaluated by its reliability, availability, productivity and market value. The reliability of an asset measures its ability to function at any point in time. On the other hand, the availability of an asset is determined by the readiness of the asset to operate/produce [1]. The success and competitiveness of an enterprise is highly dependent on the output of the production system in terms of quantity, quality, and safety [1], thus maximizing the availability and reliability of the assets will lead to an increase of the organization's productivity. Producing the desired quantity of products, with the required quality specifications, in a timely manner will eventually be translated to an increase in the market value [2]. When this is attained in a cost-effective way, maintenance is then acting as a profit contributor. Such goal can only be attained with a highly effective and efficient maintenance system that maintains high rate of manufacturing equipment availability with long term maintainability capable of keeping high level of asset value [1].

An effective maintenance is a multidisciplinary system composed of plans and operations that guarantees material, spares, tools, human and financial resources availability at the right time with the required quality and quantity [1]. This proves that industrial maintenance is a complex system that is required to be linked with several departments or services to operate properly.

A proper maintenance system integrates several maintenance techniques according to the needs of the organization. Several maintenance techniques are exposed in the following section.

2.1 Maintenance Strategies and Techniques

The occurrence of unexpected disturbances in production leads to the loss of productivity which can result in the loss of business opportunities. Thus, in order to increase an organization's competitiveness, it is necessary to reduce production disruptions, which can be achieved through the implementation of maintenance strategies and techniques in the production environment.

There are different strategies to perform the necessary maintenance to the assets and these maintenance strategies can be grouped in different ways depending on the authors that study them.

Considering the distinct terminologies used by the authors, in this research work the terminology adopted consists of two main strategies, namely the Reactive or Corrective Maintenance and the Preventive Maintenance. The Reactive or Corrective Maintenance strategy can be divided into two techniques, the Immediate and the Differed Corrective Maintenance. On the other hand, the Preventive Maintenance strategy comprises two main techniques, explicitly the Schedule or Predetermined Maintenance and the Predictive Maintenance which includes all techniques that are designed to detect a failure prior to its occurrence. The adopted terminology is depicted in Figure 2.3.

The adopted terminology for this work, i. e. strategies and techniques, is detailed in the following sub-sections.

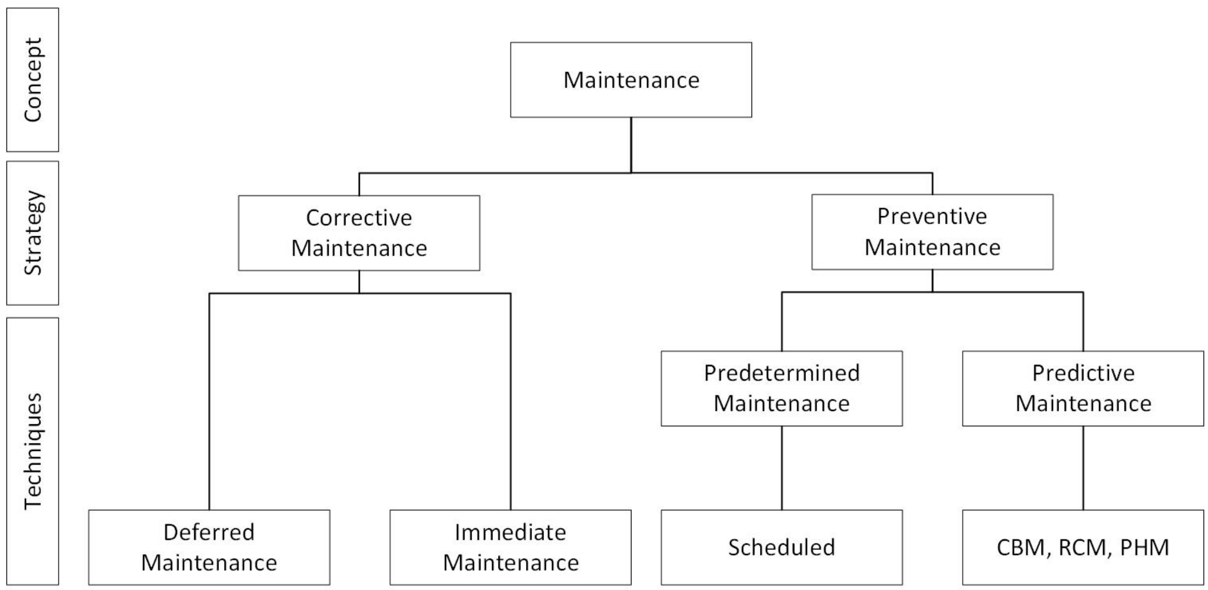


Figure 2.3: Adopted Terminology (adapted from [13])

2.1.1 Reactive or Corrective Maintenance Strategy

Reactive or Corrective Maintenance, also designated as breakdown or operate to failure maintenance, can be defined as an unscheduled maintenance intervention performed to restore an asset to its operating state [1], [4]. These interventions are reactive in nature, meaning that actions are only triggered after a breakdown or a loss of function, and this merely implies "wait until it breaks, then fit it!" [2], [13].

Corrective actions are difficult to predict as equipment failure behavior is stochastic and breakdowns are unforeseen, and when unpredicted failures happen they cause shutdowns, delays in production, and rise the need of unexpected and unplanned repairs, becoming one of the most expensive and costly type of maintenance activity [7]. To avoid high costs, Reactive or Corrective Maintenance should be applied only to non-critical situations where capital costs are small, consequences of failure are slight, no safety risks are immediate, and quick failure identification and rapid failure repair are possible [13].

When failure occur Corrective Maintenance interventions can be splitted into two actions, deferred and immediate maintenance. Deferred corrective maintenance occurs when a failure is detected and the actions to repair it can be schedule to a suitable time since the correction of the failure is not urgent . On the other hand there is the immediate corrective maintenance, which consists of perform maintenance interventions immediately after a failure is detected.

The replacement of a failed light bulb, repair of a ruptured pipeline and the repair of a stalled motor are some examples of corrective actions [2].

The implementation of preventive maintenance techniques aims to reduce the need of Reactive or Corrective Maintenance and consequently diminish the effects of unpredicted failures occurrence. However, Corrective Maintenance strategy can never be discarded due to the unpredictable behavior of machines and other assets.

2.1.2 Preventive Maintenance Strategy

This type of maintenance is more complex than the previous due to the fact that the Preventive Maintenance strategy attempts to diminish the failure probability of an asset and/or to anticipate, or avoid if possible, the consequences of a failure occurrence [2].

All techniques within the Preventive Maintenance strategy, such as Predetermined Maintenance, Condition-Based Maintenance, among others, scored great success in eliminating unexpected and unplanned unavailability's [1].

There are several tasks performed in the manufacturing world that represent the implementation of Preventive Maintenance, such as lubrication, bi-monthly bearing replacements, inspection rounds, vibration monitoring, oil analysis, design adjustments, among others [2]. These tasks can be included in different preventive techniques, namely Schedule or Predetermined Maintenance and Predictive Maintenance, which are further detailed in the following subsections.

Schedule or Predetermined Maintenance

The Schedule or Predetermined Maintenance is defined in [2] as an asset maintenance strategy based on replacing, overhauling or re-manufacturing an item at fixed or adaptive intervals, regardless of its condition at the time. This maintenance technique is used to reduce unexpected failure of critical assets and to promote improved safety, health and working environment conditions [1].

The main goal of the Schedule or Predetermined Maintenance is to diminish the failure probability of the physical assets, which can lead to the extent of its life time [1], [2]. This type of maintenance is suitable for failures which present a clear wear-out characteristic [14]. The frequency of Schedule or Predetermined Maintenance interventions is determined by the equipment, its age and condition, and consequences of its failure.

In order to determine interventions intervals, equipment reliability indexes, such as Mean Time Between Failure (MTBF) and Mean Time To Repair (MTTR), are used. The referred indexes provide a rough estimate of the time between two adjacent breakdowns and the mean time needed to restore a system when such breakdowns happen. Nevertheless, equipment degradation processes vary from case to case, and the causes of failure can be different as well [2].

In order to reduce the risks of failure, during or after an intervention, and to minimize total costs of the Schedule or Predetermined Maintenance while maximizing total benefits, it is very important to optimize the timing for the maintenance interventions [1].

Actions such as lubrication, bi-monthly bearing replacements and inspection rounds are some examples of Schedule or Predetermined Maintenance implementation.

Predictive Maintenance

Predictive maintenance involve all techniques that aim to early detect failure occurrence and, for this purpose, data generated in the shop floor is extremely important. However if the data is not processed in an effective way, providing context and meaning that can be understood, this data is not useful [15]. There are several techniques developed, or under development, used to implement predictive maintenance in manufacturing systems, such as Prognosis and Health Management (PHM), CBM, RCM, among others.

PHM is an engineering process where algorithms are used to detect anomalies, diagnose faults and predict the Remaining Useful Lifetime (RUL) of assets. Although the main goal of PHM is to provide the health state and estimate the RUL of the components or equipments, also financial benefits such as operational and maintenance cost reductions and extended lifetime are achieved [16]. In short, PHM is a method that evaluates the reliability of a system in its actual life-cycle conditions, in order to determine the advent of failure, and mitigate the system risks [17]. A PHM analysis involves a variety of steps including the collection of data and data characterization, the extraction of features from collected data, and finally the diagnosis and prognosis. The essential steps for implementing a PHM system are discussed in detail in [18]. The PHM concept is often used with other approaches like CBM [19].

The CBM technique aims to avoid unnecessary maintenance interventions by performing maintenance actions only when there is evidence of the abnormal behaviors of a physical asset [20]. It uses data generated in the shop floor in order to monitor, in a continuous or periodical manner, several operational indexes [2].

Initially, this technique only focused in the condition monitoring and diagnosis, but with the rising interest on a predictive manufacturing world, a prognosis layer was added by the Open System Architecture for Condition-Based Maintenance (OSA-CBM) [21], transforming the CBM in an important tool for predictive maintenance.

There are various international standards related to the CBM approach, for example the ISO 13374 [22] that addresses the OSA-CBM, held by MIMOSA [21], representing formats and methods for communicating, presenting, and displaying relevant information and data.

Lebold and Byington [23], based on the OSA-CBM, presented the CBM architecture which was divided into seven generic layers in order to attain a well constructed system, as shown in Figure 2.4.

This architecture has suffered slight changes and currently the OSA-CBM presents six functional blocks according to [21], namely (1) Data Acquisition, (2) Data Manipulation, (3) State Detection, (4) Health Assessment, (5) Prognosis Assessment, and (6) Advisory Generation. Comparing the current architecture and the one presented by Lebold and Byington [23] the major difference is the aggregation of the Layers 6 and 7, respectively Decision Support and Human Interface or Presentation Layer, into a single functional block, designated Advisory Generation. The function blocks of this architecture are defined as follows [21]:

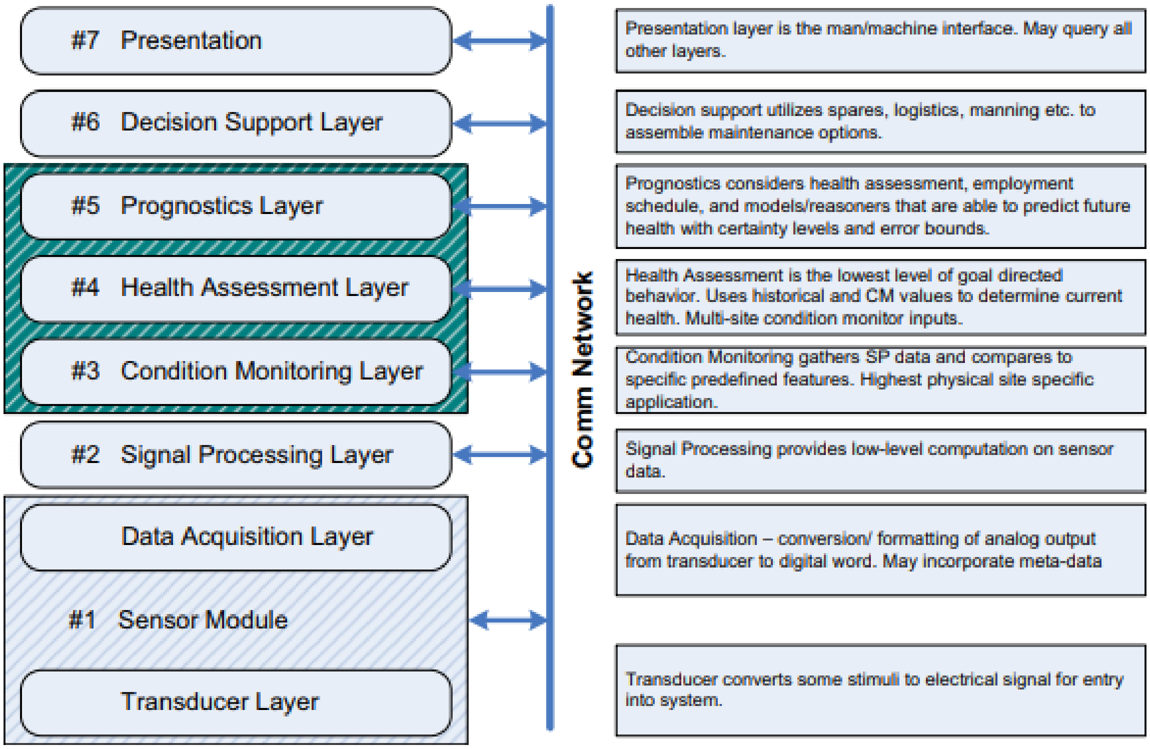


Figure 2.4: Architecture for a CBM system [13].

1. Data Acquisition: provides the access to digitized sensor or transducer data and records this data.

2. Data Manipulation: may perform single and/or multi-channel signal transformations and may apply specialized feature extraction algorithms to the gathered data.

3. State Detection: performs condition monitoring by comparing features against expected values or operational limits and returning conditions indicators and/or alarms.

4. Health Assessment: determines if the system's health is suffering degradation by considering trends in the health history, operational status and maintenance history.

5. Prognostics Assessment: projects the current health state of the asset into the future by considering an estimation of future usage profiles.

6. Advisory Generation: provides recommendations related to maintenance actions and modification of the asset configuration, by considering operational history, current and

future mission profiles and resource constraints.

For the OSA-CBM architecture, the data flow usually occurs between adjacent functional layers. Nevertheless, if required each layer may be able of requesting data from non adjacent functional layers [23]. CBM can be very helpful to predict the failure occurrence but its installation is costly, especially if the equipment is already installed and it is necessary to proceed to further instrumentation [13]. In these cases, it is important to decide if the installation of a CBM system justifies its costs.

When the condition of the component is combined with its importance from a functional point of view, we arrive at RCM. RCM is not a single maintenance method, but it allows the comparison of different maintenance methods, of which the most cost effective can be chosen without compromising reliability [24]. RCM is a technique that supports the implementation of a predictive philosophy in the industrial world and is a method for maintenance planning that was developed within the aircraft industry and later adapted to several other industries and military branches [2].

In IEC 60300-3-11 standard RCM is defined as a "systematic approach for identifying effective and efficient preventive maintenance tasks for items in accordance with a specific set of procedures and for establishing intervals between maintenance tasks." A major advantage of the RCM analysis process is a structured, and traceable approach to determine the optimal type of preventive maintenance [25]. Several standards and guidelines have been issued where the RCM methodology is tailored to different application areas, e.g., IEC 60300-3-11, MIL-STD-217, NAVAIR 00-25-403 (NAVAIR 2005), SAE JA 1012 (SAE 2002), USACERL TR 99/41 (USACERL 1999), ABS (2003, 2004), NASA (2000) and DEF-STD 02-45 (DEF 2000).

2.1.3 Optimal Balance between Maintenance Strategies

The ground for the emergence of new maintenance approaches have been to find an optimum balance between the costs of maintenance and the ability to maintain sufficient reliability of a asset or system [24].

When performing predictive maintenance, the optimum between remaining useful lifetime of parts and the chance of downtime is when maintenance should take place [26], as depicted in Figure 2.5.

