Università degli Studi di Modena e Reggio Emilia

DOTTORATO DI RICERCA IN INGEGNERIA
DELL'INNOVAZIONE INDUSTRIALE

Ciclo XXXIV

Distributed Systems Engineering to orchestrate new I4.0 technologies for Agile Product Development

Candidato: Andrea Cervo

Relatore (Tutor): Prof. Cesare Fantuzzi

Coordinatore del Corso di Dottorato: Prof. Franco Zambonelli

Esame finale anno 2021

Abstract

Industry 4.0 is a concept describing the implementation of Industrial cyber-physical systems in modern industries, also defined “smart industries”. With Industry 4.0 a wide set of technologies is intended: Internet of things, cloud computing, data analytics, distributed control systems, advanced robotics, artificial intelligence. Major barriers for the rapid and widespread adoption of these new technologies in SMEs are represented by the technical background required to operators, the time to learn new methodologies and technologies and the costs.

The purpose of this PhD thesis is to describe the role of Systems Engineering in the Industry 4.0, to enable a structured approach for product development when these technologies are required for the applications in scope. The focus of the research is on the cross-domain methodology based on model-driven design applied to distributed applications, increasingly relevant as they are the basis of IoT systems and decentralized control systems. These technologies are well suited also for the flexibility and scalability they offer in case of lean/iterative project management methodologies like Agile, SaFE.

The thesis presents an E2E approach consisting of: Requirements elicitation and management, System Architecture design, Software design, Verification and Validation methodologies, KPI definition for analysing technical success of the application.

The technologies analysed and deepened in this thesis are industrial multi-agent systems, data analytics leveraging also on Artificial Intelligence and Natural language processing, data presentation and IoT networks, both with standard internet protocols and low level, energy-optimized, communication protocols.

For each topic discussed in the thesis, a review of current state of the art, leveraging on recent articles and papers of scientific relevance, and a technical use case developed and validated in laboratory are presented. The technical use cases are all based on requirements coming from Industrial needs, to keep the concreteness for the design and validation process. The domains verified with the technical use cases are: Automation Engineering, Software Engineering and Smart infrastructure engineering.

Finally, purpose of this thesis is also to provide ideas and suggest room for improvement or further study for each topic, with the aim to create an entry point for deepening on related issues for researchers in the Software and Automation community.

Sommario

Industria 4.0 è un concetto che ha l'intento di descrivere l'implementazione dei sistemi ciberfisici nelle industrie moderne, anche chiamate "Industrie Intelligenti". Con Industria 4.0 si intende un insieme vasto di tecnologie: internet delle cose, computazione tramite sistemi cloud, analisi dati, controllo di sistemi distribuiti, robotica avanzata, intelligenza artificiale. Le principali barriere per un'adozione rapida e diffusa di tali tecnologie nelle piccole medie imprese sono rappresentate dalle competenze tecniche richieste agli operatori, il tempo necessario per imparare nuove metodologie e tecnologie e i costi correlati.

Lo scopo di questa tesi di Dottorato è di descrivere il ruolo dell'ingegneria dei sistemi in Industria 4.0, per abilitare un approccio strutturato allo sviluppo di prodotto quando tali tecnologie sono richieste per le applicazioni in scopo. L'attenzione della ricerca è posta su metodologie per l'analisi di domini differenti basate sulla progettazione guidata dai modelli applicata ai sistemi distribuiti, sempre più rilevanti poiché sono la base di sistemi IoT e di controlli decentralizzati.

La tesi presenta un approccio completo costituito da: elicitazione e gestione dei requisiti, progettazione dell'architettura di sistema, progettazione software, metodologie di verifica e validazione, definizione di metriche per analizzare il successo tecnico di un'applicazione.

Le tecnologie analizzate e approfondite nella tesi sono i sistemi industriali multi-agente, l'analisi dati basata su intelligenza artificiale e analisi del linguaggio naturale, sistemi di presentazione dei dati e reti di dispositivi IoT, sia con protocolli di rete internet standard che con protocolli di comunicazione di basso livello ed energeticamente ottimizzati.

Per ogni tema discusso nella tesi sono presentati lo stato dell'arte basandosi su recenti articoli di rilevanza scientifica e uno caso studio tecnico sviluppato e validato nei laboratori dell'Università. I casi studio sono basati su requisiti provenienti dal mondo industriale al fine di mantenere concretezza nel processo di progettazione e validazione. I domini di questi prototipi di validazione sono: Ingegneria dell'Automazione, Ingegneria del Software, Ingegneria delle infrastrutture intelligenti.

Infine, scopo della tesi è anche quello di fornire idee e suggerire spazi di miglioramento o di ulteriori studi per ciascun ambito, con il fine di creare un punto di accesso per l'approfondimento su temi correlati rivolto ai ricercatori della comunità software e automazione.

Summary

Chapter 1: Industry 4.0

1.1 Introduction

Industry 4.0 is the conceptualization of the rapid evolution from technological perspective of the industries of 21st century, driven by the greater flexibility required by the markets. Such journey towards digitalization is impacting the products and services delivered by manufacturing industry, the internal processes used by the companies and societal patterns.

The terms Industry 4.0 comes from a German initiative, “Industrie 4.0”, that was formalized in November 2011 with an article published by German government containing the pillars for high-tech strategy for 2020 to empower industry competitiveness. Nowadays I4.0 is used to refer to the global digital transformation in industries aimed to reach the concept of Smart Industry.

This industrial revolution is mainly identified by the pervasive introduction and integration of Cyber-physical systems into manufacturing and logistics from automation perspective as well as the usage of Internet of Things (IoT), Big Data and software services in industrial processes from software and data management perspective [1].

1.1.1 New technologies and paradigms

Differently from previous industrial revolutions, Industry 4.0 was anticipated and announced by the academic community prior to actual implementation in the industrial sector: the purpose of this new technological revolution is to place the needs of the individual customer as a central objective.

In order to achieve such digitalization objective together with the goal of flexible deliveries, the extant academic research asserts that the I4.0 transformation takes place with the introduction of new technological skills and disciplines: Big Data Analytics and Machine Learning, Autonomous Robots, Digital Twin, Horizontal and Vertical System Integration, Internet of Things (IoT) (including sensors), Cloud Computing, Additive Manufacturing, Augmented Reality [2], [3].

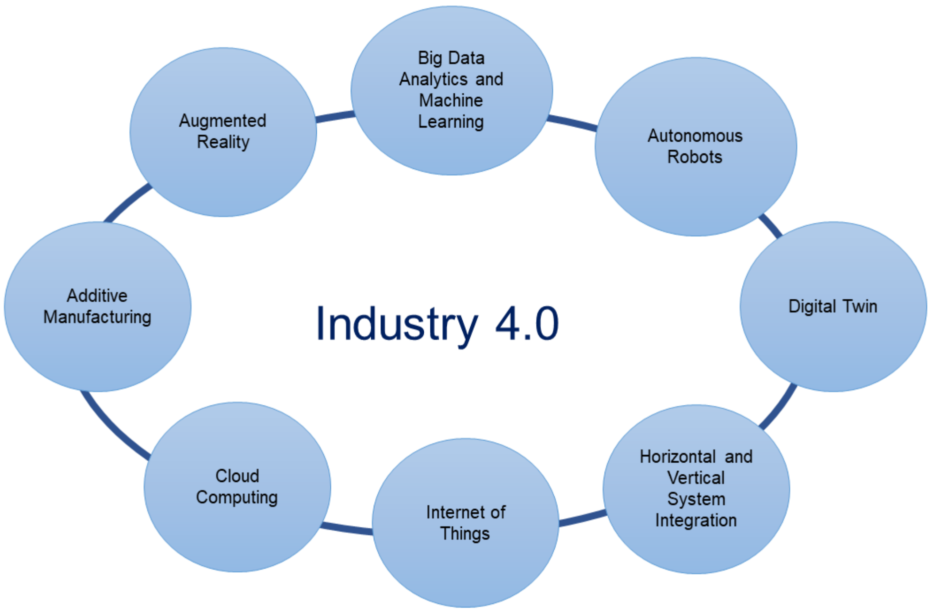

Figure 1: New technological portfolio of Industry 4.0

The purpose of these new technologies is to facilitate and provide the enabling elements to realize the principles of the smart industry:

- **Decentralised decision-making**: the monitoring and control system is no longer centralized in a single computational unit, due to the physical complexity of the system to be monitored and the high flexibility and reconfiguration capability required by modern systems. The data collection and control algorithms are distributed over multiple nodes: each computational node can collect data and information, make local decisions and exploit the interconnection of the systems to make decisions guided by a shared system policy.
- **Information transparency**: sensor information must be collected and recorded by each computational node, stored and exposed for other services in order to aggregate and create analytics. The ultimate goal is to create a dataset populated from various sources to extrapolate high-level information about the system used for monitoring, diagnostics and prognostics. The success of IoT systems is also due to the fact that embedded devices that enable data collection are becoming cheaper and low-power protocols to allow for data exchange and control policies are emerging and continually improving [4].
- **Decision support system**: the industrial cyber-physical system must have software modules that, following data analysis rules, can create alarms, feedback

and graphs to support human operators and managers in understanding the global state of the plant and production processes for daily activities, to support quality and optimization operations. In summary, the main goal is to support human operations through data-driven services.

1.1.2 New design principles for I4.0 readiness

The main drivers of such technological revolution are:

- the new market demand that aims for products with high quality, high customization and lower prices, [5].
- the increasingly globalized distributed supply chain, which involves various companies and plants that need to exchange data to synchronize their activities, improving overall efficiency, [6].

To bring these new paradigms to the industry, it is necessary to introduce new design principles that facilitate the digitalization process such as modularity, scalability, decentralization, virtualization, real-time data availability and processing, mass personalization, interoperability, [7].

1.1.3 Challenges in the adoption of I4.0 enablers

Currently, especially in small-medium enterprises, a slow change towards I4.0 technologies is observed. From the literature review the main reasons for this slowness are identified in obstacles such as infrastructural problems, lack of skilled labour, lack of standards [8] and also a gap in literature analysis of empirical studies linking I4.0 technologies to organizational advantages, especially in emerging economies, [9].

The main problem is that, in parallel with the diffusion of new technologies and new enablers, there is still no homogeneous and rigorous approach to the management of complex and interconnected systems. The challenges of this industrialization initiative are not fully understood by small-medium enterprises (SMEs) as there is no single and clear vision of the new manufacturing paradigm both in terms of implications and consequences [10]. The main technological barriers studied by researchers in SMEs are the lack of IT skills, which creates the fear of losing their jobs and the unclear role of the human operator in a sector that is becoming more and more automated [11] [12].

1.2 Outline: PhD scope and “PID: Impresa 4.0” initiative

This thesis describes the research results achieved with the projects developed during the 3 years of the PhD.

The purpose of this research was mainly oriented to understand the difficulties of small and medium-sized enterprises in adopting I4.0 enablers, supporting them in the transfer of technical knowledge and in the development of concrete prototypes by highlighting the project management, design and implementation workflow, leveraging on the principles of Systems Engineering methodology.

For this reason, the focus is more on designing clear guidelines and a standard approach for requirements analysis and system architecture design, than on full attention to specific technology. As highlighted in the previous section, academia has already spent a great deal of energy documenting the technological opportunities in adopting I4.0 enablers, but there is still a lack of integrated support for adoption in SMEs, guiding the design and implementation process end- to-end.

In my humble opinion, currently it is important to focus on the central role the University and associated engineering research centres can play in knowledge transfer for practical adoption of I4.0 enablers by SMEs, making the wide-scale implementation of digital technologies a reality.

During the PhD I had the opportunity to focus on projects aimed at supporting small-medium enterprises in understanding how I4.0 technologies can enhance and expand their business, defining which technical skills are needed in terms of design, implementation, validation and life cycle support.

The result of these experiences was the generalization of the design and development process, defining standard guidelines.

In this journey towards I4.0 technologies, an initiative born as a collaboration between the University of Modena and Reggio Emilia and the Chamber of Commerce of Reggio Emilia was relevant, the project was called "PID: Impresa 4.0" [13]. Another important opportunity was the possibility of carrying out technical research in the field of automation and data analysis for ongoing projects in the ARSControl automation laboratory, [14].

The elicitation of the requirements carried out during the "PID: Impresa 4.0" initiative was based on structured interviews with project managers and company representatives of local companies, presenting them with the main enabling factors recommended by Industry 4.0 and showing examples of applications in different domains. The purpose of these interviews was to understand their current gaps in digitalization and their main interest in adopting new enabling factors to improve their core business, in order to focus on recurring interests and develop prototypes that meet these needs.

By interviewing 25 companies, we identified the recurrent digitalization topics of interest:

- Sensors and IoT systems [6 requests]
- Data analytics and Data visualization [6 requests]
- Data classification and decision support systems [3 requests]
- Digital Twin [2 requests]
- Cooperative robotics [2 requests]
- Artificial computer vision [3 requests]

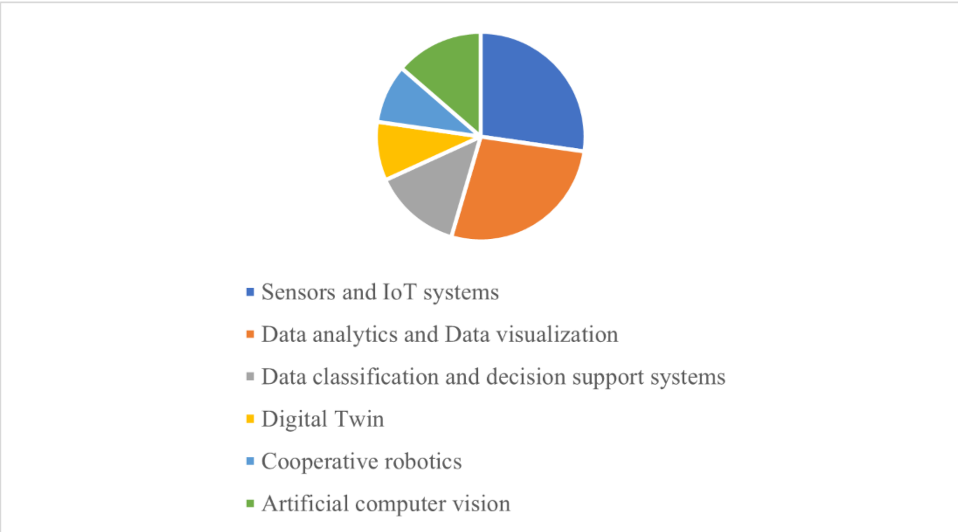

Figure 2: Interest in I4.0 technologies with interviews for PID: Impresa 4.0

The aim of the initiative was mainly to identify the common difficulties and obstacles in the implementation of such technologies, agreeing the general requirements and expectations for the development of the prototype. All the prototypes developed both in the "PID: Impresa 4.0" project and in the applied research projects of ARSControl were intentionally left sufficiently generic, so that more companies with similar interests could then proceed in the customization. In this technology transfer program, I mainly focused, together with other researchers from the same university laboratory, on the topics of the software and automation domain.

1.2 PhD thesis structure

The PhD thesis is organized with the following structure:

- Chapter 1: State of the art of Industry 4.0 and current challenges, scope of the PhD and structure of the thesis.
- Chapter 2: Systems engineering principles and their application in the management of distributed systems and I4.0 technologies, for agile product design and implementation. Software configuration management and version control tools.
- Chapter 3: Systems Engineering design flow for "Signal and Data Processing and Visualization Systems". Presentation of the use cases "Grape ripeness calculation systems" and "Real-time recognition system of acoustic pulse frequencies".
- Chapter 4: Systems engineering design flow for “Digital Twin and IoT multi-agent systems”. Presentation of the use cases “BIM Plant Digital Twin updated by MQTT-based IoT WSN” and “BLE-based WSN for energy optimized applications with synchronized deep sleep phases”.
- Chapter 5: Systems engineering design flow for “Big Data and Decision Support Systems”. Presentation of the use cases “Analysis of the causes of downtime in ceramic plants” and “Similarity analysis in unstructured texts for non-compliance reports”.
- Chapter 6: Conclusions, with a summary of the research achievements and the lesson learned.

Chapter 2: Systems Engineering principles for the management of distributed systems and I4.0 technologies

To define the discipline of Systems Engineering, we leverage the definition provided by INCOSE (International Council on Systems Engineering): “Systems Engineering is a transdisciplinary and integrative approach to enable the successful realization, use, and retirement of engineered systems, using systems principles and concepts, and scientific, technological, and management methods”, [15].

2.1 Systems Engineering principles

The goal of systems engineering is to analyse and design complex systems, generally including different technologies, starting from a structured approach in stakeholder needs collection and analysis: the role of this discipline is relevant in the domain of Industry 4.0 and digitalization process. One issue highlighted by researchers is the difficulty of business and management in understanding how to leverage on the new technologies for current business and new opportunities: this represents an objective difficulty in transforming business requirements into system requirements implemented through new technologies.

Another reason for adopting a structured systems engineering approach is because modern IT solutions are generally geared towards modularization: this will allow for a quick and independent release process to fix and add functionality on the one hand, but on the other hand it requires a more careful and rigorous approach to configuration management and verification and validation compared to the past, to provide consistent and integrated solutions.

The Systems Engineering discipline has the purpose to define for a specific system the design, implementation, verification and validation and life cycle phases.

The systems engineering approach is typically represented by the V-model (Figure 3) that summarizes the entire development flow of a mechatronic product, from stakeholder requirements elicitation to product life cycle support. It is called “V” because there are two sides in the diagram:

- the left side represents the design and implementation phases, indeed there are the activities starting with high-level requirements analysis and arriving to the subsystem module implementation.
- the right side of the "V" represents the phases of systems integration and testing better known as verification and validation process, prior to the commercial release and the solution deployment on the market, [16].