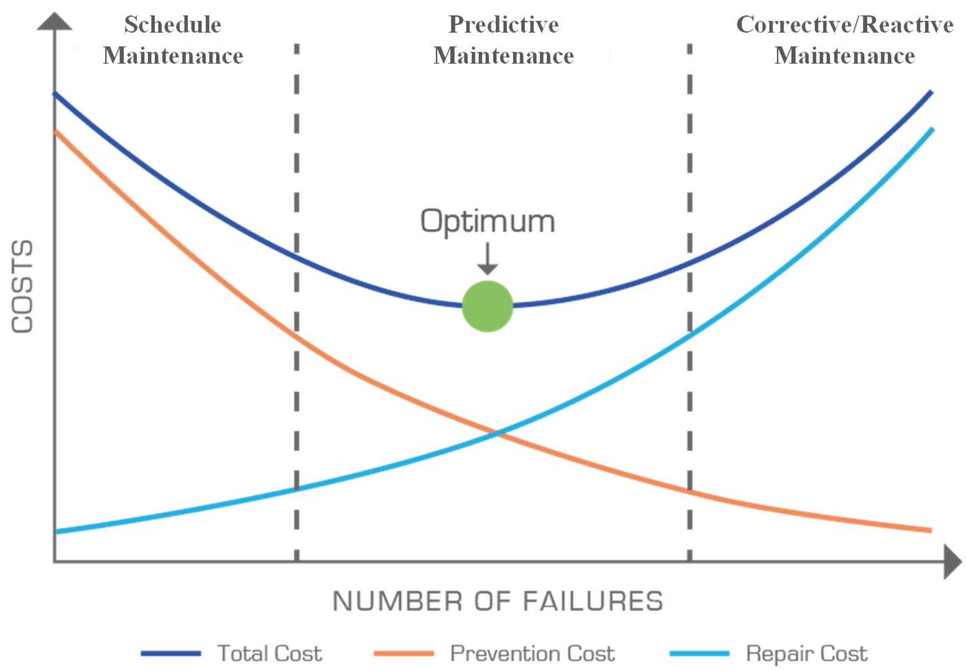


Figure 2.5: Maintenance costs vs number of failures (adapted [26])

However, the implementation of a Predictive Maintenance system in existing systems may present itself with high costs up front, due to the additional hardware and software investment, cost of manning, tooling, and education that is necessary to implement such a system. Nevertheless, it provides the organization with a basis for failure diagnostics and maintenance interventions, and supplies increased equipment reliability and enough information to improve planning and, consequently, decreases unexpected downtime and operation costs [2].

A brief description of the different types of maintenance is presented in Table 2.1.

Note that however accurate and reliable predictive techniques, such as CBM, RCM and other, may be the physics of failure are not yet fully understood and unpredicted failures will continue to occur. Therefore, predictive maintenance should not be seen as a substitute for the more traditional maintenance management methods but as a valuable addiction to a comprehensive maintenance program [27].

Table 2.1: Maintenance strategies (extracted from [13])

Category	Maintenance approaches			
Sub-category	Corrective	Preventive		
	Run-to-fail	Predetermined	Predictive	
	Fix when it breaks	Scheduled maintenance	Condition based maintenance diagnosis	Condition based maintenance prognostics
When scheduled	No scheduled maintenance	Maintenance based on a fixed time schedule for inspect, repair and overhaul	Maintenance based on current condition	Maintenance based on forecasting of remaining equipment life
Why scheduled	N/A	Intolerable failure effect and possibility of preventing the failure effect	Maintenance scheduled based on evidence of needs	Maintenance need is projected as probable within mission time
How scheduled	N/A	Based on the useful life of the component forecasted during design and updated through experience	Continuous collection of condition monitoring data	Forecasting of remaining equipment life based an actual stress loading
Kind of prediction	None	None	On and off-line, near real-time trend analysis	On and off-line, real-time trend analysis

2.2 New Trends in Maintenance

Maintenance strategies, such as schedule maintenance, may be inefficient in the sense of costs as well as in monitoring the lifetime of components and assets. Due to these reasons, the recent approach has been towards flexible maintenance policies in order to be able to take advantage of the information obtained through condition monitoring and carry out maintenance based on the needs and priorities [24].

The recent trends in maintenance management are aligned with Industrie 4.0 initiative [14], which foresees a data-centric production vision, starting with data collection and its analysis, allowing to extract its added value. Emergent technologies like Machine to Machine (M2M) communication, developments in operational sensor technologies, combined with advances in information technologies, such as cloud-based platforms, big data and analytics, are envisioned to sustain this new approach by enabling the unused potential of equipment by providing real-time data on the operational state and performance levels [28].

This data-based approach is taking the concept of maintenance to the next level and soon "Maintenance-as-a-Service" will become a reality. Integration and solution providers are already moving accordingly, researching and presenting new methodologies and approaches for the maintenance solutions.

In the recent years many efforts have been done in order to implement predictive maintenance in the factories. Companies such as Siemens, Rockwell Automation, Schneider Electric and Quant Service offer outsourcing maintenance services, namely cyclical assessment of system condition with data measurement and diagnosis, data analysis for product life cycle monitoring and advice on replacements, maximizing plant service life, minimizing component wear and avoidance of unplanned production downtime and costs. Other example, the *Senseye* company [29] provides a system that gathers data from several sources, analyzes this data and sends a notification to a designated person every time a abnormality is detected or failure is predicted. This solution uses machine learning to perform condition monitoring and prognosis analysis. On the other hand, there is the Watchdog Agent-based Real-time Remote Machinery Prognostics and Health Management (R2MPHM) platform presented and detailed in [30]. The Watchdog Agent consists of embedded computational prognostic algorithms and a software toolbox for predicting degradation of devices and systems. The Watchdog Agent-based R2MPHM platform receives data, processes it and extracts features that allows to detect possible failure occurrence and supports the estimation of the remaining useful life. For this purpose, the Watchdog Agent Toolbox makes use of several techniques and algorithms, such as Neural Networks, Bayesian Belief Network, Fuzzy Logic Prediction.

Despite the use of data measurement and diagnosis, these approaches are still being managed in a restrict and localized manner. Companies do not have enough trust in the new maintenance approach to implement them in their business. Also, the usage of innovative ways (e.g. augmented reality) of performing maintenance tasks is still done at academic level or in well controlled environments. The use of augmented reality in industry can happen at different levels, for example augmented reality can be applied during the maintenance procedure phase, i.e. when the actual repair actions are taking place. Here, machine learning algorithms allied with augmented reality will have a symbiotic connection aiming to provide an intelligent, interactive, simple and effortless maintenance repair operation. In fact, machine learning algorithms will dynamically guide the maintenance repair technician while the virtual reality environment will provide a comfortable human-system interaction.

In order to increase the levels of confidence of companies in the new maintenance approaches there is the need for further studies of intelligent and innovative maintenance systems. Thus, in the next chapter a architecture for a Condition-based Maintenance system is presented.

Chapter 3

System Architecture

As mentioned previously, the system architecture for CBM should present specific modules such as those described in the previous section. The developed system architecture for the Maintenance 4.0 project integrates all the referred modules to create a functional system that allows the implementation of intelligent and predictive maintenance and the overall system takes advantage of a broad spectrum of technologies, such as IoT technologies, machine learning, expert systems, among others. The developed architecture is depicted in Figure 3.1.

The system functionality is initiated with the Data Collection module, where the data from several sources is collected and stored in a database. This database will feed the Off-line Data Analysis module, where advanced data analytics, machine learning and cloud technologies are used to perform the knowledge generation. The outputs of this module are the generation or adjustment of rules, procedures and facts, which will be used by the Dynamic Monitoring functional block.

The Dynamic Monitoring module is divided into two components, the Visualization and the Early Detection of Failures. The Visualization component allows to compare Key Performance Indicators (KPI) against the expected operational limits. In order to determine these operational limits the facts resultant from the Off-line Data Analysis are considered and the raw data is displayed in a graphic format to facilitate its interpretation. On the other hand, the Early Detection of Failures component processes the facts and rules through the use of an inference engine, and triggers a maintenance warning when an anomaly is detected in an earlier stage. Depending

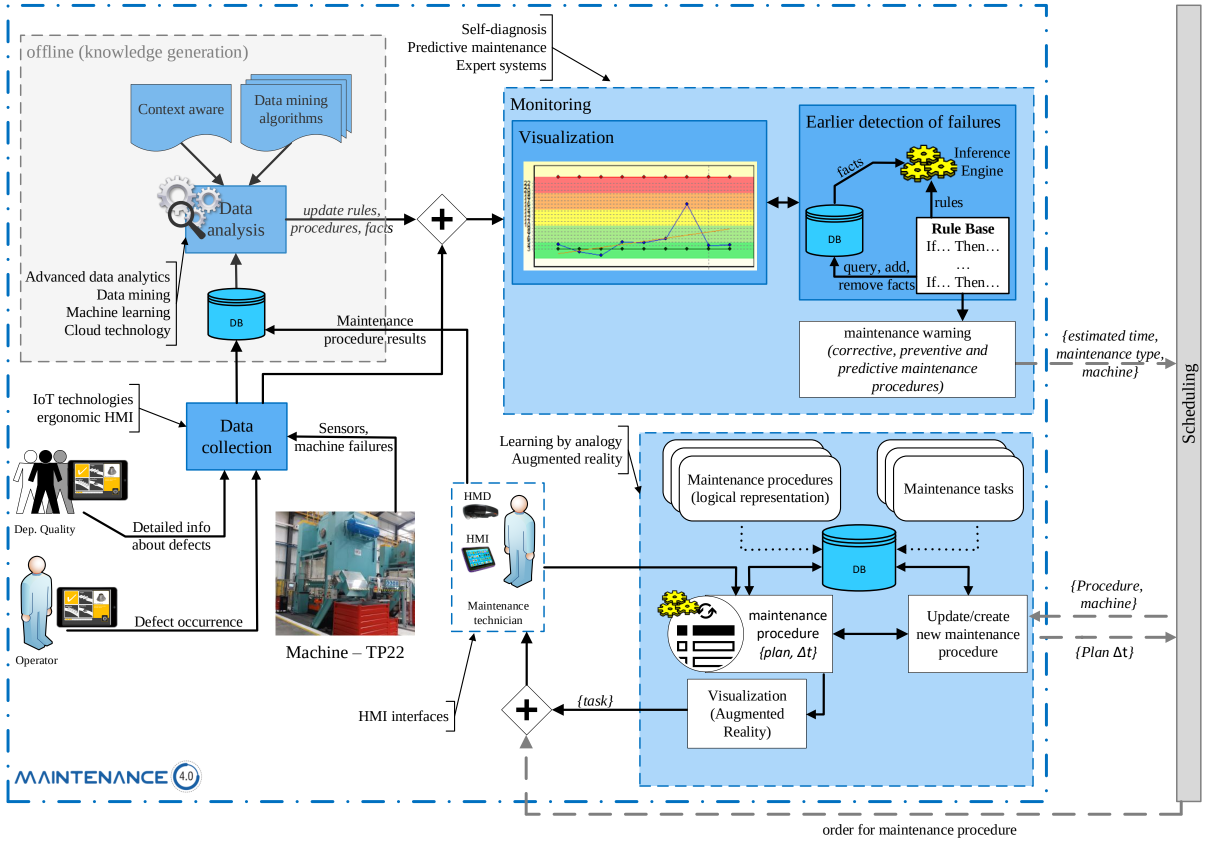


Figure 3.1: Intelligent and Predictive Maintenance System Architecture [31]

on the detected anomaly, the maintenance warnings can lead to different maintenance actions, namely corrective, preventive or predictive. Once the need for a maintenance intervention is detected, this information is sent to the scheduling tool, that will schedule the intervention according to the current production state and the maintenance resources availabilities. In spite of the importance of the scheduling system, this is out of scope of this work and consequently will not be detailed in this work.

The execution of scheduled maintenance interventions is guided and supported by a decision support system that selects the appropriate maintenance procedure and translates it into a language understandable by the human. The Intelligent Decision Support module is also able to adapt or create new maintenance procedures in cases that there are no known maintenance procedures for the detected anomaly. The maintenance procedure is provided to the maintenance technician, while performing the required maintenance actions, by using advanced Human-Machine Interface (HMI), e.g., head mounted devices.

Comparing the proposed system architecture with the OSA-CBM architecture, it is possible to verify that both present similar functional blocks. Figure 3.2 illustrates how the OSA-CBM architecture was adapted to the Maintenance 4.0 approach.

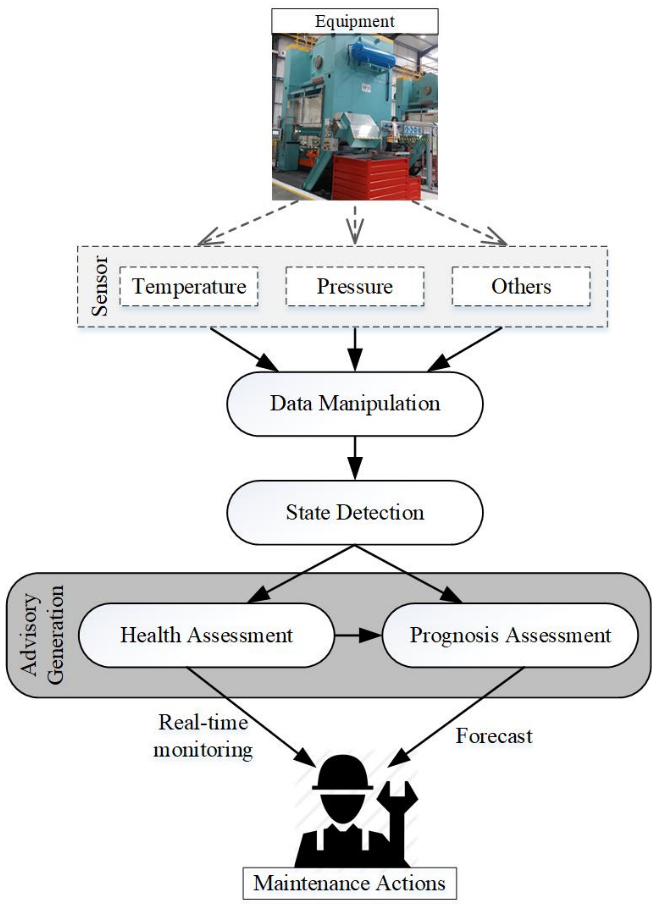


Figure 3.2: Adaptation of the interaction between the functional blocks of the OSA-CBM for the Maintenance 4.0 project.

The Data Collection module of the proposed architecture is equivalent to the Data Acquisition functional block of the OSA-CBM. Also, the Data Collection module fits, partially, within the Data Manipulation functional block of the OSA-CBM. The Off-line Data Analysis module is aligned with the Data Manipulation and the State Detection functional blocks since the knowledge needed to perform condition monitoring is generated in this module. The Dynamic

Monitoring of the proposed architecture matches the State Detection, Health Assessment and Prognosis Assessment functional blocks of the OSA-CBM architecture. The Decision Support module is completely aligned with the Advisory Generation functional block.

3.1 Data Colection

The Data Collection module considers different ways to collect data, namely automated, semi-automated and manual data collection. The data collection is considered automatic when is automatically acquired and stored in the database that will feed the rest of the system, semi-automatic when the information is automatically recorded in a database but has to be manually transferred to the system's database. Finally, the manual data collection occurs through the use of interfaces, where the information is manually inserted by a worker. Thus, this module can receive information from multiples sources, i.e. from several machines or assets, or from several departments or services. The collection of the required data is performed considering the IoT technologies and the ergonomic standards of HMI.

The design principles, to create user interfaces for data collection in industrial environments, should always take into consideration relevant features, such as the information presentation and the type of interface. The overall effects of the HMI appliance could not be totally predictable or even measurable since they do not depend only on the system design. The design goals and the consequent application should take into account the hierarchy of needs, such as 1) substantial procedures and advices, 2) continuous performance efficiency checklist, 3) ratings that come from data assessments, 4) specifications concerning reliability and validity, and 5) usability and efficacy of the system.