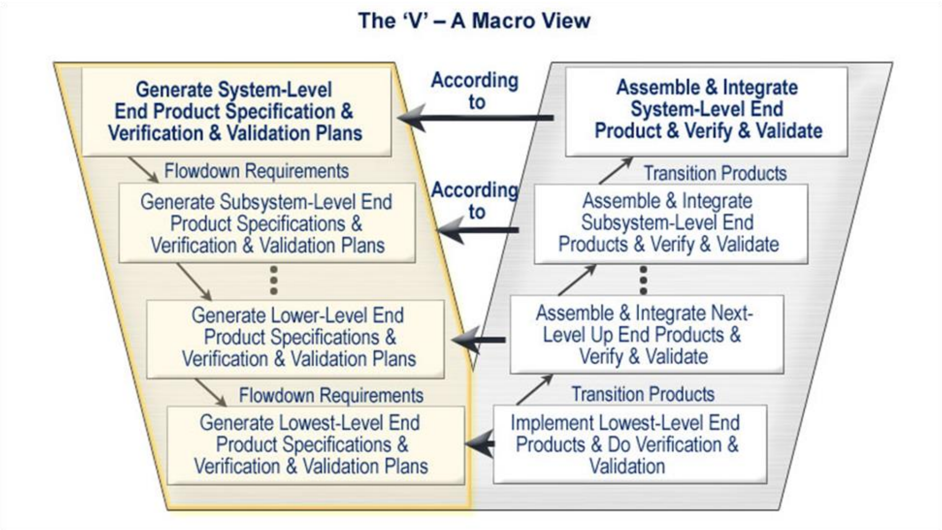

Figure 3: Systems Engineering V-model, (Prosnik 2010). Released by the Defense Acquisition University (DAU)/U.S. Department of Defense (DoD)

2.2 Systems Engineering in Iterative processes

The challenge of the Systems Engineering discipline applied to I4.0 technology management is that the modern software development process is generally based on an incremental and iterative approach to gather early feedback from stakeholders with shorter iterations for implementation. The implication of shorter iterations and frequent changes to requirements requires more attention in configuration management and precise verification and validation activities to ensure new development and avoid regressions in the previous functionality to be maintained.

This way of designing and developing is called Agile and it is an iterative development process peculiar of software development, [17]. The Agile way of working is promoting the adoption of modular, distributed and decentralized systems that can make systems scalable and easier to be modified for change requests: for this reason, two important design principles to be taken into account for I4.0 are modularization and decentralization.

To tackle the complexity coming from distributed systems engineering, the Scaled Agile

Framework (SAFe) recommends the adoption of a development process entirely based on the Systems Engineering principles by means of specific artifacts to be discussed in different project governance forums:

- Stakeholder/Business need = Epic
- Systems Requirements = Product Feature
- Architectural Design = Architectural Runway Enabler
- Sub-system Requirements = User stories for development
- Verification = User stories for testing
- Validation = System Demo with stakeholders before official release

2.3 Agile framework to organize Research projects

During the "PID: Impresa 4.0" initiative we decided to apply the Agile SCRUM framework to organize the research activities: according to the framework, we planned recurrent system design sessions with technology specialists and recurrent validation sessions with representatives of management and business of the involved companies interested in the research topics. The overall hypothesis was that using such framework, not exclusively limited to software development according to recent research [18], it was possible to facilitate the sharing of technical knowledge between universities and companies as well as continuously meeting high-level needs with solutions implemented with modern digital enablers.

The SCRUM development framework is really focused on the incremental Systems Engineering approach with development iterations (defined sprints) lasting up to 4 weeks, defining ceremonies to keep monitored the evolution of the solution in terms of requirements, architecture, design and verification and validation:

- Sprint planning: session in which the Scrum team defines what can be developed and delivered in the sprint and how the set of new features will be implemented, [19].
- System and code review: session where developers and test engineers can evaluate the code and test results of test automation platform before merging the new source code in a branch prior to quality assurance (QA) or test environment, [20].

- Product Backlog refinement: session in which the Product Owner (or Project Manager) together with the team adds details, estimates and orders the Product Backlog items (user stories), [21].
- Sprint review: session in which the team inspects the outcome of the development iteration, demonstrates new features with a product demo, and determines future adaptations with inputs from key stakeholders, [22].
- Sprint retrospective: session in which the development team analyses the quality of the product delivered at the end of the development iteration and searches for possible improvements in the way of working.

This iterative development framework has been surprisingly successful in enabling technology knowledge transfer during the PhD activities. The short iterations and the constant feedback obtained by key stakeholders guaranteed quick fix in case of wrong business need understanding. After a few months of applying the framework, we noticed a progressive understanding by the management of SMEs of new possibilities and opportunities in their business area, together with a proactive approach in the investigation of some innovative technologies that were not even considered before.

2.4 Version control and configuration management to orchestrate the release process

During the PhD research activities for "PID: Impresa 4.0", a careful approach to configuration management was used, since all systems were designed to maximize the modularity and parallelization of the verification and validation activities on the modules to be integrated.

Modern software engineering tools support the configuration and version management to keep consistency among the different components, starting from a clear management of source code in different branches.

The most important tool we adopted is GIT [23], a free and open source version control system, widespread both in software engineering and automation engineering: the tool is used to synchronize the development activities carried out by different developers in a team possibly even on the same software modules, using branching strategies to control the added features, to improve the quality of pull requests and finally to optimize the release process.

The core of GIT is the branching model, that provides the opportunity:

1. to control the consistency of the software before merging on the main branch via pull-requests (i.e., requests from contributors to development lead to the original software),
2. to automate the versioning with a semantic versioning approach,
3. to trigger automatic actions on the software while integrating previous to pull-requests approval, like automatic test pipelines, and to produce automatically the expected artifacts to be deployed on QA environment or directly on customer hosting systems,
4. to create release branches to maintain in life cycle previous versions of the software commercially released.

Git is very well integrated with most of Integrated Development Environments (IDEs) and open source services for Continuous Integration and Continuous deployment: it is no coincidence that the latest DevOps trends identify a new approach, identified with the term GitOps, based on the intensive use of GIT.
Git has effectively become a standard for software engineering and for these reasons there are several cutting-edge development flows, which suggest different branching approaches depending on the type of product and the iterations needed for the specific project, such as Git Flow, GitHub Flow and Trunk-based Development, [24].

2.5 Formal models for the analysis of requirements and architectures

For the analysis and design of the use cases, I used formal models widely adopted in Systems Engineering and Software Engineering: Systems Modeling Language (SysML) and Unified Modeling Language (UML).

In particular, the most used diagrams were:

1. Use case diagram, to allow a functional decomposition of the use cases and to prioritize the functionalities to be technically analysed and implemented in the various development iterations. The use case diagram was the starting point for the formalization of the Stakeholder Requirements and System Requirements that will be shown in the future chapters of the thesis.
2. Sequence diagram, to allow the modeling of the interactions between distributed systems and to allow a study of the temporal dependencies between the various software modules to be created. The sequence diagrams were the starting point

for the analysis of the architectural system, in order to have a clear view of the interfaces between the modules.

3. State machine diagram, to allow the analysis of the internal behaviour of each module. A precondition for this model has always been the finalization of the sequence diagram to be able to consider all possible external requests / triggers for correct management in the business logic.

For the realization of the diagrams, we used a free software, Diagrams.Net [25]: the advantage of this software is the presence of numerous library elements for the construction of UML and SysML diagrams and the possibility of sharing the diagrams on cloud platforms.

Chapter 3: Signal and Data processing and visualization systems

The first I4.0 topic investigated during the PhD was about systems for "Signal and data processing and visualization systems".

The use cases selected for this area came from different industrial domains: agricultural and automotive.

In both cases it was interesting to note that the data acquisition systems were already present in the company; the missing part was a data processing and visualization system optimized to interpret and extract value from the collected data.

- Use case 1 - "Grape ripeness calculation systems": a web-based data analytics tool to compute the current ripeness status by means of IoT sensors data on environmental parameters (e.g. temperature, humidity, pressure etc.) collected via embedded devices.
- Use case 2 - "Real-time recognition system of acoustic pulse frequencies": a classification system to recognize high frequency acoustic pulses, where the frequency encodes a specific message-meaning.

3.1 State of the art

Signal processing and analysis is the engineering discipline that refers to acquisition, storage, visualization and extraction of information from electrical signals, [26]. Nowadays, talking about signal processing is synonymous with digital signal processing, due to the spread of high-speed processing systems and inexpensive software libraries for implementing mathematical algorithms to support signal analysis.

In the acoustic engineering field, signal analysis is generally used:

1. to separate signals of interest from noise,
2. to extract features of interest for classification, pattern recognition and automatic speech recognition.