The HMI should be designed to involve users in the definition of the display and customization of the right assistance by applying the User Centered Design approach [32]. Also, the user interfaces must follow ergonomic guidelines established for the well-design of software and hardware systems.

Some examples of the parameters that can be collected to generate knowledge are described in Table 3.1.

Table 3.1: Example of collected data parameters.

Parameter	Description
Type of defect	Type of defect detected while the visual inspection is performed
Date	Date and time when the defect is detected
Part reference	Identification of the part where the defect is detected
Pressure	Pressure of the hydraulic piston during the operation
Temperature	Temperature in the surroundings of the equipment
Humidity	Humidity in the surroundings of the equipment
Vibration	Vibration of some components of the equipment
Operational noise	Characteristic noise produced during the operation of the equipment

The collected data will be used to feed the off-line data analysis and the visualization and dynamic monitoring modules.

3.2 Off-line Data Analysis

This module will take advantage of several technologies, namely advanced data analytics, machine learning and cloud technologies to extract knowledge from the collected data in order to create new monitoring rules and procedures or update the existing ones taking into consideration the correlation between different operational parameters.

For this purpose, a deep machine learning approach with supervised learning for the early fault prediction and predictive maintenance will be developed. A deep machine learning approach has the advantage of detecting underling patterns that may not be detected by a human operator/programmer [33], [34].

The evaluation of this machine learning approach will be tested on two types of Recurrent Neural Networks (RNN), the Long Short-Term Memory (LSTM) and the Gated Recurrent Unit (GRU) [35]. These networks are especially attractive to the predictive maintenance domain since they can learn from past sequences and forecast the next probable event.

This type of analysis is performed off-line and its results, i. e. the rules, applied in real-time in the dynamic monitoring module. With the implementation of these rules is intended to detect and generate alerts concerning possible failures of the systems normal functioning and additionally is intended to predict future health states of the assets.

3.3 Visualization and Dynamic Monitoring

The Visualization and Dynamic Monitoring module comprises two components, the Visualization and the Early Detection of Failures.

The Visualization receives inputs from the Off-line Data Analysis module that provides information about the normal functioning of assets operational limits, and data from the Data Collection module, which allows a real-time monitoring of specific variables. The real-time parameters will be graphically represented in a chart and when a parameter exits its tolerance interval, or the trend shows that it will eventually exit its operational interval, an alert is generated.

On the other hand, the Early Detection of Failures will be performed by an inference engine that uses a set of rules to match the existing facts, which can be retrieved from the database where the collected data is being stored. The rules are generated in the Off-line Data Analysis module through the detection of the correlation between different parameters. The rules follow a simple structure based on the `If Condition then Action` syntax, which are processed by the inference engine. When a rule is fired, a correspondent action is triggered, namely a warning for the need of maintenance interventions. When the need for maintenance interventions is detected, the information is sent to the Decision Support module through the scheduling tool.

3.4 Intelligent Decision Support for Maintenance

An important piece in this intelligent and predictive maintenance architecture is the decision support system for maintenance technicians during the execution of maintenance interventions.

This intelligent decision support system, articulated with human-machine interaction technologies, e.g. augmented reality, contributes for a faster and more efficient reaction and recovery of the failure occurrence when compared to paper procedures [36].

3.4.1 Intelligence of Decision Support Engine

The Intelligent Decision Support module is composed by a database, which can be shared by the database used in the other modules, an engine that processes the maintenance procedures, a tool that allows to update or create maintenance procedures and a human-machine visualization tool.

The database contains, amongst others, the maintenance procedures expressed in a formal logical representation, e.g., BPMN as the example depicted in Figure 3.3, which defines the sequences of single maintenance tasks. This database is connected to an inference engine that selects the proper maintenance procedure, and translates the procedure into a language understandable by the human that will be applied in the maintenance intervention.

For each case, if the necessary maintenance intervention does not have a correspondent maintenance procedure in the database, the engine enriched with machine learning, and particularly learning by analogy algorithms, will attempt to adapt or create a new maintenance procedure. This means that the engine will search for maintenance procedures applied to failures with similar features and show it to the operator.

3.4.2 Augmented Reality to Support Interactive Maintenance Operations

Augmented Reality (AR) is not a new technology, but instead it has been an active research area since almost three decades. AR is an interesting technology as it enhances the user's interaction and the perception of the real world by supplementing it with virtual things that coexist in the same space as the real ones. Therefore, AR supplements reality, rather than replacing it [37], [38]. Recent interest is being driven by enhancements of graphics capabilities, in particular in mobile and wearable devices, and in the plentifulness of wide assortment of sensors and, as well as, the support of advances on tracking combined with the availability of affordable AR

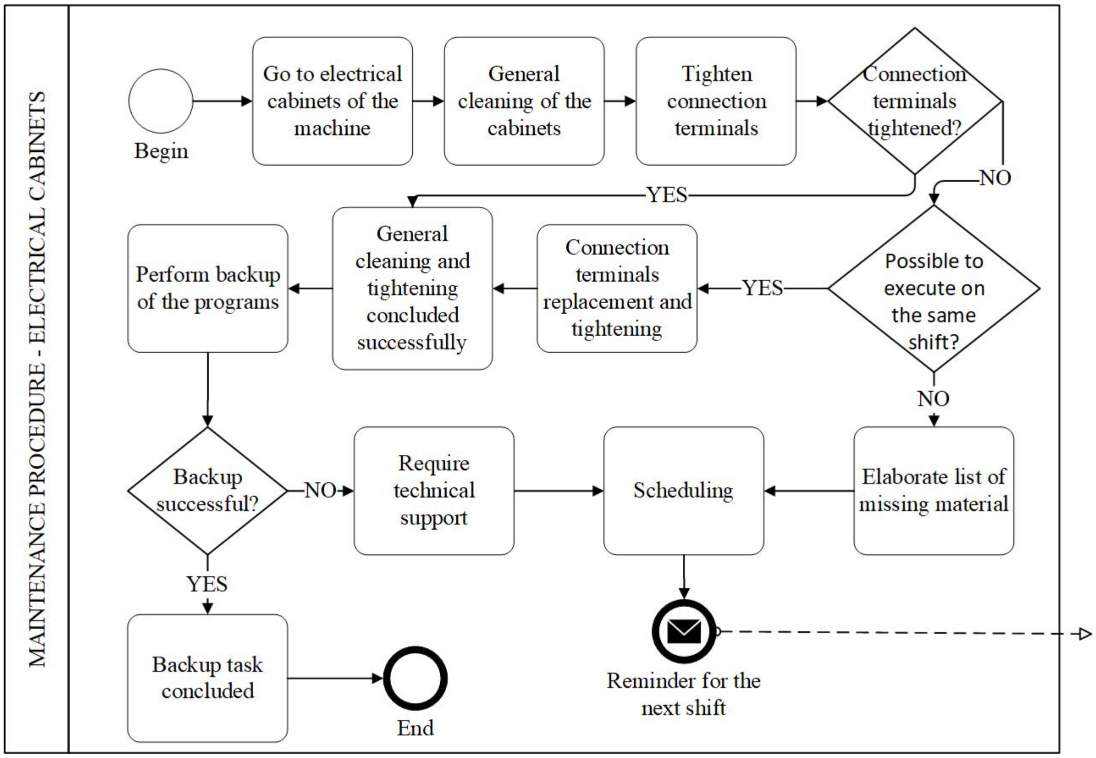


Figure 3.3: Example of a maintenance procedure expressed in BPMN format.

enabled hardware, such as Microsoft HoloLens [39], DAQRI smart glasses [40] or Atheer AiR Glasses, made AR an emergent topic of applied research.

All of this is also supported by the growth in the number of things with a virtual counterpart. Information (3D models and animations, real time information, documentation, etc.) regarding every thing with a virtual/digital existence, feeds applications that can supplement the user with a multitude of relevant, up-to-date and contextualized knowledge. Thus, AR technologies are interesting to many sectors, including industrial applications, from design through maintenance processes.

Since its origin, several AR applications have been envisioned and conceived to the manufacturing sector, such as assembly, maintenance, and repair of machinery [37], [38], [41], [42]. There have been several systems demonstrating the ability and usefulness of AR to act as an instruction and guidance tool [43]. Recently, the AR adoption by key players on industrial innovation, such as Boeing and General Electrics, resulted in the following main benefits: improved productivity, higher product and process quality (decreasing error rates), and better ergonomics.

This augmented reality component is being implemented with a hololens head mounted display and a tablet display. In both cases, the operator sees the real scene (the machinery) and sees the maintenance operation augmented over the real scene. The first offers an immersive experience where the interaction is done with hand movements to make selections. The second follows a traditional approach through the tablet display and the interaction is achieved through touch. The task is entrusted to a operator that follows the instructions presented on the AR display. In a first approach the system will be aimed for the operator training.

Each maintenance operation has several steps to follow in order to successfully achieve the task. The augmented reality process is supported by text, audio and 3D models and animations. The text describes the operation step together with an audio description which can be controlled during the process. The 3D models and animations represent the machinery and explains the operation with a 3D animation of the process to be done. It may assume that pieces are involved in the maintenance process, in which case, they are identified by its 3D model and factory reference.

At each maintenance step, the operator must confirm its completion before continuing, by interacting with the display interface. The interface includes buttons to confirm steps, to retrieve steps, to cancel the whole operation or to open a communication channel with a senior operator. The operator may be assisted by a more experienced operator during the maintenance process. For such purpose, the operator may start a video or audio conference with an assistant to make any questions concerning doubts or problems one may have during the maintenance process.

Chapter 4

Case Study Scenario

In this chapter it is presented a brief description of the organization where the case study is inserted, designated Catraport, Lda. It is also described the equipment under study, as well as the most common failures.

Catraport, Lda., from now on designated Catraport (Figure 4.1), it is a company founded in July 21 of 2015, in Portugal, with the intention of produce components and accessories for the automobile industry, through processes of cold industrial stamping. Catraport belongs to the entity CATRA SPA which has as main activity cold stamping, plastic stamping, construction of tools and molds, welding, assembly and painting of components.

Figure 4.1: Front entrance of Catraport facilities.

The company operates under the management philosophy Just in Time, applying a rigorous policy of quality reflected on the quality, management and environmental certificates. The group have a supplier-customer relationship with the company FAURECIA - Systems of Escape Portugal Lda., producing components that are sold to this entity, that later does the assembly of the pieces. Because of the growth of this company needs, associated to the interest that the group has in the Iberian market, CATRA SPA decided to invest on the creation of a company in Portugal. The factory shop floor is depicted in Figure 4.2 and is divided in different areas including production, maintenance, packing and storage areas.

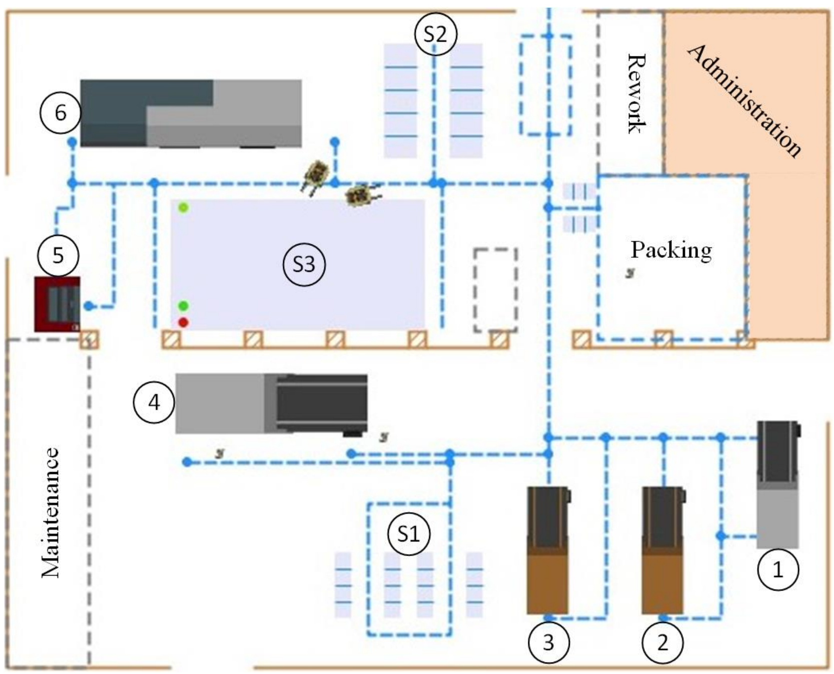


Figure 4.2: Catraport shop floor layout.

Briefly, the primary product is received as a metal sheet coil and is stored in storage area S1. These coils are then cut into metal disks of different diameters and thickness in machine 1, according to the production needs. Posteriorly, the discs are transformed in finished parts through the use of the stamping machines. The transformation process used consists of cold metal stamping, a multi-stage process of deformation of a sheet of metal until the finished part is achieved. The transformation process can occur in machines 2, 3 and 4, and for some parts the process is finished in machine 5. After this transformation the parts are placed in the storage area S2 waiting to be washed, in machine 6, due to the existence of lubricant in the parts. Once the this process is finished the parts are stored in storage area S3, where they stay until they are packed to be delivered to the customer. In the shop floor there are also the *Maintenance* area where all maintenance tasks on parts or stamping tools are performed. The *Rework* area is dedicated to the repair of defective parts. The blue tracks, in Figure 4.2, represent the path available for the forklifts to operate.

4.1 Equipment under Study

The equipment considered as case study consists of a metal stamping unit installed in Catraport facilities, illustrated in Figure 4.3, which is composed by several components/sub-systems that must be synchronized to operate correctly. The metal stamping unit under study presents itself with 400 tons of force and operates within a range of 10 to 24 strikes per minute.

Figure 4.3: Metal stamping machine.

This equipment is constituted by a metallic discs storage system, feeding system, transfer system and the stamping tool. Figure 4.4 depicts how the different system of the stamping machine interact and the respective sequence of transformation of the parts.

In a generic way, the metal stamping unit is composed by a system that allows the storage

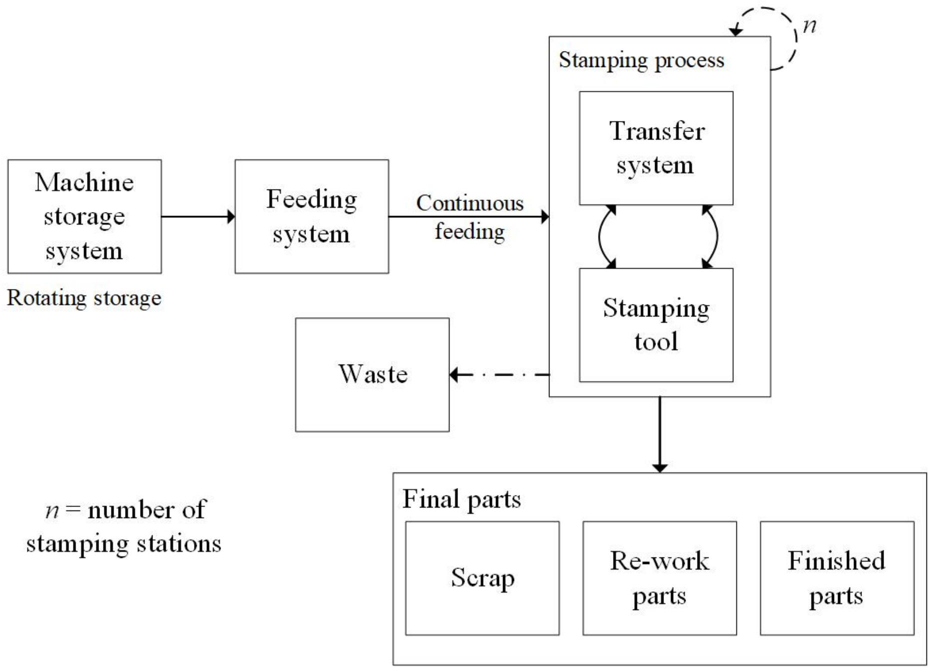


Figure 4.4: Sequence and interaction of the several systems of the stamping machine.

of metal discs which is proceeded by another system that feeds these discs continuously to the remaining systems to be processed. In order to transform the metal discs into finished parts, the stamping tool and the system which transfers the parts to the next position, from now on designated as *transfer*, operate simultaneously. The *transfer* system places the parts in the stations and when the parts are released the stamping tool presses these parts in order to gradually deform them into the final part. Then the *transfer* system takes the parts and moves them to the next station. This process is repeated n times, being n the number of station present in the stamping tool. At the time that the parts are being deformed there are excess material that is considered waste and is disposed automatically.