In the domain of signal analysis, both advanced mathematical model-driven algorithms (such as spectrum analysis with Fourier series, Fourier transform, Shannon theorem, etc) and black-box data-driven algorithms (such as machine learning classifiers) are used, [27]. Generally, machine learning is applied to complex models that are difficult to represent with explicit mathematical models, such as for human speech recognition, music classification, acoustic scene classification, tagging and generation problems, [28]. Currently, machine learning (ML) algorithms are mainly used on electronic devices endowed with great computing power, operating system and the ability to interpret high-level programming languages such as Python, Java, JavaScript and C++, [29]. The class of ML problems related to tag classification and prediction is also called Supervised Machine Learning: the classification problems are mainly clustered in "Lazy learners" (e.g. k-nearest neighbour, Case-based reasoning) and "Eager learners" (e.g. Decision Tree, Naive Bayes, Artificial Neural Networks), [30]. The risk underlying Machine Learning algorithms is to avoid a deep understanding of the engineering models that produced the dataset: physical understanding remains a prerequisite since a good fitting model is produced only by a good and balanced training dataset, capable of stimulating all the input variables and producing the related outputs.

Model-based algorithms, on the other hand, remain the most widely used approaches for engineering problems related to physics (as there is a solid idea of how the physical system works), where data processing capability in real-time is also required, such as failure-detection and prognostics in mechanical engineering, model pole and zero recognition for automatic control in automation engineering or spectrum analysis in acoustic engineering.

Regarding the state of the art of dashboard and visualization systems in the context of Smart Industry, web-based human-machine interfaces (HMIs) are becoming increasingly popular in industrial environment, especially for "Supervisory Control And Data Acquisition" (SCADA) systems and data analytics applications. We define HMI system, a communication system between industrial system and operator, in order to understand the current state of the system and view aggregate parameters and KPIs to understand states, values, alarms and trends, [31]. The main advantages deriving from the use of web technologies for HMI systems are:

1. Adaptability and multi-device support: nowadays web development frameworks for front-end applications offer responsive graphics capabilities to adapt the web page layout to the screen size of the client device. This is an important feature considering the spread of mobile devices in the industrial field. Another great advantage deriving from web-based visualization systems is the possibility of

exploiting the application service even remotely if hosted on cloud services, interoperability on different operating systems and the ability to multi-access.

2. Evolving web framework: the second reason for choosing web-based HMI and dashboards is the continuous evolution, research and growth of web technologies, which guarantee a good portfolio of software libraries, large communities for the development of open source software and development prospects continuous for improvement against cyber-attacks.

3.2 Problem statement: Use cases and Stakeholder needs

In both industrial use cases selected for the development of the research topic, a system for data generation and transfer was already present and developed, but a data/signal processing system was missing to provide high-level output for human understanding and producing any triggers for automatic alerts.

Stakeholder need ID	Stakeholder need definition
SN1	The signal processing module for use case 1 shall provide a dashboard system with the possibility to upload and store environmental sensor datasets. The system is aimed to provide to the user: 1. Information on the state of ripeness of the grapes, 2. The ability to visualize historical information about harvest dates of previous years, 3. The comparison of the trends of ripeness of different years.
SN2	The data processing module for use case 2 shall provide a real-time signal classifier for the identification of specific frequency ranges of acoustic pulses, with the possibility to produce in output of a digital IO a different trigger depending on the frequency range identified.

3.3 System Requirements elicitation

Requirements elicitation for use case 1

System Requirement ID	System Requirement definition
SR1	The data processing system shall be capable to process data stored in a comma-separated values (CSV) file.
SR2	The data processing system shall have the capability to compute the aggregate parameter used by human operator, leveraging the last updated dataset uploaded by the user.
SR3	The data processing and storage system shall be hosted on a cloud platform.
SR4	The historical data uploaded by the user to the system must be archived in a file archive to be developed in the system.
SR5	The user interface must be a web application, suitable for different devices (smartphone, tablet, laptop).

Requirements elicitation for use case 2

System Requirement ID	System Requirement definition
SR1	The signal processing system shall be capable to process the Fast Fourier transform of a sampled signal in real-time, temporarily stored in RAM as an array.
SR2	The signal processing system shall have the capability to compute on polling the time series indicators to recognize the system status, having the internal time schedule conforming to a maximum latency between signal generation and signal recognition.
SR3	The signal processing system shall provide the possibility to interface with external system to trigger automatic action on a predefined condition on the calculated parameters.
SR4	The signal processing system shall be hosted on an embedded device.

SR5	The feedback about signal recognition shall be visual feedback to the user via LEDs, with an association between frequency range and the lit led.
SR6	The signal processing system shall be capable to reject frequencies caused by random noise.

The general guidelines, which summarize the use case referral, are important for defining:

1. the hosting system in which to install the signal/data processing system due to the impact on overall costs and information availability requirements,
2. the interface to collect the dataset, its contract and the time required for data processing, since these choices have an impact both on the technologies to be applied and on the constraints on the type of algorithm to be implemented,
3. the type of output expected by the system, both in terms of HMI (human-machine interface) and the technical signal to trigger other automatic actions.

3.4 Definition of functional architecture and physical architecture

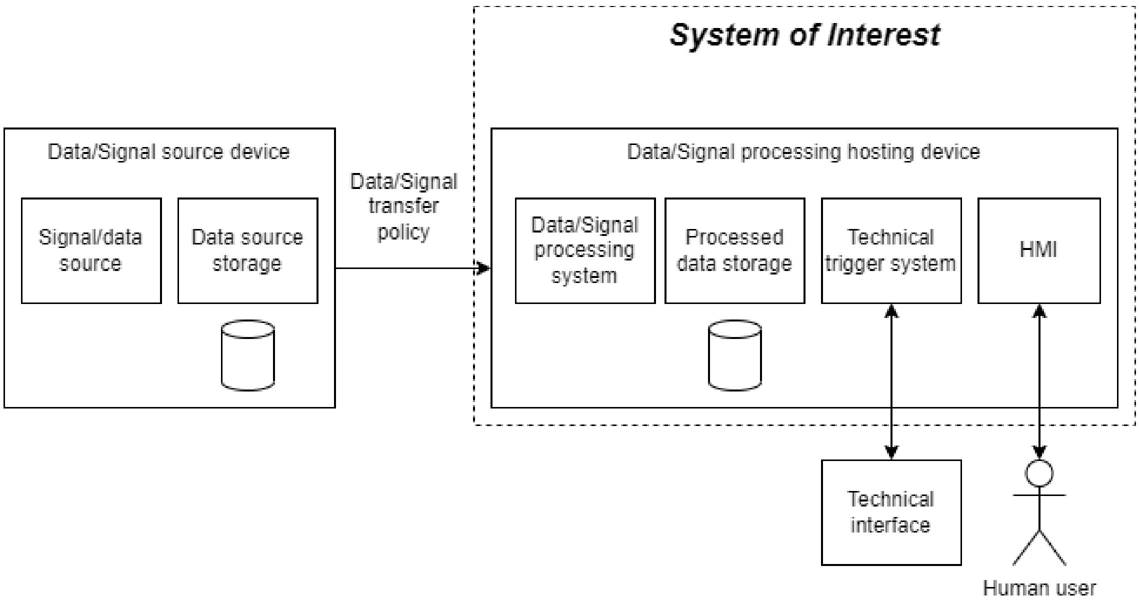

Figure 4: Functional Architecture for Data processing and visualization systems

The design of a clear functional architecture (Figure 4) is important to visualize the main interfaces to be considered in the development of the System of Interest, according to the

agreed system requirements. Interface management is important when analysing the physical architecture and a modular approach can maximize system reusability when contract requirements change with external systems.
The functional architecture resulting from the two selected use cases is common, even if the physical architectures developed are distinct, due to different non-functional requirements defined in the system requirements phase.