All the referred systems operate synchronously in order to obtain an automatic production line. In order to guarantee the synchronism between systems, alongside with several Sinamics motor controllers, a Siemens PLC S7 series is used.

Briefly, the production process for the presented machine is initiated in the storage system (depicted in Figure 4.5), where metallic discs, i. e. the raw parts, are stored. The storage system integrates four columns and it is responsible to place the discs in the correct position to be retrieved by the feeding system.

Figure 4.5: Storage system for the metal discs.

The storage system is equipped with sensors that indicate when the discs of a certain column are finishing. When this is detected the storage receives the order to rotate automatically in order to guarantee that the equipment continuous to have material to proceed with the production process. The metal discs are picked from the storage by the feeding system, which collects each metallic disc through the use of a suction grip, as depicted in Figure 4.6.

Figure 4.6: Feeding system for the metal discs.

Once the suction arm releases the metal disc on a conveyor, a sensor verifies if there is only one disc. If it detects the presence of more than one disc the machine stops in order to prevent posterior damages in the stamping tool.

The conveyor system that follows the arm, illustrated in Figure 4.7, transports the metallic discs to the first position of the stamping process and the portico placed over this conveyor is responsible for the lubrication of each disc.

Figure 4.7: Feeding system for the metal discs.

Once the disc is at the end of the conveyor, the transportation of the discs is handover to the *transfer* system (Figure 4.8).

Figure 4.8: Transfer system for the metal discs.

This system is responsible to transfer the parts through the successive stamping stations. The synchronism between the *transfer* system and the stamping tool is crucial for the proper execution of the stamping process. When these systems start to work asynchronously the machine stops the production.

After the stamping process is finished the parts are transported to the end of the production line by another conveyor. At the end of the production line an operator is responsible to execute visual inspection of each part produced. The parts that shows defects are separated from the rest and are considered waste or, if the defect is recoverable, the parts will be reworked.

Depending on the part that will be produced, different stamping tools can be coupled to the machine and depending on the complexity of the part the number of stamping stations may vary as well.

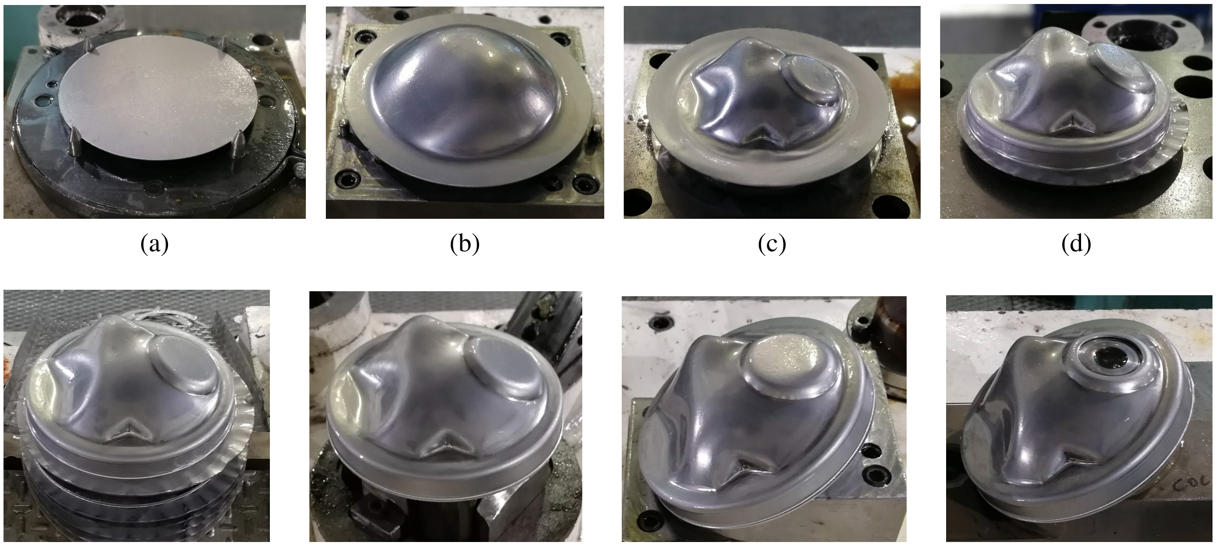
The part represented in Figure 4.9 is one of the several parts produced in the shop floor of Catraport and in order to obtain it, is necessary to use of a stamping tool with eight stamping stations.

Figure 4.9: Representation of a finished part.

In order to obtain the part represented above a metal disc is progressively deformed according to the parts depicted in Figure 4.10.

The number of stations of a metal stamping tool is defined according to the properties of the material and the geometry of the part.

This machine allows the production of several types of parts, in Figure 4.11 some examples

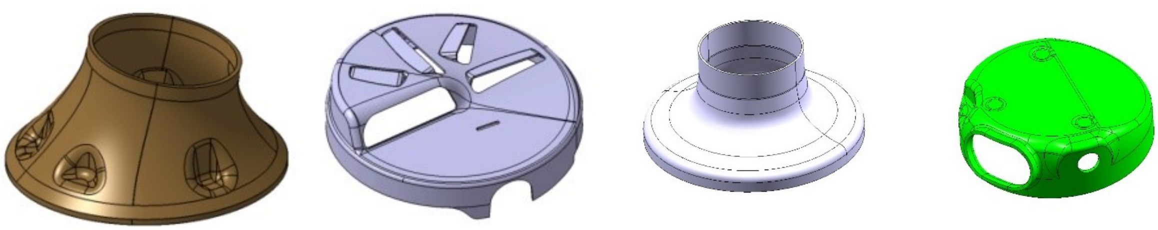


(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 4.10: Representation of the several stages of a part.

of the parts produced are depicted.



(a) (b) (c) (d)

Figure 4.11: Example of the produced parts.

The production of such parts generates data that should be collected and analyzed. The characterization of the available data is performed in the following subsection.

4.1.1 Available Data

In the shop floor of the presented case study there is a great quantity of data that is not being collected nor analyzed. After the characterization of the case study it was possible to identify the following critical types of data:

1. Data regarding warnings and stops of the machine: this data is stored in the memory of the machine and, if analyzed, may contain important information concerning the unknown failure patterns.

2. Data regarding the defective parts: the operator, at the end of the production line, performs a visual inspection of the part, but in case of defects the information about the type of defect is not registered in real time. However, this information may be useful to detect the degradation of the stamping tool or the related systems.

3. Data in analog format: data such as the hydraulic pressure of the piston is presented in analog format. The installation of additional sensors will allow to collect such data and analyze it.

4. Data regarding the conditions in the surrounding of the machine, such as temperature and humidity, and operational parameters, such as vibration and noise produced by the machine, should be collected in the attempt of find correlations with the failures occurred.

For the situation presented in item 1 the data is currently extracted from the machines memory manually. The data regarding the defective parts (item 2) can be collected through the use of interfaces design for the collection of specific data. Information characterized in items 3 and 4 can be transformed in digital data and automatically stored by the use of IoT technologies and communication protocols. The correct analysis of this data can help predict some of the most common failures.

4.1.2 Common Failures

Almost all failures are indicated in the displays of the stamping machine and result in the interruption of the productive process. Some of the common failures of the industrial case study are:

1. Failure in the storage system.

   This failure can be caused by the lack of metallic disc in the system or the storage system could not execute the rotation. The average time to recover is 5 to 10 min.

2. Failure caused by double disc.

   This failure is caused by the presence of more than one disc at the begging of the conveyor. The average time to recover is 2 min.

3. Failure due to nonexistent part in the station.

   This failure can be caused by the conveyor's velocity variation. The average time to recover is 3 min.

4. Malfunction of the storage system sensor.

   This failure can be caused by the lack of discs in storage or by the existence of malformed discs (wavy discs). The average time to recover is 5 min.

5. Signal of the encoder of the stamping machine did not work.

   The transfer system and the piston stop working synchronously.

All the presented failures result in interruptions of the productive process.

The monitoring of the parameters related to the failures aforementioned can lead to the early detection of these failures.

Chapter 5

Online Monitoring Module

The online condition monitoring is a very important part of an intelligent maintenance system since it allows to keep track of assets condition in real time. Furthermore, this can be exposed to the operators through the use of a visualization tool, which will facilitate the interpretation of information by displaying data and warnings. The designed online condition monitoring is a modular system. Figure 5.1 illustrates the organization of this module.

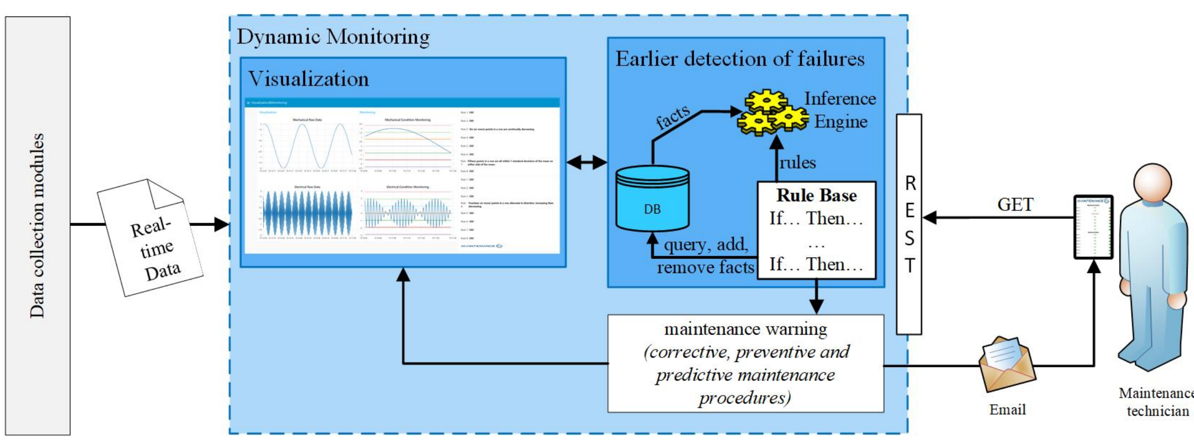


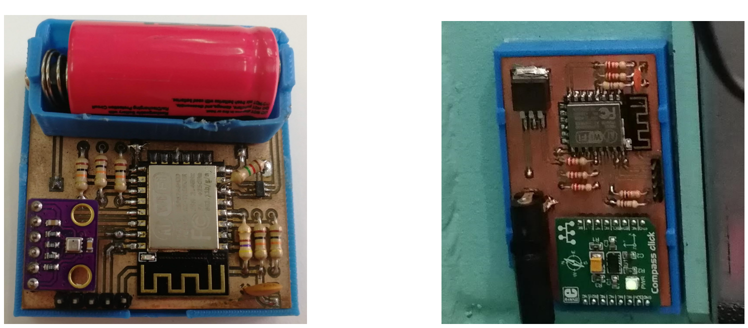
Figure 5.1: Online monitoring module.

As shown above, the online monitoring module receives real-time data from the equipment, which can be collected through the use of additional sensors installed in the equipment. The information that arrives to this module is shown in a visualization tool. On the other hand, the early detection of possible failure tool is used to implement the rules generated in the off-line data analysis module through the detection of the correlation between different parameters. Thus, the data generated in the shop floor is matched with the existing rules and if a rule is verified a maintenance warning is triggered. Once a rule is fired a warning for the need of a maintenance intervention is generated, which will send a notification in the visualization tool and will also send an email to an assigned person, e.g. the maintenance technician responsible for the equipment. To facilitate the verification of the assets' working condition the result of the rules is displayed on an Android mobile application.

The following sections present the work performed for the development of the online monitoring module, from the development of the data acquisition devices to the development of the tools that integrate the online monitoring module.

5.1 Data Acquisition

As mentioned in Chapter 4 there is data that is not being collect which can be acquired in a automatic way. An automatic data collection can be applied to several variables that may affect the production, namely temperature, humidity, atmospheric and/or hydraulic pressure and acceleration. The collection of such information is achieved by the implementation of additional modules capable of reading the desired values and send them to be stored and/or directly analyzed, such as the devices depicted in Figure 5.2.



(a) (b)

Figure 5.2: Representation of the sensing modules developed.

Figure 5.3 illustrates the electronic schematic of the module developed to collect the environmental data variables, such as temperature, humidity, atmospheric pressure and altitude. In order to acquire such data, of all the components used, two of them are highlighted, namely:

- ESP8266: a low-cost, wifi capable, microcontroller;
- BME280: an integrated environment sensor, being suitable for applications where consumption is a restriction (e.g., in battery powered situations).

To ensure the proper function of the two main components it was necessary to couple some hardware to them, as illustrated in Figure 5.3. The hardware is powered by a rechargeable battery (3.7 V and 2300mAh), which presents high autonomy due to the fact that must components are able to enter in a deep-sleep mode when are not being used. Thus, to improve the autonomy, a controlled switch was integrated in the powered circuit, allowing to temporary power-off all components, except the microcrontoller which is kept in deep-sleep mode when it's not transmitting data.

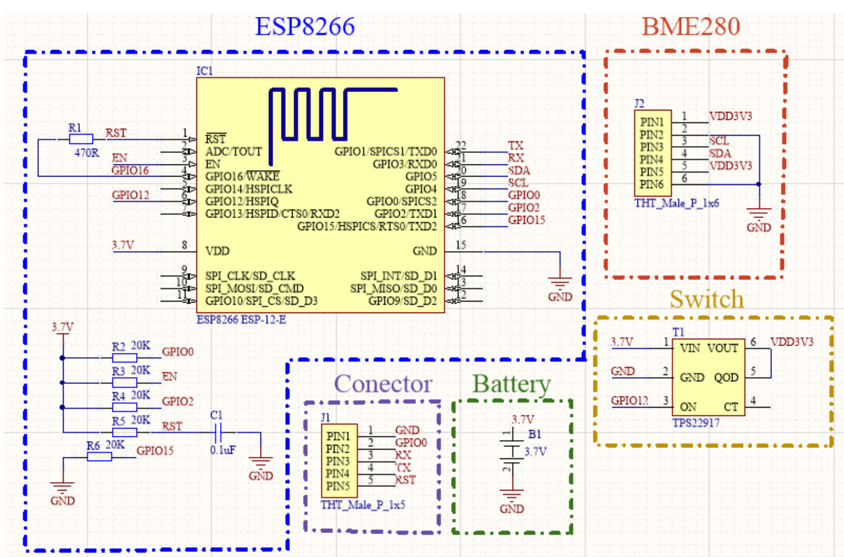


Figure 5.3: Electronic board schematic.

Given the fact that environmental parameters do not suffer significant changes in short periods of time it is possible to perform the parameters' readings in-between large periods of time.

The collected data is transmitted over wifi and following a JSON file format. The transmission protocol is the widely used Message Queuing Telemetry Transport (MQTT), which is a message protocol optimized for TCP/IP that allows messages to be easily handled, visualized on a Web platform and stored in a data base. Figure 5.4 presents a flow diagram that details the functioning of the developed module.

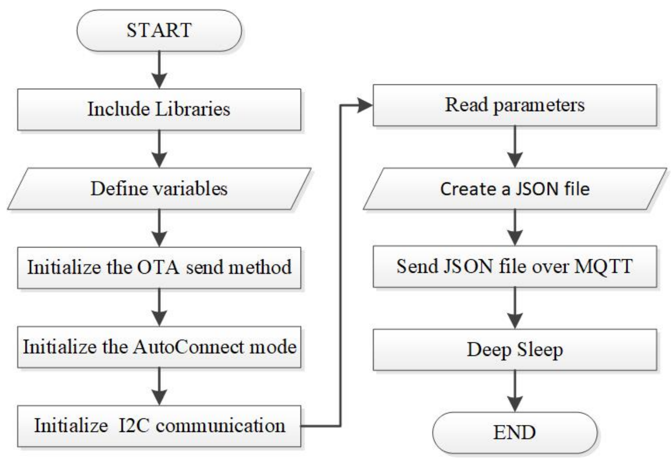


Figure 5.4: Environmental parameter acquisition fluxogram.