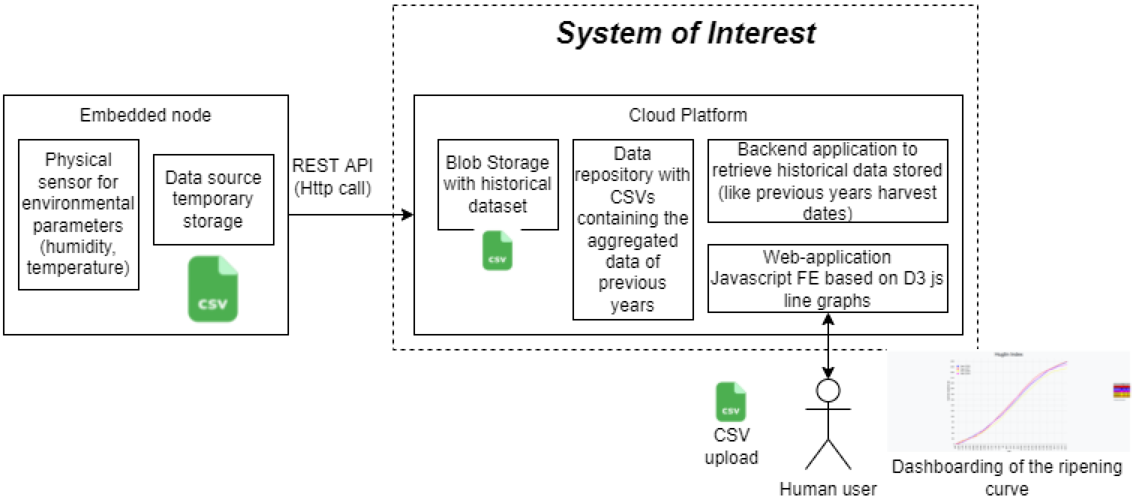

Figure 5: Physical architecture for use case 1

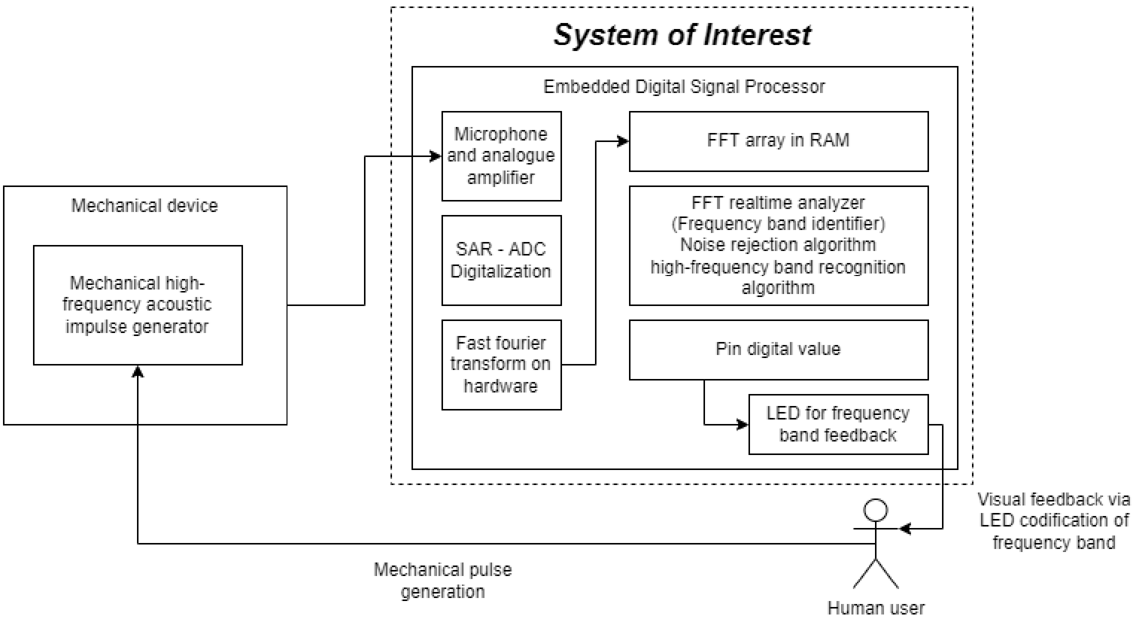

Figure 6: Physical architecture for use case 2

3.5 Detailed design principles

For the first use case the relevant physical modules from a detailed design point of view are:

3.5.1 UC1: Web application design

Hosted on an Apache web server, the web application was developed in JavaScript language with the support of the Bootstrap framework [32] as regards the user interface (UI) style and the layout of front-end and the D3 js open source library [33] for the implementation of dashboards (line charts). For the development, the creation of unit tests and the debugging of the application, Microsoft Visual Studio Code was used as IDE, a free application in strong diffusion thanks to the numerous plug-ins it offers for compatibility with different platforms and different programming languages.

In the web application are present:

- the upload module for saving new datasets in the data repository (Figure 8)
- the Huglin Index graph where it is possible to visualize the complete Huglin curves produced with the dataset of the IoT nodes (Figure 7). Trends are available both for previous years and the current year (with the last uploaded dataset) in order to compare with previous years trends (Figure 9).
- the harvest dates of previous years, to support decisions regarding the current year by comparing the data with the trend of the Huglin index (Figure 10).

```
"Date & Time","Temp - °C","High Temp - °C","Low Temp - °C","Hum - %","Dew Point - °C","Wet Bulb - °C","Wind Speed - km/h","Wind
"1/1/19 00:00","-1,4","-1,2","-1,5","94,0","-2,3","--","0,0","--","0,00","0,0","--","-1,4","-1,5","-1,5","1026,8","0,0","0,0","0,
"1/1/19 00:30","-1,8","-1,4","-1,8","94,0","-2,7","--","0,0","--","0,00","0,0","--","-1,8","-1,9","-1,9","1027,1","0,0","0,0","0,
"1/1/19 01:00","-1,7","-1,6","-2,1","95,0","-2,4","--","0,0","--","0,00","0,0","--","-1,7","-1,7","-1,7","1026,6","0,0","0,0","0,
"1/1/19 01:30","-1,6","-1,6","-1,9","95,0","-2,3","--","0,0","--","0,00","0,0","--","-1,6","-1,7","-1,7","1026,9","0,0","0,0","0,
"1/1/19 02:00","-1,8","-1,3","-1,8","94,0","-2,6","--","0,0","--","0,00","0,0","--","-1,8","-1,8","-1,8","1027,4","0,0","0,0","0,
"1/1/19 02:30","-1,6","-1,6","-2,2","95,0","-2,3","--","0,0","--","0,00","0,0","--","-1,6","-1,6","-1,6","1027,6","0,0","0,0","0,
"1/1/19 03:00","-1,8","-1,5","-1,8","94,0","-2,7","--","0,0","--","0,00","0,0","--","-1,8","-1,9","-1,9","1027,8","0,0","0,0","0,
"1/1/19 03:30","-2,3","-1,8","-2,4","95,0","-3,0","--","0,0","--","0,00","0,0","--","-2,3","-2,3","-2,3","1027,6","0,0","0,0","0,
"1/1/19 04:00","-1,6","-1,6","-2,3","95,0","-2,3","--","0,0","--","0,00","0,0","--","-1,6","-1,6","-1,6","1027,7","0,0","0,0","0,
"1/1/19 04:30","-2,0","-1,6","-2,0","94,0","-2,8","--","0,0","--","0,00","0,0","--","-2,0","-2,1","-2,1","1027,7","0,0","0,0","0,
"1/1/19 05:00","-2,7","-2,0","-2,7","94,0","-3,5","--","0,0","--","0,00","0,0","--","-2,7","-2,7","-2,7","1027,5","0,0","0,0","0,
"1/1/19 05:30","-2,7","-2,5","-2,7","94,0","-3,6","--","0,0","--","0,00","0,0","--","-2,7","-2,8","-2,8","1027,3","0,0","0,0","0,
"1/1/19 06:00","-2,7","-2,4","-2,8","95,0","-3,4","--","0,0","--","0,00","0,0","--","-2,7","-2,7","-2,7","1028,0","0,0","0,0","0,
"1/1/19 06:30","-3,2","-2,7","-3,2","94,0","-4,1","--","0,0","--","0,00","0,0","--","-3,2","-3,3","-3,3","1028,3","0,0","0,0","0,
"1/1/19 07:00","-2,7","-2,7","-3,3","95,0","-3,4","--","0,0","--","0,00","0,0","--","-2,7","-2,8","-2,8","1028,1","0,0","0,0","0,
"1/1/19 07:30","-1,8","-1,8","-2,7","95,0","-2,5","--","0,0","--","0,00","0,0","--","-1,8","-1,9","-1,9","1028,2","0,0","0,0","0,
"1/1/19 08:00","-2,2","-1,8","-2,2","95,0","-2,9","--","0,0","--","0,00","0,0","--","-2,2","-2,2","-2,2","1028,5","0,0","0,0","0,
"1/1/19 08:30","-2,3","-2,2","-2,4","94,0","-3,2","--","0,0","--","0,00","0,0","--","-2,3","-2,4","-2,4","1028,8","0,0","0,0","0,
"1/1/19 09:00","-1,2","-1,2","-2,4","95,0","-1,9","--","0,0","--","0,00","0,0","--","-1,2","-1,2","-1,2","1028,6","0,0","0,0","0,
"1/1/19 09:30","1,3","1,3","-1,2","96,0","0,8","--","4,8","NNE","2,41","12,9","NNE","0,0","1,3","-0,0","1028,5","0,0","0,0","0,0
"1/1/19 10:00","1,6","1,6","1,3","95,0","0,9","--","1,6","NNE","0,80","8,0","NNE","1,6","1,6","1,6","1028,4","0,0","0,0","0,00","
"1/1/19 10:30","1,8","1,8","1,6","95,0","1,1","--","4,8","NE","2,41","14,5","NE","0,6","1,8","0,6","1028,3","0,0","0,0","0,00","
"1/1/19 11:00","2,1","2,1","1,8","95,0","1,4","--","6,4","NE","3,22","16,1","NE","0,3","2,1","0,3","1027,9","0,0","0,0","0,00","
```

Figure 7: CSV dataset produced by IoT nodes

Figure 8: Detail in the UI for importing a new dataset and choosing the ones to be visualized