As it can be seen, the process starts by the microcontroller to load the necessary libraries, initiating the Inter-Integrated Circuit (I2C) communication protocol (need to acquire the data from the BME280), by formating the data in a JSON form and by transmitting this over wifi. After this, the controller shuts-down the power to all the devices entering in a deep-sleep mode afterwards.

Another important parameter to be collected is the machine vibration values. Figure 5.5 illustrates the schematic for the module that measures the vibration, acquiring the accelerations in three axis (X, Y and Z). For the construction of this module a ESP8266 and a *breakout* with an integrated accelerometer were used.

Given the fact that this module acquires the acceleration data, processes it and sends it (using a JSON format) in a continuous manner using MQTT, it has a high energy consumption. Therefore, this module in supplied by a 5VDC charger, which is later converted in 3.3V (nominal voltage of the module). The functioning of this module follows the abovementioned one with the exception of the power-off the board and the microcontroller entering in deep-sleep.

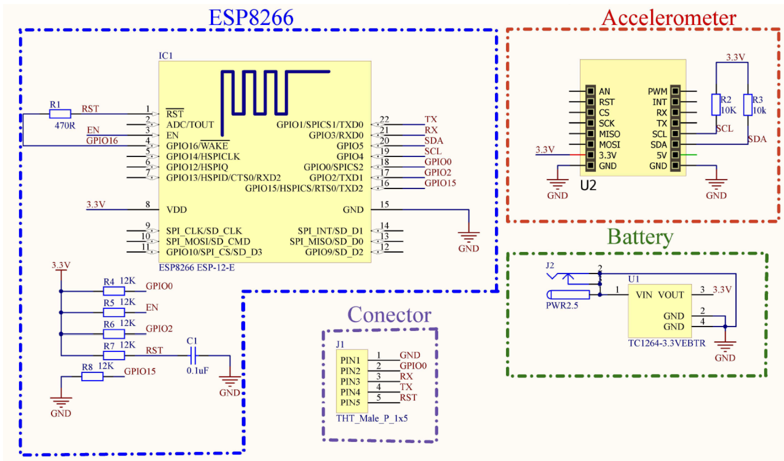


Figure 5.5: Vibration acquisition eletronic board schematic.

Figure 5.6 presents a flow diagram for the functioning of the module.

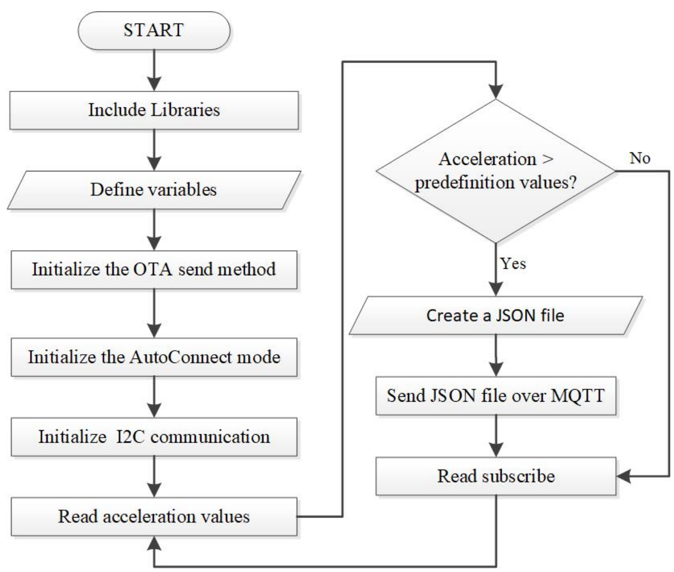


Figure 5.6: Vibration parameter acquisition fluxogram.

Figure 5.7 overviews an excerpt of the code for the creation of the JSON file and the transmission of data using the MQTT protocol.

From this code excerpt it is possible to denote that, besides the creation of the JSON string, the microcontroller uses a digital I/O, configured as output, to control the load switch regulator that powers-down the board electronics and that also, before getting into deep sleep mode, the microcontroller disconnects itself from the MQTT broker.

```
root["machine_id"] = "TP22";
root["Voltage"] = ESP.getVcc();
setupBME280();
root["Temperature"] = bme.readTemperature();
root["Humidity"] = bme.readHumidity();
root["Pressure"] =(bme.readPressure() / 100);
root["Altitude"] = bme.readAltitude(SEALEVELPRESSURE_HPA);
digitalWrite(Switch,LOW);
root.printTo(JSONmessageBuffer, root.measureLength() + 1);
client.publish(outBME280, JSONmessageBuffer);
yield();
client.disconnect();
```

Figure 5.7: Code excerpt for the creation of the JSON file and data transmission.

5.2 Online Monitoring Solution

The online monitoring solution consists of three tools, a visualization tool and the monitoring tool, both were developed using the platform Node-RED, and a mobile Android application developed using the MIT App Inventor platform. Since at the time of the development of this tool the data acquisition devices were not installed in the case study a different set of data and rules were used to validate the implementation. Nevertheless, a proof of concept is presented since this solution is modular, i.e. at any time the input data can be change if it follows the same data model (format). As for the rules, when the Off-line Data Analysis provides the specific rules for the case study they can also be easily implemented.

5.2.1 Nelson Rules

Due to the lack of rules regarding the case study of this work, the online monitoring solution was designed based on the Nelson rules, which were first published in the October 1984 issue of the Journal of Quality Technology in an article by Lloyd S. Nelson [44].

Nelson rules, evolved from the Western Electric Rules, consists of a method in process control of determining if some measured variable is out of control or presents a trend that shows that the variable will be out of control. The rules are applied to a control chart on which the magnitude of some variable is plotted against time. These rules are based on the mean value and the standard deviation of the samples. There are eight different Nelson rules [44], namely:

- Rule 1: One point is more than 3 standard deviations from the mean (outlier)

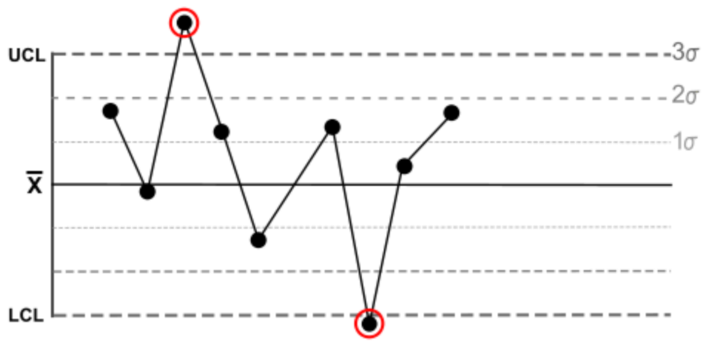


Figure 5.8: Nelson Rules - Rule1 [45].

- Rule 2: Nine (or more) points in a row are on the same side of the mean (shift)

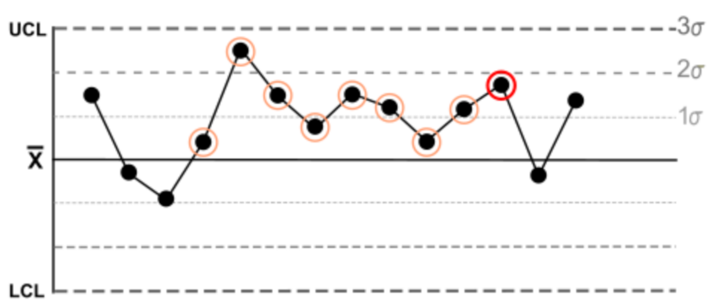


Figure 5.9: Nelson Rules - Rule2 [45].

- Rule 3: Six (or more) points in a row are continually increasing (or decreasing) (trend)

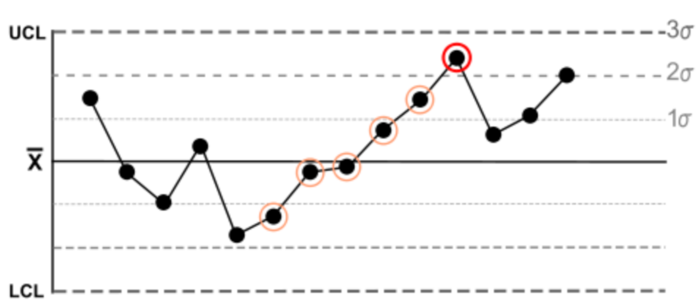


Figure 5.10: Nelson Rules - Rule3 [45].

- Rule 4: Fourteen (or more) points in a row alternate in direction, increasing then decreasing (bimodal, 2 or more factors in data set)

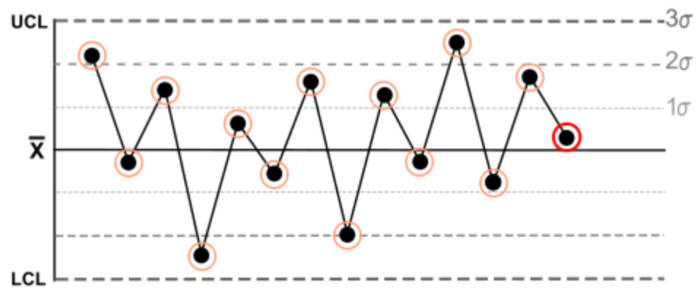


Figure 5.11: Nelson Rules - Rule4 [45].

- Rule 5: Two (or three) out of three points in a row are more than 2 standard deviations from the mean in the same direction (shift)

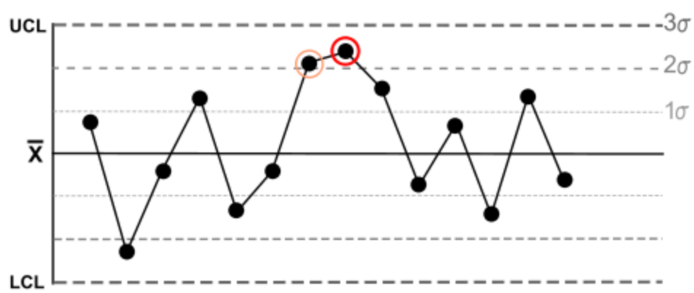


Figure 5.12: Nelson Rules - Rule5 [45].

- Rule 6: Four (or five) out of five points in a row are more than 1 standard deviation from the mean in the same direction (shift or trend)

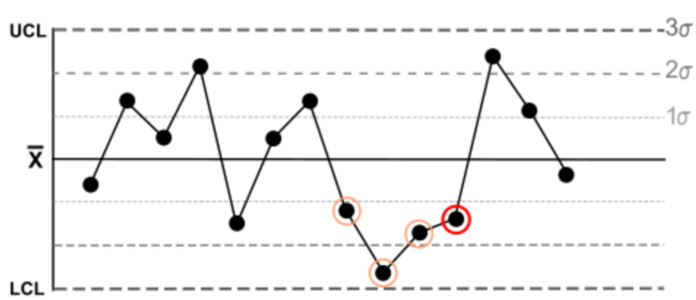


Figure 5.13: Nelson Rules - Rule6 [45].

- Rule 7: Fifteen points in a row are all within 1 standard deviation of the mean on either side of the mean (reduced variation or measurement issue)

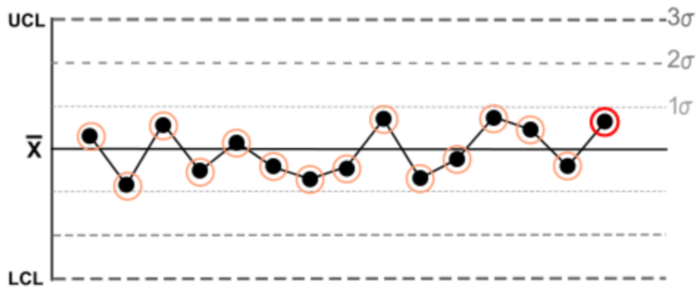


Figure 5.14: Nelson Rules - Rule7 [45].

- Rule 8: Eight points in a row exist with none within 1 standard deviation of the mean and the points are in both directions from the mean (bimodal, 2 or more factors in data set)

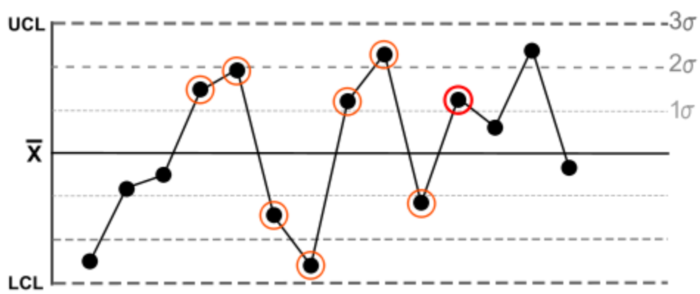


Figure 5.15: Nelson Rules - Rule8 [45].

Applying these rules indicates when a potential "out of control" situation has arisen. However, some false alerts may occur and the more rules applied the more will take place, meaning that for some processes it may be beneficial to omit one or more rules. On the other hand, there may be some missing alerts where some specific "out of control" situation is not detected. Empirically, the detection accuracy is good.

These rules were implemented in the Node-RED platform, which is a programming tool for wiring together hardware devices, APIs and online services [46].

5.2.2 Node-RED Solution

The Node-RED solution consists of a tool where real-time data is received and cross-checked with the rules implemented and, according to the rules results, generates warnings informing that the data is "out of control" or presents a trend that will lead to such state. The data and rules results are displayed in the UI. The code that controls this tool, illustrated in Figure 5.16, was developed using the platform Node-RED and is represented as a flow of several programmable blocks wired together. In this flow there are some specific nodes that allow the creation of the visualization tool.

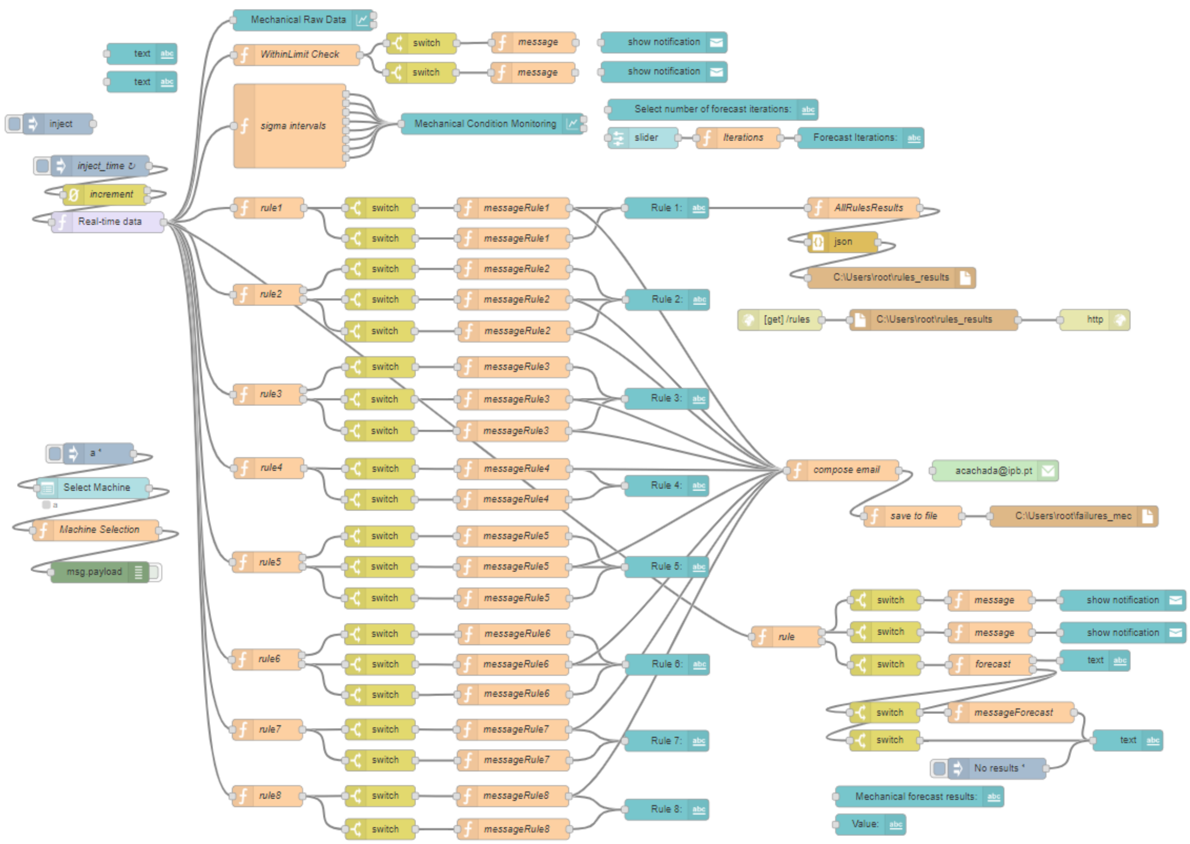


Figure 5.16: Node-RED flow for the implementation of rules.

The flow represented in Figure 5.16 can be decomposed into blocks that concern different functionalities. The initial part of the flow, and fundamental to execution of the tool, concerns the generation of data (depicted in Figure 5.17).

The block on the top injects a message, namely the time stamp, every 1 seconds. This message when injected in the *increment* block will generate on the output port a value, which starts on 0 (zero) and is continuously incremented. The wiring of these blocks to the mathematical function block *Real-time data* creates a defined mathematical function. In this, case there are

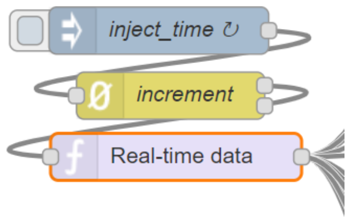


Figure 5.17: Node-RED blocks for data generation.

two data sets being generated, two sinusoidal functions, one with a smooth progress and another that presents strong variations. These data sets mimics the data that will latter be the input data of this solution and also presents a controlled and well known behaviour which allows to verify if the rules implemented are working properly. This data is then injected into the several blocks that process the data generated.

As mentioned before, the rules implement are applied to a control chart and are based on the calculation of the average and standard deviation of the input data, therefore the block *sigma intervals*, depicted in Figure 5.18 is where the these parameters are defined.

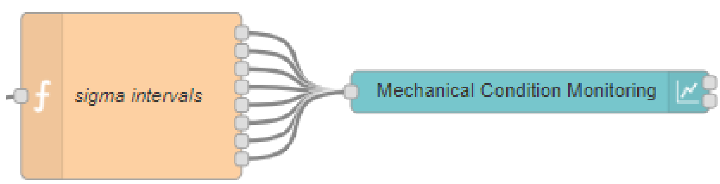


Figure 5.18: Node-RED blocks for the calculation of average and standard deviation.

The function block (in orange), allows to write a program to execute specific tasks. If the block is used as default will only pass through the message object that is receiving. For this case the block *sigma intervals* is responsible for the definition of the average and standard deviation values.

The *Mechanical Condition Monitoring* block allows for the raw data, the average and the intervals of +/- one, two and three standard deviations to be visualized on the UI. The visualization of different sets of data in the same graphic is only possible by the definition of a different topic property to each one.

The implementation of the monitoring rules can follow two simple structures according to the expected output. For rules that return only `true`, if the rules if triggered, or `false`, if the rule is not verified, the used structure consists of a function block, where the rule is programmed, two switch blocks, each one only lets through a specific type of messages, followed by function blocks which contains a message that will be send to a text block which will print the message in the UI. This was the structure used for the implementation of the Nelson rules 1, 4, 7 and 8 and the refereed structure is depicted in 5.19.

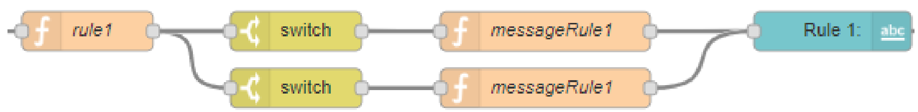


Figure 5.19: Node-RED blocks for the implementation of Nelson rule's Rule 1.

In the second structure that was used, illustrated in Figure 5.20, the function block where the rule is programmed presents two outputs due to the fact that for some rules, namely 2, 3, 5 and 6, when the rule is triggered there are two possible outputs.

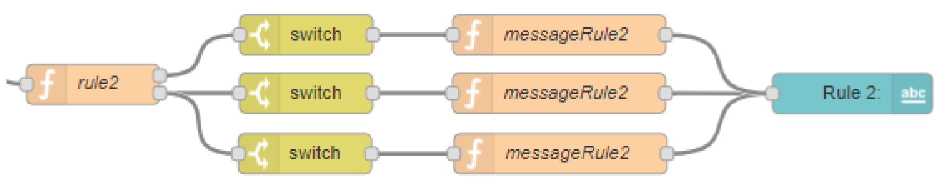


Figure 5.20: Node-RED blocks for the implementation of Nelson rule's Rule 2.

Besides the implementation of the Nelson Rules, this tool presents the implementation of other rules and functionalities. An important functionality of this tool is the capacity to execute a simple forecast. The group of blocks that allow to detect the need for forecast are illustrated in Figure 5.21.

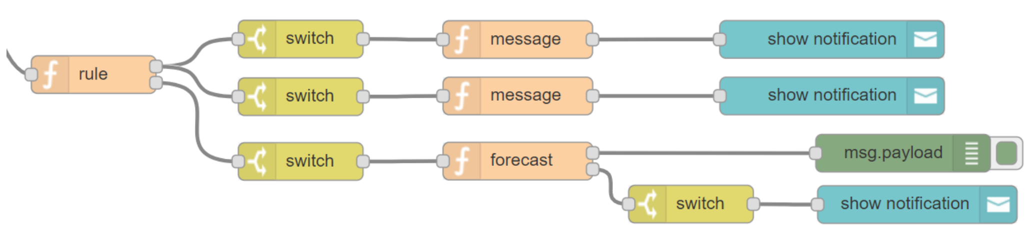


Figure 5.21: Node-RED blocks for implementing the forecast functionality.

The first block verifies if there are at least three points higher than +/- one standard deviation, if this proves to be true the rule will check if the values are rising, if so the rule returns `true` and a notification alerting for this fact appears in the UI. On the other hand, if there are at least three consecutive data points higher than +/- one standard deviation and their values are decreasing a notification informing that the values are returning to their normal operational limit is also shown in the UI.

Once the rule detects that there are three consecutive data points beyond +/- one standard deviation, and their values are rising, the variable that controls the performing of the forecast is set as `true` and the function block *forecast* is executed. In this function the average value of the difference between each of the three data points is calculated and this value is then added to the last data point as many times as the user, which can be the operator, the maintenance technician or other, defines. Then it is verified if the resulting value is higher than a defined interval or threshold, in this case is verified if the value belongs within the +/- one and two standard deviations interval or if is higher than +/- two standard deviations, if so notifications such as *forecast within the limits -1 and -2 sigma* and/or *forecast exceed the 2 sigma limit* are displayed in the UI. The number of iterations for the forecast is defined in the UI and the set of blocks that allow it is depicted in Figure 5.22.

Figure 5.22: Node-RED blocks for the definition of the number of forecast iterations.

The first block allows to setup the slider that appears in the UI. In this case the slider is defined with a minimum value of 0 (zero) and a maximum of 200, with a step of 1 unit. The value defined by the user is the input of the function block *Iterations*, where it is transformed in a global variable meaning that it can be used in any other block of the flow. This value is also displayed in the UI so that the user is aware of the selected value.

Another functionality presented by this solution is the sending of an alert by email to a designated person. This functionality is implemented by the use of a function block and an email block, as depicted in Figure 5.23

Figure 5.23: Node-RED blocks for implementing an automatic alert by email.

The *compose email* block is where the message to be sent is created, which is a variable message given the fact that it changes according to the machine that is in operation and the rule that has been triggered. In Figure 5.24 the code used to program the email to be sent is illustrated.

```
msg = {
    payload:"In machine " + context.global.machineName.payload + " was detected that " + context.global.whatRule.payload +
        " has been triggered at " + Date().toString(),
    topic: "Maintenance Needed on " + context.global.machineName.payload,
};

return msg;
```

Figure 5.24: Code used to program the *compose email* function block.

To send the maintenance warning by email is necessary to codify the body and the subject of the email. In order to have a variable message being created in a single function block is necessary to incorporate variables in the message. For this purpose it was needed to incorporate two variables, namely *context.global.machineName* and *context.global.whatRule*, that have assigned values elsewhere in the flow and when the message is built in the *compose email* block these variable will be replaced by their current values. The body of the email is constructed in the payload of the message and the *context.global.machineName* value concerns the identification of the machine and the variable *context.global.whatRule* pinpointing the rule that was triggered. The code excerpt *Date().toString()* incorporates the time stamp in the message of the email. On the other hand, the subject of the email is built in the topic of the message. The *msg* object (payload and topic) is the input of the email block and will be sent to the assigned addresses.

By the implementation of the blocks shown in Figure 5.23 together with the code in Figure 5.24, every time that a rule is triggered, the maintenance technician or another designated person, will receive an email informing of such occurrence. The email send by this application follow the structure of the email illustrated in Figure 5.25.

In order to save the historical data and to keep track of all events the blocks represented

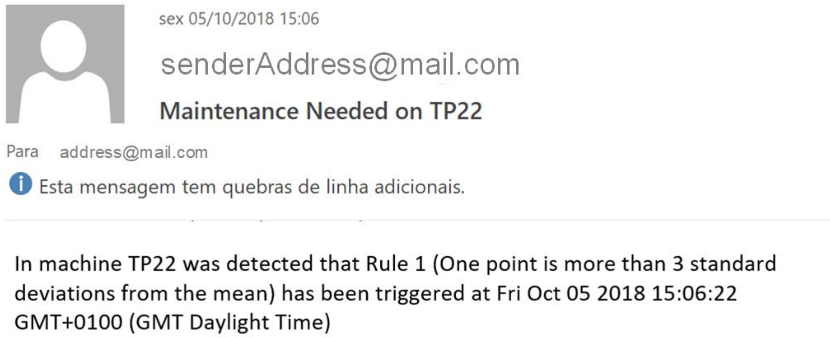


Figure 5.25: Structure of the email receive by the addressee.

in Figure 5.26 were implemented in the solution. These two blocks together allow to save structured information in a file.

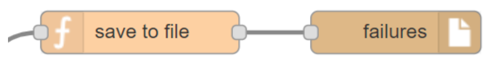


Figure 5.26: Node-RED blocks for implementing a file save functionality.

The function block *save to file* contains a code similar to the one found in Figure 5.24, which creates a message with information regarding the rule that was triggered, the machine and the time stamp. In case that the monitoring tool stops, when it returns to normal functioning the file is not replaced by another, new lines are added to each payload, ensuring that the historical data is not erased.

The blocks presented in Figure 5.27 save the rules results, i. e. `true` or `false`, in a JSON format. The results of all rules are first aggregated in a single message object and then, the message is injected in a *json* block which converts between a JSON string and its JavaScript object representation, in either direction. Given the fact that the input is a JavaScript object, this block returns a JSON string which is saved in a file.

Figure 5.27: Node-RED blocks to parse and save the rules' results.

The function block *AllRulesResults* saves the results of all rules, i.e. mechanical and electrical rules, in a single message through the execution of the code shown in Figure 5.28. Since the rules are saved in global context variables, their values can be accessed in any part of the flow.

```
msg.payload = {
        "Mechanical": [
        {"rule1": context.global.resultRule1},
        {"rule2": context.global.resultRule2},
        {"rule3": context.global.resultRule3},
        {"rule4": context.global.resultRule4},
        {"rule5": context.global.resultRule5},
        {"rule6": context.global.resultRule6},
        {"rule7": context.global.resultRule7},
        {"rule8": context.global.resultRule8},
        ],
        "Electrical": [
        {"rule1": context.global.resultRule1_elec},
        {"rule2": context.global.resultRule2_elec},
        {"rule3": context.global.resultRule3_elec},
        {"rule4": context.global.resultRule4_elec},
        {"rule5": context.global.resultRule5_elec},
        {"rule6": context.global.resultRule6_elec},
        {"rule7": context.global.resultRule7_elec},
        {"rule8": context.global.resultRule8_elec}
        ]
};
return msg;
```

Figure 5.28: Code excerpt to aggregate all rules results in a single JavaScript object.

The result of all rules can be retrieved by an external application using the method GET of the REST protocol. The sequence of blocks which allows this functionality is illustrated in Figure 5.29.

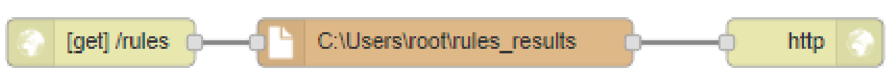


Figure 5.29: Node-RED blocks for implementing the method GET of the REST protocol.

The first block creates an HTTP end-point for creating web services, the information shown in the end-point is read from the file where the results of the rules are stored with the use of the second block in Figure 5.29. The last block send back the information, i. e. a JSON string, to requests received from an HTTP input node. The data send to the end-point is represented in the following format: "Mechanical": ["rule1": false , "rule2": false , "rule3": false , "rule4": false , "rule5": true , "rule6": false , "rule7": false , "rule8": false], "Electrical": ["rule1": false , "rule2": false , "rule3": false , "rule4": false , "rule5": false , "rule6": false , "rule7": false , "rule8": false]

The described functionality is used to send the results of all rules to an Android application which allows to visualize the status of the rules anywhere. The development of the referred application is detailed in the following subsection.

User Interface for the Node-RED Application

The flow represented in Figure 5.16 was constructed in a way that allows to create two user interfaces, one where the user can parameterize the tool and other that allows the visualization of the input data and the results of the rules implemented. The user interface for the parameterization of the tool, depicted in Figure 5.30, also contains the results of the forecast, namely message and value.

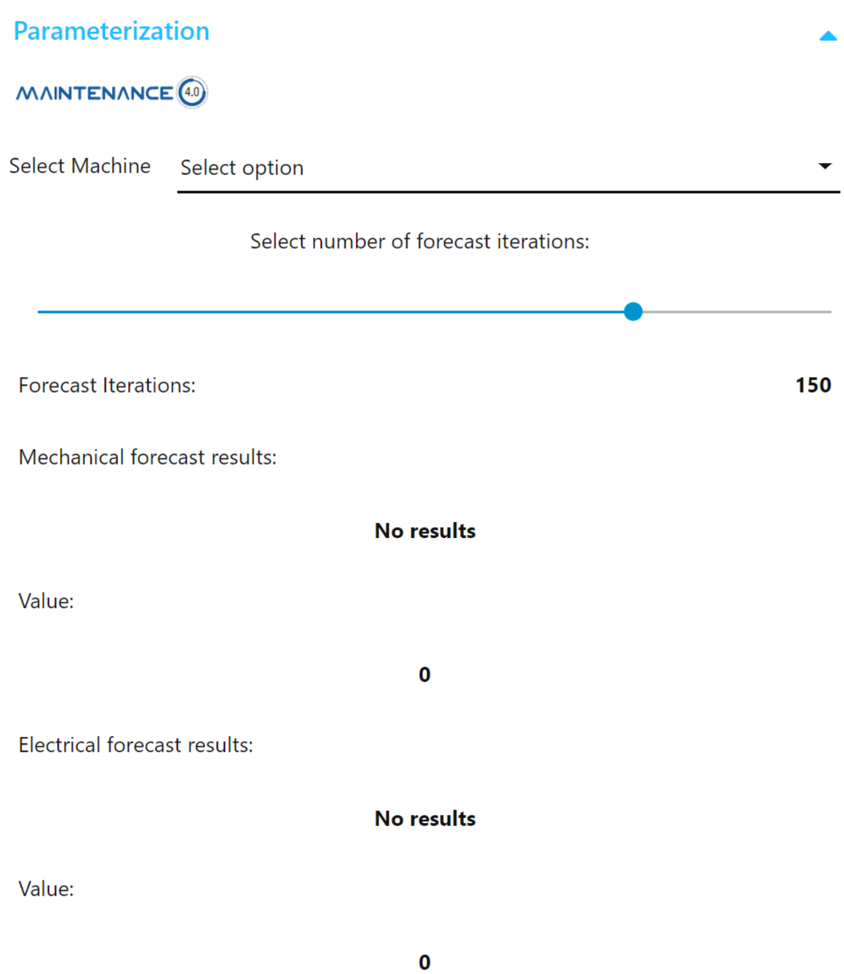


Figure 5.30: User interface for parameterization of the tool and visualization of forecast results.