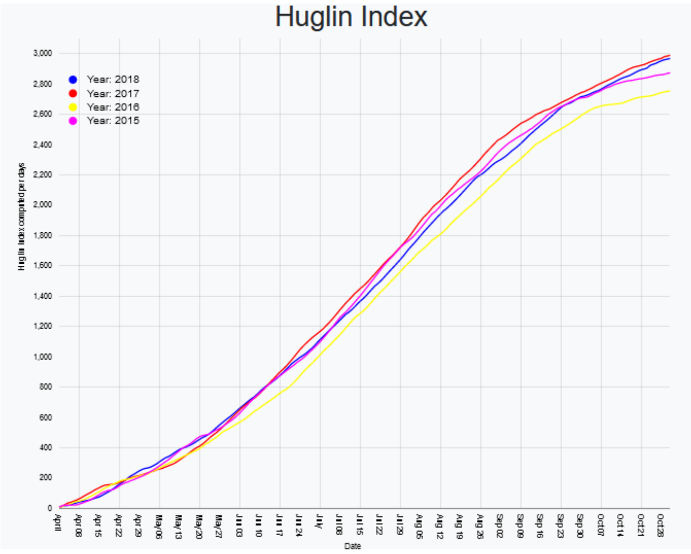

Figure 9: Huglin Index graph in the web application

Harvest date	Babo	A/T
16/08/2017	17.8	4.3
29/08/2018	14.6	5.6
01/09/2015	15.6	5.4
02/09/2016	16.3	6.3
05/09/2013	17.8	6.4

Figure 10: Harvest date table in the web application

3.5.2 UC1: Algorithm design

In order to implement the dashboard for supporting the agricultural enterprise in planning the harvest days, a literature review was carried out on the analytical functions most used for the estimation of ripeness based on the environmental parameters available from the

sensor node. The choice of the indicator was performed and validated with domain specialists.
The selected index is the Huglin Index [34], a bioclimatic heat index for vineyards.

$$HI = \sum_{1st\ April}^{30\ Sept} \frac{(T_{mean} - 10) + (T_{min} - 10)}{2} \cdot k \qquad [Eq1]$$

In the mathematical equation [Eq1] the index "k" represents a corrective coefficient to estimate the length of the day and, for this reason, it is a parameter that depends on the latitude of the region, [35].

For the second use case, the relevant physical modules for a detailed description are:

3.5.3 UC2: FFT module

Due to the non-functional requirement of having real-time signal processing application hosted on a digital signal processor (DSP) board and with maximum allowable latency of 1.5 seconds, it was decided not to implement a machine learning classifier, but a time series analyser based on the Fast Fourier transform (FFT).
The Fast Fourier Transform is used as an implementation of Discrete Fourier Transform: this is a necessary step since the system used to elaborate the signal is an integrated electronic circuit. This type of devices (currently the most popular in the field of acoustic signal processing) cannot work with analogue values: the sampling and discretization phases are mandatory steps before applying any transformation and logic with the algorithm.
According to the Nyquist-Shannon theorem, the signal can be correctly reconstructed (i.e. avoiding aliasing) after sampling process up to the f_{max} frequency, given a sampling frequency (f_s) that guarantees:

$$f_s > 2\ f_{max} \qquad [Eq2]$$

The Discrete Fourier Transform [36] is defined as

$$X(f) = \sum_{n=-\infty}^{+\infty} x[n]e^{-i2\pi f n T_s} \qquad [Eq3]$$

And can be written as

$$X(k) = \sum_{n=0}^{N-1} x[n] e^{-i2\pi n\frac{k}{N}} \text{ , k = 0,1,2,…, N-1} \qquad \text{[Eq4]}$$

Where in [Eq3] and [Eq4], x[n] is the time series of the discrete signal and T_s is the time interval between one sample and the next.

3.5.4 UC2: FFT real-time analyser

The scope of the algorithm designed for the prototype was to:

- Recognize the frequency band of the acoustic pulse
- Reject the frequencies produced by random noise
- Recognize the acoustic pulse even with an overlapping random noise

The complexity in recognizing the frequency band is that the waveform is not constant and uniform, as it is generated by human action (application of pressure) on a mechanical device that acts as a sound source. The main challenge was to design an algorithm with the right compromise between the flexibility in the recognition of the pulse frequency band and the noise rejection.

The algorithm is based on an evaluation function acting on frequency intervals (also called bins) with the objective to aggregate in a single metric the different characteristics that a high frequency pulse should have:

$$Ev(k) = \alpha \cdot A(k) + \beta \cdot S(k) + \gamma \cdot D(k) \qquad \text{[Eq5]}$$

In [Eq5] A(k) represents the normalized FFT, S(k) the “sparsity” function designed to null adjacent comparable frequencies, since they are not expected in a defined and mono-frequency pulse, D(k) is the norm of the difference between two consecutives FFT arrays.

The algorithm recognizes a frequency bin if the maximum element in the array of [Eq5] is greater of a threshold (to be adjusted according to the environment where the system is deployed).

The rational for this choice is that:

- The largest amplitude is expected concentrated in a single frequency bin of the spectrum for a pulse and not randomly distributed as for noise.
- The first time span in which the FFT is applied and containing the initial part of the waveform of the sound pulse is expected to have significant amplitude values, highlighting the pulse discontinuity.

For the realization of the prototype a Texas Instruments TMS320C5515 microcontroller, a 16-bit DSP, was used. This device features a hardware accelerator for calculating the Fast Fourier Transform. For the software programming of this device the IDE Code Composer Studio was used, exploiting the integration libraries between this IDE and the device offered by Texas Instruments.

3.6 Verification and Validation

An important phase to establish the technical success of the prototypes was the system integration and the execution of verification and validation activities.

In both cases it was decided to follow the Agile guidelines which divide the technical verification phase carried out internally by technology specialists and testers, from the validation phase generally performed with real users or company representatives. The first activities were mapped as test user stories, while the validation mapped as recurrent activity during the system demos with research stakeholders.

For the first use case, the verification phase was carried out by dividing and parallelizing the three main modules present in the web application:

1. Import of new CSV dataset
2. Verification of the graph
3. Previous datasets retrieval from the WEB server data storage module

This choice simplified end-to-end integration and verification, preventing bugs on unit level, difficult to find during system-level verification activities.

Verification ID	Module	Description	Goal
Ver1	Web application dataset import module	1. Only CSV datasets are accepted by the import module	Check the functionality of the module and the possibility of

		2. Only CSVs with recognized contract (fields, datatypes and separators) are accepted	avoiding the propagation of problems by filtering incorrect datasets
Ver2	Web application dashboard module	1. Previously loaded datasets are recalled and displayed in the Huglin index chart and in the harvest date matrix 2. Partial datasets about current year are visualized in the Huglin chart and compared with pre-computed values in excel using Huglin equation 3. The dashboard is responsive design compliant, ready to be used with laptops, tablets and smartphones using the most popular web browsers (Firefox, Chrome, Edge, Safari)	Check the main functions of the module, also considering the reactivity for different types of client devices
Ver3	Data storage Module	1. Check the persistency of already uploaded datasets 2. Verify that the dataset is updated if a newer	Verify the ability to store and update internal datasets

		version of the same dataset is loaded	

The advantage of this type of Verification methodology, driven by the physical modules tested before full system integration, is the parallelization of verification activities that optimize:

- The integration time thanks to early detection of unit problems
- The performance issue identification
- The granular verification, difficult to cover with same accuracy performing only end-to-end verification on system level

Another advantage of the modular approach in design and testing is the consequent use of accurate version and configuration management.
For the second use case, we organized the verification activities checking these features:

Verification ID	Module	Description	Goal
Ver1	Noise rejection algorithm	1. Frequencies produced by sources of random noise (coins, hook shots, keys etc) must be filtered	Check the functionality of the random noise filter
Ver2	High frequency band recognition algorithm	1. Sound pulses with fundamental frequency in a frequency bin of interest shall be recognized and a pre-assigned pin activated 2. Sound pulses with fundamental frequency out of frequency bins of interest shall be rejected	Check the functionality of the sound pulse classification

A confusion matrix was used to document and evaluate the output of the technical verification activities, which is very often used to evaluate the maturity of the classification algorithms (Figure 11). The confusion matrix is used to compare the actual results of the classification with the expected results in order to define the technical readiness of the algorithm. The possible system outputs following an input to be recognized are recorded on the vertical and horizontal axis of the matrix. The outputs on the vertical axis represent the expected outputs and therefore defined as "true", while the outputs on the horizontal axis represent the classifications made by the system and therefore defined as "predicted". The goal of the tool is to display how many times the system recognizes the input correctly. From what has been said, it is clear that a good classification system must tend towards a diagonal type matrix.