The parameterization of the tool is executed by choosing the machine in operation and the number of iterations to perform forecast. Within the flow there are blocks to create another UI, depicted in Figure 5.31, which will allow the user to easily visualize and interpret all the information.

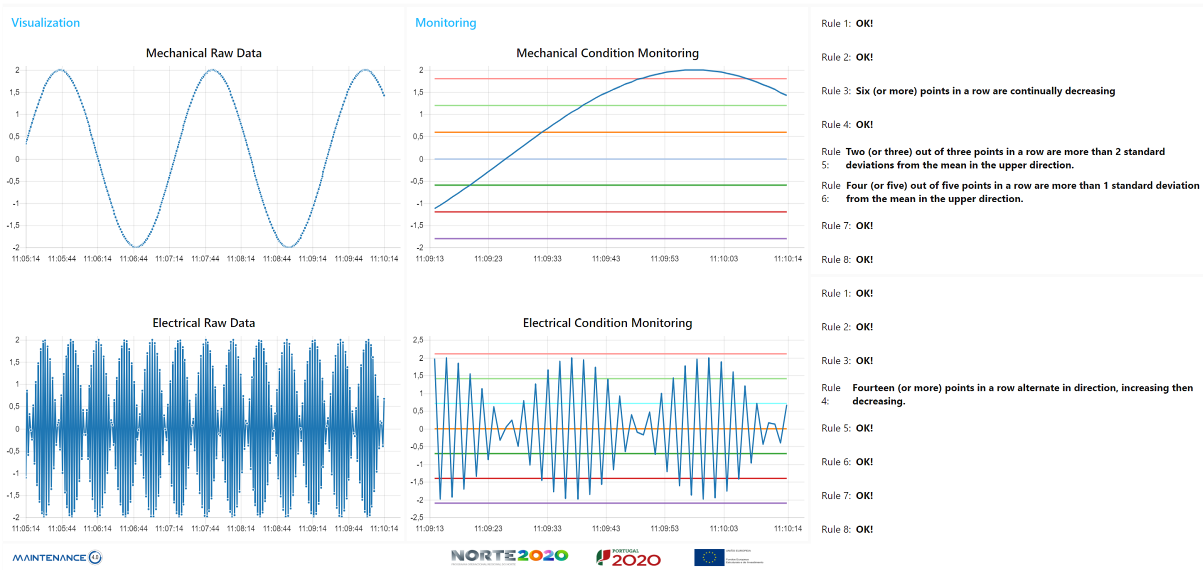


Figure 5.31: Node-RED dashboard for data visualization and monitoring.

This UI is divided in different areas, in area 1 is possible to graphically visualize the raw data, i. e. data is represented as it enters the tool, in area 2 the incoming data is represented with the respective average and +/- one, two and three standard deviations intervals and in area 3 is where the message indicating the state of the rules can be visualize. For example, in Figure 5.31 is possible to see that the rules 3, 5 and 6 for the mechanical data set and rule 4 for the electrical data set have been triggered and the others are OK. The UI for the parameterization and the UI for the visualization exists in different tabs.

5.2.3 Android Mobile Application Development

The purpose of developing an Android interface is to allow the maintenance technician, or another person, to be aware of the working state of the assets through the use of a clean and simple interface, which displays simple text and images that will facilitate the interpretation of the information presented. This mobile application will allow the maintenance technician to check the state of a given asset anywhere.

For the development of the Android application the platform MIT App Inventor was used. In Figure 5.32 is illustrated the full set of blocks used to program the application which are translated in a user interface.

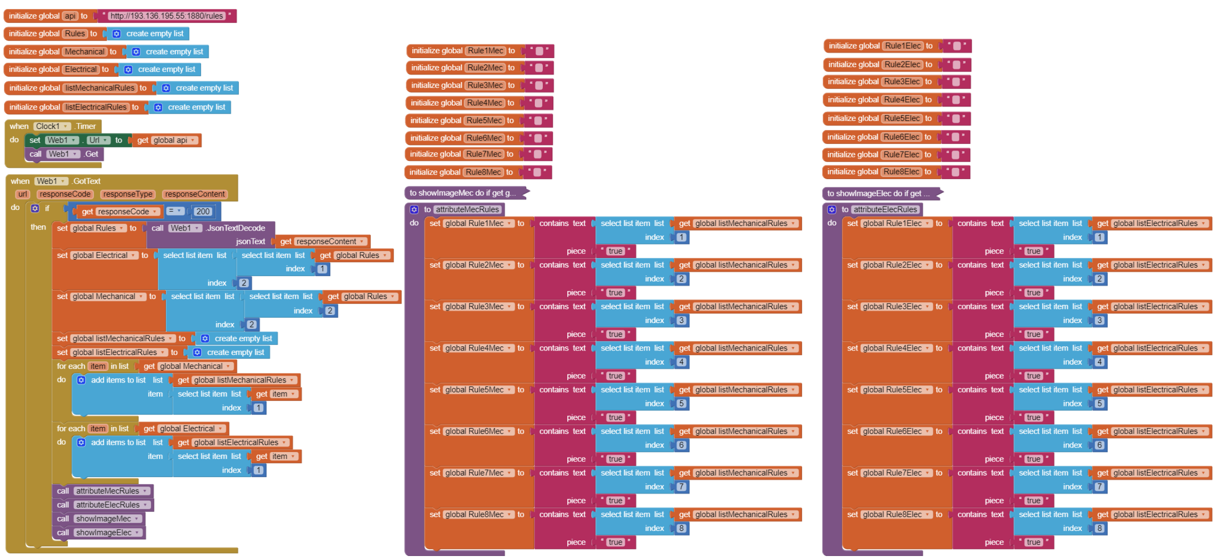


Figure 5.32: Environment of the application development tool.

Since the Node-RED application is prepared to receive a GET request, it is necessary to code such request on the MIT platform, which is executable through the use of the set of blocks illustrated in Figure 5.33.

Figure 5.33: Blocks to execute the GET request.

These blocks are responsible to, with a defined frequency, set the Url to the web server Url and execute a GET request. The variable *global api* contains the Url for the web server and is initialized with the other variables when the application is initiated. Once the Url is defined the application executes the GET request.

When the GET request is finished, the group of blocks represented in Figure 5.34 is executed. This group of blocks is responsible for the parse of the received JSON message. This is performed through the use of the *.JsonTextDecode* block, which transforms the JSON object in a list two indexes, i. e. the results for the mechanical and electrical rules, and each one contains items correspondent to each rule.

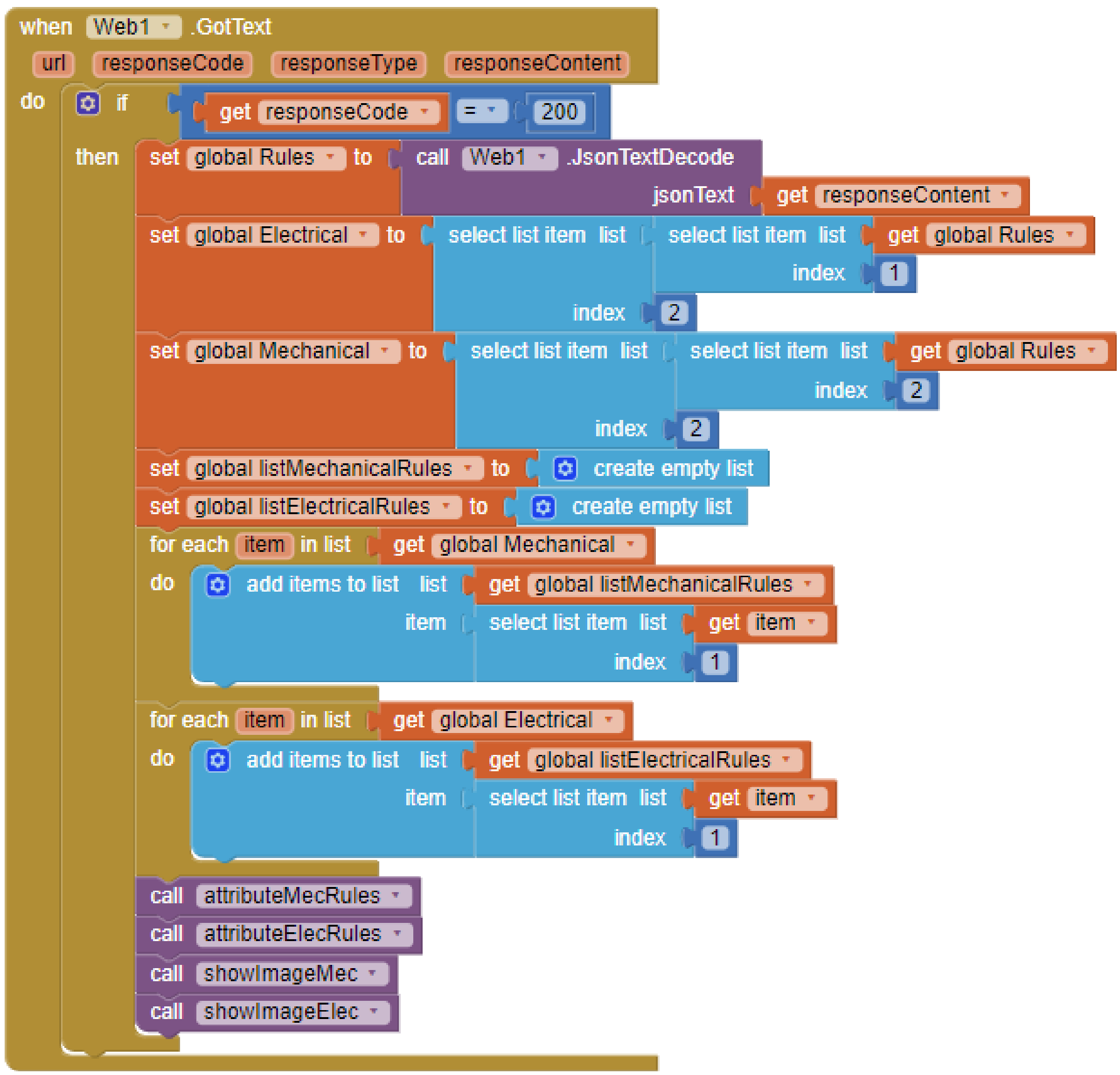


Figure 5.34: Blocks executed when the application receives message.

Once the different sets of rules results are separated in two different lists, namely *Electrical* and *Mechanical*, each item of these lists of results are saved on another variable where the result of each rule is stored in an individual item. Then two procedures are called, namely *attributeMecRules* and *attributeElecRules*, to execute the verification of the state of each rule, as shown in Figure 5.35.

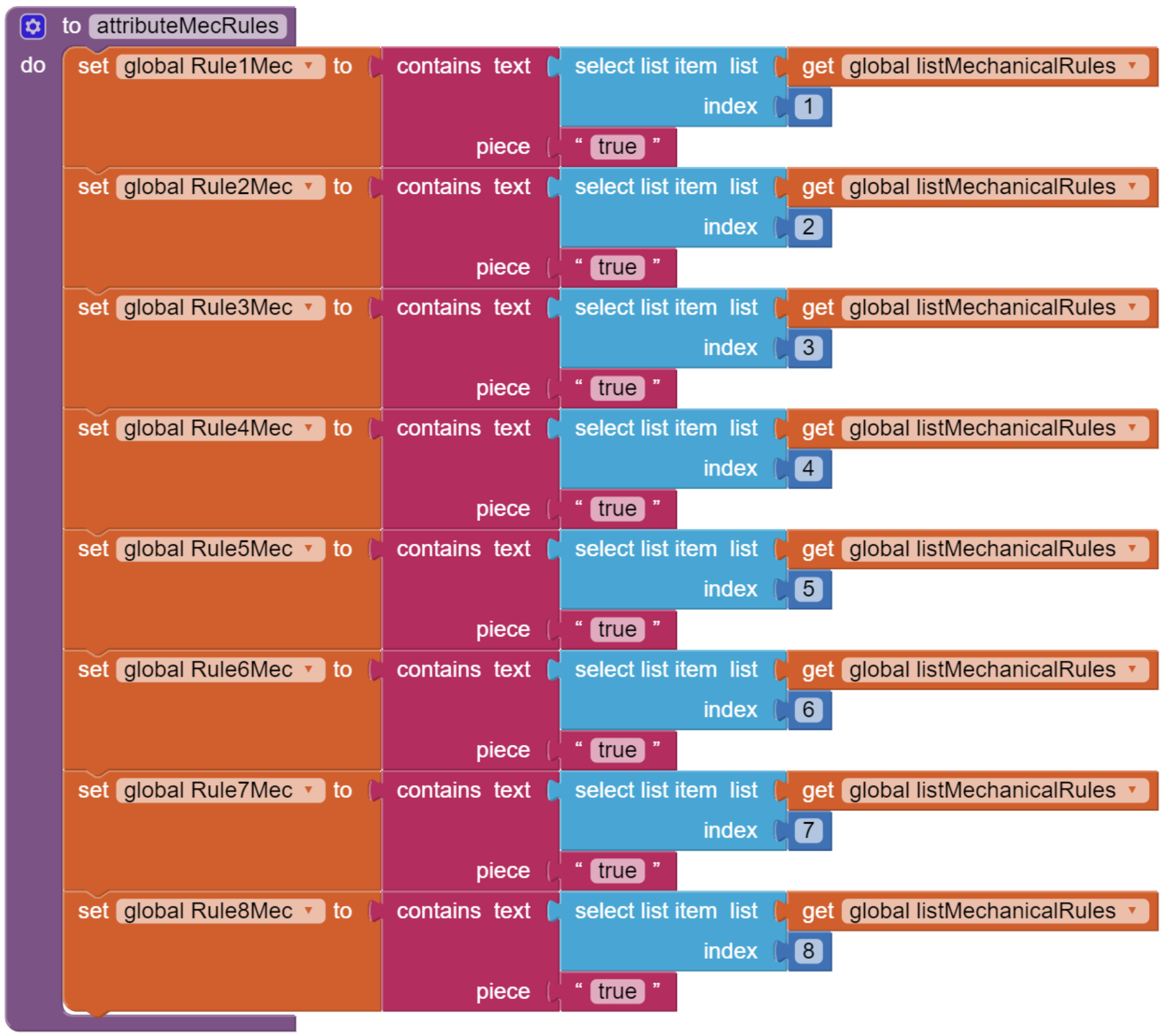


Figure 5.35: Procedure executed to verify the rules status.

For this procedure each item in the list of mechanical rules is verified and the value of each rule, `true` or `false`, is saved in independent variables. For the set of electrical rules a similar procedure is applied.

The last two procedures, *showImageMec* and *showImageElec*, are used to control the appearance of text and image according to the current rule result. For each rule, there are different attributes such as message, text color and image to show, that are defined. Figure 5.36 show the group of blocks that control the appearance of text and image for rule 1 of the mechanical rules. This set of blocks exists within the *showImageMec* procedure.

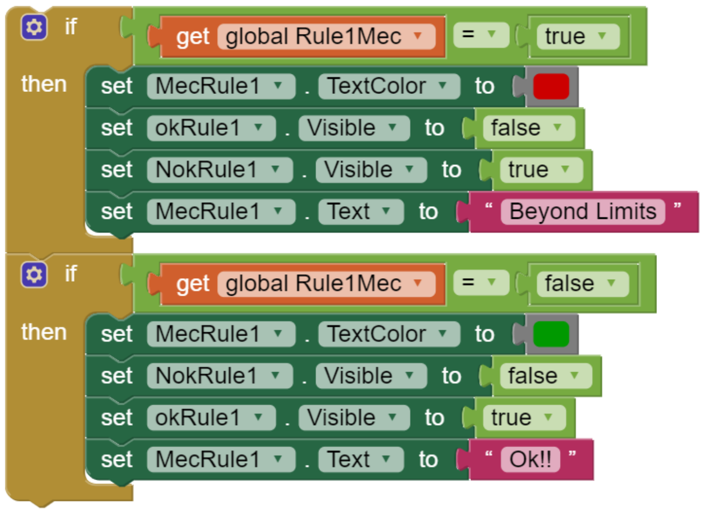


Figure 5.36: Blocks used to control the messages, text color and images in the interface.