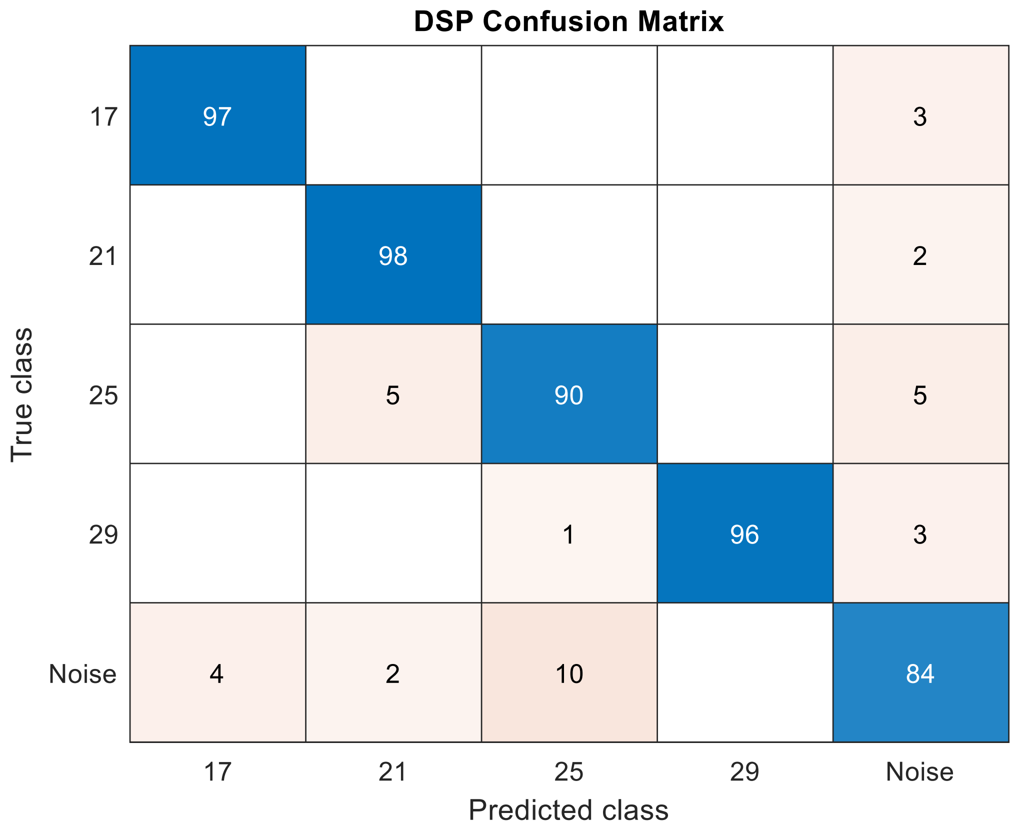

Figure 11: Confusion Matrix to evaluate the reliability of classification

3.7 Conclusions

With these two use cases chosen as samples for the area “Signal and Data processing and visualization systems” it was possible to support SMEs involved in the initiative in understanding:

1. How to leverage on already existing enablers to empower their business using I4.0 technologies.

2. How to improve the modularization and decentralization in the design that can facilitate the development and verification parallelization, the configuration management of the overall system, the focus in terms of technology specialization driven by the different functionalities avoiding a complex monolithic approach.
3. How to formalize the system requirements derived by stakeholder needs to drive and cover with a complete test plan the system integration.

Chapter 4: Digital Twin and IoT multi-agent systems

The second area of I4.0 investigated concerned distributed systems for “Digital twin and IoT multi-agent systems”.

The selected proof-of-concepts for this area came from different industrial sectors: air quality monitoring in manufacturing plant and consumable monitoring for public and large buildings (like airports, hospitals, exhibition halls etc.). We decided to select these use cases because centralized monitoring of distributed systems is crucial to understand the overall behaviour of complex systems.

- Use case 3 - “BIM Plant Digital Twin updated by MQTT-based IoT WSN”: distributed data collection and monitoring system based on MQTT protocol, to create a real-time updated model for monitoring the air quality parameters of a production plant.
- Use case 4 - “BLE-based WSN for energy optimized applications with synchronized deep sleep phases”: wireless sensor network (WSN) application based on low energy protocol and synchronized deep sleep phase of the devices to optimize the battery duration of the sensor nodes.

4.1 State of the art

In recent years, multi-agent systems have once again become a very popular research topic in the field of industrial automation and IT, due to the diffusion of IoT systems and decentralized control systems. Using decentralized policies, it is possible to implement independent monitoring strategies by exploiting the communication between resources, [37].

Multi-agent systems represent both a design pattern to realize service-oriented architectures and a set of software technologies, applied in automation engineering for the orchestration of distributed controllers and embedded devices and in software engineering for the design of distributed modules in IoT systems and microservice-oriented architectures [6].

Michael Wooldridge writes “An agent is a computer system that is capable of independent (autonomous) action on behalf of its user or owner. A multi-agent system is one that consists of a number of agents, which interact with one-another”, [38].

When multi-agent systems are used to achieve the deep integration of computing, communication and process control with humans in the loop, they are also defined Cyber-physical systems, [39] [40].

In the development of distributed systems, careful metadata design becomes essential in order to allow all the systems involved to understand the data published on the distributed platform: this problem is also known as “ontology design”, a formal software-processable taxonomy [41]. The problem of ontology definition was extensively investigated by FIPA (The Foundation for Intelligent Physical Agents) [42] in the domain of Service-oriented architectures and industrial agent systems for production systems (mainly for planning and scheduling control) and thought for agents designed for computer-hosted applications. A standardized approach for the design of the ontology in the new Industrial Internet-of-things (IIoT) systems is not yet clearly defined, when not directly addressing high-level tasks such as production planning, scheduling and monitoring, like in the use case for the air quality monitoring or other applications where the multi-agent software is also deployed on the edge.

Currently there are consolidated software technologies to develop software agents like Jason, Jade, Jadex etc but they are frameworks designed primarily for computers and it seems that their diffusion has slowed down in recent years.

The current trend of software engineering is moving towards new applications oriented to microservices, using containerization technologies and asynchronous message queueing protocols, like MQTT, AMQP etc. These protocols are suitable both to implement data exchange and integration among different software modules like microservices, but also to stream data from IoT devices like sensor nodes, also called smart nodes.

We define “smart node” the embedded device:

- endowed of physical sensors to collect information from the environment,
- equipped with software modules capable to filter and apply simple algorithms for autonomous actions,
- endowed of software interfaces with high-level processing systems for data analytics applications [39].

Global research in this field is intensive as smart nodes can be used in many fields: buildings, healthcare, manufacturing, agriculture, transport; more than 27 billion IoT devices are expected to be active in 2022 [43].

4.2 Problem statement: Use cases and Stakeholder needs

In the third use case, the focus is on the design of a distributed wireless sensor network using typical internet protocols to collect real-time samples about air quality parameters of the indoor environment of a production plant (or a civil building). The scope was to interface it with a Building Information Model (BIM) to create a real-time digital twin of the infrastructure, to understand the air-quality evolution over time for a continuous verification of a comfortable working environment and providing a tool for timely action in case of deviation from standard parameters. It is indeed an important research topic the creation of collaborative systems: on the one hand, cooperation can be understood as support in practical tasks for the purpose of production; on the other hand, the possibility of designing an automated work environment capable of monitoring, analysing and identifying anomalies in real-time to alert human operators, is certainly at the basis of the creation of a collaborative work environment, capable of preserving and improving psychophysical well-being.

In the fourth use case, however, it is necessary to design a wireless sensor network with lower energy consumption compared to the previous use case, in order to monitor the availability of consumable resources in large buildings and collect the values in real-time on a cloud platform to provide optimized services.

Stakeholder need ID	Stakeholder need definition
SN1	The wireless sensor network shall collect new samples of air quality parameters with a sample time lower than one minute and update the values of the virtualized elements of a Building Information Modelling System. The system shall also allow the possibility of setting rules on the time series of sampled data in order to send automatic alerts via email or trigger possible actions on the smart nodes.
SN2	The wireless sensor network shall collect updated values from sensor nodes and transfer them to a cloud platform with the scope of minimizing battery consumption of the smart nodes during the network uptime, considering both the active time when sampling the data and the idle time between one sample and the successive one.