In this set of blocks, if the rule is `true`, meaning that the rule has been triggered, the text color is set to red, the image used to show that the rule is OK is hidden, the NOK symbol is set as visible and the message for this situation is set for *"Beyond Limits"*. Otherwise, if the rule value is `false` the text color is set to green, the image used to show that the rule is OK is set as visible, the NOK symbol is hidden and the message for this situation is set for *"Ok!!"*. For the other rules there are similar groups of blocks that control the appearance of text and image in the UI, all the groups are place within the *showImageMec* and *showImageElec* procedures.

User Interface for the Android Mobile Application

These blocks all together are translated into a simple interface, depicted in Figure 5.37, that allows to keep track of the operating state of the asset. This interface should be simple and intuitive, therefore the information is presented with short text and supported by images and a color code that easily allow for the user to interpret the meaning of the message. The messages shown in the UI for each rule are those presented in 5.1.

Table 5.1: Messages presented in the UI.

Rule	**Message**
Rule 1	Beyond Limits
Rule 2	Prolonged bias exists
Rule 3	Trend exists
Rule 4	High oscillation exists
Rule 5	Small trend possibility
Rule 6	Strong trend possibility
Rule 7	Stratification
Rule 8	Over-control

Depending on the input data the messages can be customized according to its meaning. In Figure 5.37, due to the triggering of rule 3, the message *"Trend exists"* is visible.

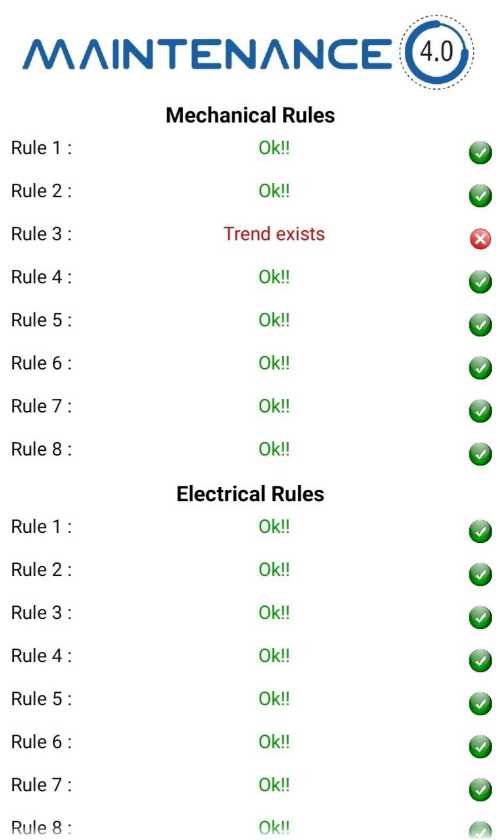


Figure 5.37: User interface of the Android application.

The color green is used when the rules have not been triggered to show the message *"Ok!!"* implying that the asset is operating within the proper limits. Additionally to this message there is an image  that is also displayed. The color red is used to display messages correspondent to the triggered rules, indicating an abnormal situation, together with the following image . The messages shown for the triggered rules vary accordingly to the meaning of the rule.

Chapter 6

Conclusion

The present work aimed to develop the overall system architecture for an intelligent and innovative maintenance system and to develop the dynamic monitoring module that allows to monitor the condition of an asset and triggers alarms when a possible disturbance is detected or predicted. Such tool should be composed by an engine that executes the cross-check between the data and the rules and a visualization tool. Also, an Android application was developed in order to allow the maintenance technician, or another person to easily verify the system status out of the shop floor.

From the research made it was possible to understand that the maintenance strategies can strongly benefit from the application of emergent ICT technologies, such as IoT, Big data, advanced data analysis and cloud computing in order to collect, store and analyze the great amount of available data in the shop floor.

Furthermore, from the work developed it became clear that the implementation of a predictive maintenance approach in a production environment with legacy equipment presents great challenges. One of the challenges concerns the implementation of additional sensing devices on the equipments, which has associated costs that, depending on the desired parameters, may be high. Another challenge may arise when the asset is already collecting the desired parameters but, due to the privacy policies of the companies, some restrictions may be imposed to access the data.

The development of a system architecture and a set of tools to monitor the working condition of assets aims to consolidate and to help mature the predictive maintenance implementation so that manufactures can trust such systems and rely on them.

Future work efforts should focus on the improvement of the Node-RED monitoring and visualization tool by implementing the data collected from the shop floor and the specific rules of the case study. Also, the communication between the Node-RED and the Android mobile application will be improved by replacing the GET request for the POST method of the REST protocol.

Neither application has been tested by the operators in the factory environment, therefore the test and evaluation of the developed applications by the user is mandatory. It is also ambitioned to extend the developed system architecture to the remaining equipment on the shop floor.

Bibliography

[1] U. M. Al-Turki, T. Ayar, B. S. Yilbas, and A. Z. Sahin, *Integrated maintenance planning in manufacturing systems*. 2014, ISBN: 9783319062891. DOI: 10.1007/978-3-319-06290-7.

[2] K. A. H. Kobbacy and D. N. P. Murthy, *Complex System Maintenance Handbook*. Springer Series in Reliability Engineering series, 2008, ISBN: 978-1-84800-010-0. DOI: 10.1007/978-1-84800-011-7.

[3] M. Faccio, A. Persona, F. Sgarbossa, and G. Zanin, "Industrial maintenance policy development: A quantitative framework," *International Journal of Production Economics*, vol. 147, no. PART A, pp. 85–93, 2014, ISSN: 09255273. DOI: 10.1016/j.ijpe.2012.08.018.

[4] A. Sharma, G. S. Yadava, and S. G. Deshmukh, "A literature review and future perspectives on maintenance optimization," *Journal of Quality in Maintenance Engineering*, vol. 17, no. 1, pp. 5–25, 2011, ISSN: 13552511. DOI: http://dx.doi.org/10.1108/13552511111116222.

[5] G. Waeyenbergh and L. Pintelon, "A framework for maintenance concept development," *International Journal of Production Economics*, vol. 77, no. 3, pp. 299–313, 2002, ISSN: 09255273. DOI: 10.1016/S0925-5273(01)00156-6.

[6] J. Lee and B. Bagheri, "Cyber-physical systems in future maintenance," in *Lecture Notes in Mechanical Engineering*, vol. 20, 2015, pp. 299–305, ISBN: 9783319155364. DOI: 10.1007/978-3-319-15536-4_25.

[7] M. P. Stephens, *Productivity and Reliability-Based Maintenance Management*. Purdue University Press, 2010, ISBN: 978-1-55753-592-4.

[8] J. Lee, E. Lapira, S. Yang, and A. Kao, "Predictive manufacturing system - Trends of next-generation production systems," in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 2013, pp. 150–156, ISBN: 9783902823335. DOI: 10.3182/20130522-3-BR-4036.00107.

[9] Ford, *The Henry Ford story*. [Online]. Available: https://www.ford.co.uk/experience-ford/history-and-heritage%7B%5C#%7Dhenryfordstory (visited on 03/02/2018).

[10] Toyota, *Toyota Global Site | Production System*. [Online]. Available: http://www.toyota-global.com/company/vision%7B%5C_%7Dphilosophy/toyota%7B%5C_%7Dproduction%7B%5C_%7Dsystem/ (visited on 03/11/2018).

[11] Toyota., *Toyota Global Site | The origin of the Toyota Production System*. [Online]. Available: https://www.toyota-global.com/company/vision_philosophy/toyota_production_system/ (visited on 03/12/2018).

[12] J. Lee and E. Lapira, "Predictive Factories: The Next Transformation," *Manufacturing Leadership Journal. Frost Sullivan*, pp. 2–9, 2013, ISSN: 1041-4347. DOI: 10.1109/TKDE.2012.172.

[13] G. Niu, B. S. Yang, and M. Pecht, "Development of an optimized condition-based maintenance system by data fusion and reliability-centered maintenance," *Reliability Engineering and System Safety*, vol. 95, no. 7, pp. 786–796, 2010, ISSN: 09518320. DOI: 10.1016/j.ress.2010.02.016.

[14] A. G. Starr, "A structured approach to the selection of condition based maintenance," *Fifth International Conference on FACTORY 2000 - The Technology Exploitation Process*, vol. 1997, no. 435, pp. 131–138, 1997. DOI: 10.1049/cp:19970134.

[15] J. Lee, E. Lapira, B. Bagheri, and H. an Kao, “Recent advances and trends in predictive manufacturing systems in big data environment,” *Manufacturing Letters*, vol. 1, no. 1, pp. 38–41, 2013, ISSN: 22138463. DOI: 10.1016/j.mfglet.2013.09.005. arXiv: arXiv:1011.1669v3.

[16] T. Sutharssan, S. Stoyanov, C. Bailey, and C. Yin, “Prognostic and health management for engineering systems: a review of the data-driven approach and algorithms,” *The Journal of Engineering*, no. 7, pp. 215–222, 2015.

[17] N. M. Vichare and M. G. Pecht, “Prognostics and health management of electronics,” *Components and Packaging Technologies, IEEE Transactions on*, vol. 29, no. 1, pp. 222–229, 2006, ISSN: 1521-3331 VO - 29. DOI: 10.1109/TCAPT.2006.870387.

[18] S. Das, R. Hall, and S. Herzog, “Essential steps in prognostic health management,” in *2011 IEEE Conference on Prognostics and Health Management (PHM)*, IEEE, 2011, pp. 1–9.

[19] J.-H. Shin and H.-B. Jun, “On condition based maintenance policy,” *Journal of Computational Design and Engineering*, no. 2, pp. 119–127, 2015.

[20] A. K. Jardine, D. Lin, and D. Banjevic, *A review on machinery diagnostics and prognostics implementing condition-based maintenance*, 2006. DOI: 10.1016/j.ymssp.2005.09.012. arXiv: 0208024 [gr-qc].

[21] Mimosa - An Operations and Maintenance Information Open System Alliance, *MIMOSA OSA-CBM*, 2010. [Online]. Available: http://www.mimosa.org/mimosa-osa-cbm (visited on 20/03/2018).

[22] ISO, *ISO 13374, Condition monitoring and diagnostics of machines — Data processing, communication and presentation*, 2012.

[23] M. Lebold, K. Reichard, C. S. Byington, and R. Orsagh, “OSA-CBM architecture development with emphasis on XML implementations,” *Maintenance and Reliability Conference (MARCON)*, pp. 6–8,

[24] M. Lehtonen, "On the optimal strategies of condition monitoring and maintenance allocation in distribution systems," in *2006 9th International Conference on Probabilistic Methods Applied to Power Systems, PMAPS*, 2006, ISBN: 9171783520. DOI: `10.1109/PMAPS.2006.360278`.

[25] IEC, *IEC 60300-3-11:2009, Dependability management - Part 3-11: Application guide - Reliability centred maintenance*, 2009. [Online]. Available: `https://webstore.iec.ch/publication/1296`.

[26] Widget Brain, *How predictive maintenance assures your assets are always in good health*. [Online]. Available: `https://widgetbrain.com/predictive-maintenance-asset-health/` (visited on 06/25/2018).

[27] R. K. Mobley and R. K. Mobley, "4 – Benefits of Predictive Maintenance," in *An Introduction to Predictive Maintenance*, 2002, ISBN: 9780750675314. DOI: `10.1016/B978-075067531-4/50004-X`.

[28] M. Patel and Infosys, *The Future of Maintenance [White Paper]*, 2018.

[29] Senseye, *Forecast machine failure, scalable predictive maintenance*. [Online]. Available: `https://www.senseye.io/` (visited on 04/02/2018).

[30] L. Liao, H. Wang, and J. Lee, "A reconfigurable watchdog Agent® for machine health prognostics," *International Journal of COMADEM*, vol. 11, no. 3, pp. 2–15, 2008, ISSN: 13637681.

[31] A. Cachada, J. Barbosa, P. Leitão, C. A. S. Geraldes, L. Deusdado, J. Costa, C. Teixeira, J. Teixeira, A. H. J. Moreira, P. M. Moreira, and L. Romero, "Maintenance 4.0: Intelligent and Predictive Maintenance System Architecture," *IEEE 23rd International Conference on Emerging Technologies and Factory Automation*, pp. 139–146, 2018.

[32] G. A. Boy, *The Handbook Of Human-Machine Interaction A Human-Centered Design Approach*. Ashgate Publishing Limited, 2011, ISBN: 9780754675808.

[33] D. Dong, X.-Y. Li, and F.-Q. Sun, "Life prediction of jet engines based on lstm-recurrent neural networks," pp. 1–6, 2017.

[34] O. Aydin and S. Guldamlasioglu, "Using lstm networks to predict engine condition on large scale data processing framework," pp. 281–285, 2017.

[35] A. Karpathy, J. Johnson, and L. Fei-Fei, "Visualizing and understanding recurrent networks," *ArXiv preprint arXiv:1506.02078*, 2015.

[36] V. Havard, D. Baudry, X. Savatier, B. Jeanne, A. Louis, and B. Mazari, "Augmented industrial maintenance (AIM): A case study for evaluating and comparing with paper and video media supports," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9768, 2016, pp. 302–320, ISBN: 9783319406206. DOI: `10.1007/978-3-319-40621-3_22`.

[37] R. Azuma, "A survey of augmented reality," *Presence*, vol. 6, no. 4, pp. 355–385, 1997.

[38] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre, "Recent advances in augmented reality," *IEEE Comput. Graph. Appl.*, vol. 21, no. 6, pp. 34–47, Nov. 2001, ISSN: 0272-1716. DOI: `10.1109/38.963459`.

[39] B. C. Kress and W. J. Cummings, "Towards the ultimate mixed reality experience: Hololens display architecture choices," *SID Symposium Digest of Technical Papers*, vol. 48, no. 1, pp. 127–131, 2017. DOI: `10.1002/sdtp.11586`.

[40] P. Greenhalgh, B. Mullins, A. Grunnet-Jepsen, and A. K. Bhowmik, "Industrial deployment of a full-featured head-mounted augmented-reality system and the incorporation of a 3d-sensing platform," *SID Symposium Digest of Technical Papers*, vol. 47, no. 1, pp. 448–451, 2016. DOI: `10.1002/sdtp.10704`.

[41] S. Henderson and S. Feiner, "Exploring the benefits of augmented reality documentation for maintenance and repair," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 10, pp. 1355–1368, Oct. 2011, ISSN: 1077-2626. DOI: `10.1109/TVCG.2010.245`.

[42] A. Nee, S. Ong, G. Chryssolouris, and D. Mourtzis, "Augmented reality applications in design and manufacturing," *CIRP Annals*, vol. 61, no. 2, pp. 657–679, 2012, ISSN: 0007-8506. DOI: `https://doi.org/10.1016/j.cirp.2012.05.010`.

[43] S. Werrlich, E. Eichstetter, K. Nitsche, and G. Notni, "An overview of evaluations using augmented reality for assembly training tasks," *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, vol. 11, no. 10, pp. 1142–1148, 2017, ISSN: eISSN:1307-6892.

[44] L. S. Nelson, "The Shewhart Control Chart—Tests for Special Causes," *Journal of Quality Technology*, vol. 16, no. 4, pp. 237–239, 1984, ISSN: 0022-4065. DOI: `10.1080/00224065.1984.11978921`.

[45] Business Performance Improvement (BPI), *Nelson Rules - Lean Manufacturing and Six Sigma Definitions*. [Online]. Available: `http://leansixsigmadefinition.com/glossary/nelson-rules/` (visited on 06/02/2018).

[46] J. Foundation, *Node-RED*. [Online]. Available: `https://nodered.org/` (visited on 03/21/2018).