4.3 System Requirements elicitation

Requirements elicitation for use case 3

System Requirement ID	System Requirement definition
SR1	The wireless sensor network shall be capable to collect data via microcontrollers endowed of sensors and a Wi-Fi module: the microcontroller requirement is due to lower costs of the embedded systems compared to a single-board computer
SR2	The wireless sensor network shall be organized with gateway nodes collecting data from sensor nodes and pre-filtering them before publishing in a centralized server with BIM Model
SR3	The wireless gateway shall host a broker to collect the data from sensor nodes, enabling two-way communication with sensor nodes in order to collect new samples and cascade control signals
SR4	The wireless gateway shall transfer the updated values of sensor nodes to a central server containing the BIM model connecting to a high-level broker
SR5	The software agent updating BIM model shall store the values corresponding to the virtualized twin of the sensor in the CAD model in a Industry Foundation Classes (IFC) file. IFC is a text file with standard format used by all opensource and commercial BIM modelers
SR6	The gateway collecting data from sensor nodes shall store historical data with a configurable rotation policy on a SQL database server hosted on the gateway
SR7	The gateway node shall expose a web site used to monitor the trend of the sensor data over time, both with real-time dashboards and historical trend charts
SR8	The gateway node shall have a configuration file where to specify connection string to the server of BIM model, the update rate to require new data to sensor nodes, logical inequality with warning thresholds for each physical parameter within the WSN (used to

	generate automatic alarms), the list of email accounts to notify in case of an alarm
SR9	The gateway node must have an alarm manager module to send an email to the mailing list in case the threshold value is reached and to notify the alarm to the embedded device
SR10	The system shall have an ontology-based communication model in order to recognize the sender and the type of data that are communicated leveraging on MQTT topic structure

Requirements elicitation for use case 4

System Requirement ID	System Requirement definition
SR1	The wireless sensor network must also be usable in environments where Internet or Wi-Fi coverage is not guaranteed
SR2	The wireless sensor network shall optimize the energy consumption during the idle phases where it is not required to provide updated values from the sensors, being the system aimed for low-frequency updating applications
SR3	The wireless sensor network shall be easy to install in terms of number of steps required for enabling data flow from each node: the steps shall be the same for each node to lower the risk of misconfiguration during commissioning and shall be done from mobile devices (smartphone, tablet etc)
SR4	The wireless sensor network shall guarantee the communication with a node to cloud infrastructure for data collection
SR5	The wireless sensor network shall be implemented with embedded microcontroller devices thanks to low costs
SR6	The wireless sensor network physical installation shall be easy and flexible to change, so it is not in scope a wired power supply but only battery-powered devices

4.4 Definition of functional architecture and physical architecture

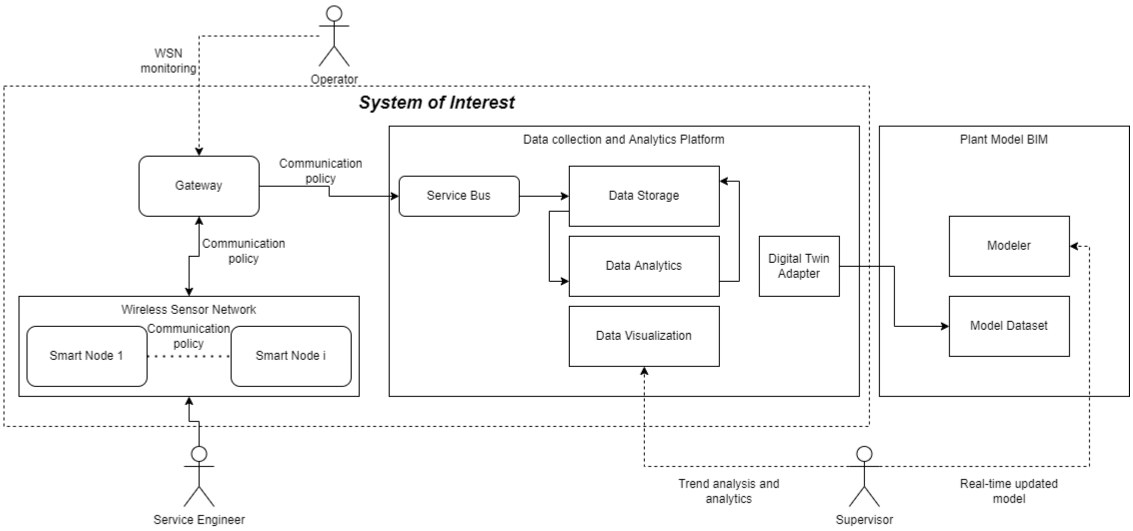

Figure 12: Functional Architecture for Digital Twin and IoT multi-agent systems

The functional architecture is common for the two use cases, since they mainly differ for non-functional requirements on energy consumption and internet availability.

The third use case (Figure 13) is based on the need of collecting data from distributed wireless sensor networks, centralizing the data in a data analytics platform for historical analysis and visualization, and on the capability of real-time update of a production plant model, that can be monitored by human operators to understand the system and to support rapid decision making.

The fourth use case (Figure 15) instead is more oriented in understanding, with different protocols and control policies, how to optimize the efficiency of the nodes during inactive phases, where it is not requested to provide updated values to the analytic platform.

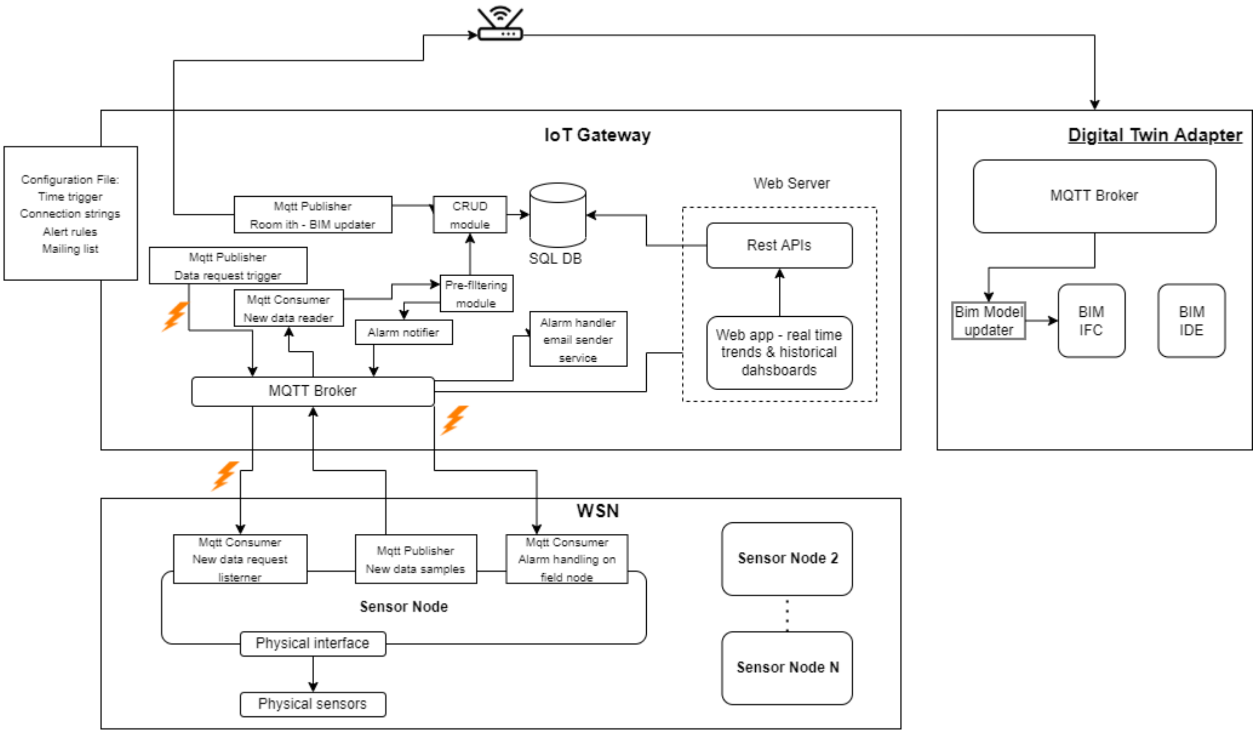

Figure 13: Physical architecture for use case 3

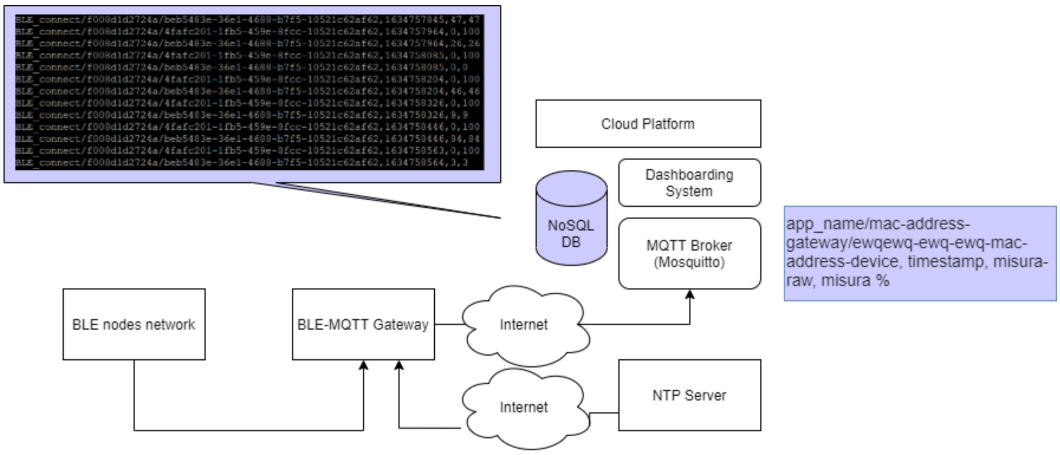

Figure 14: End-to-end solution with BLE WSN

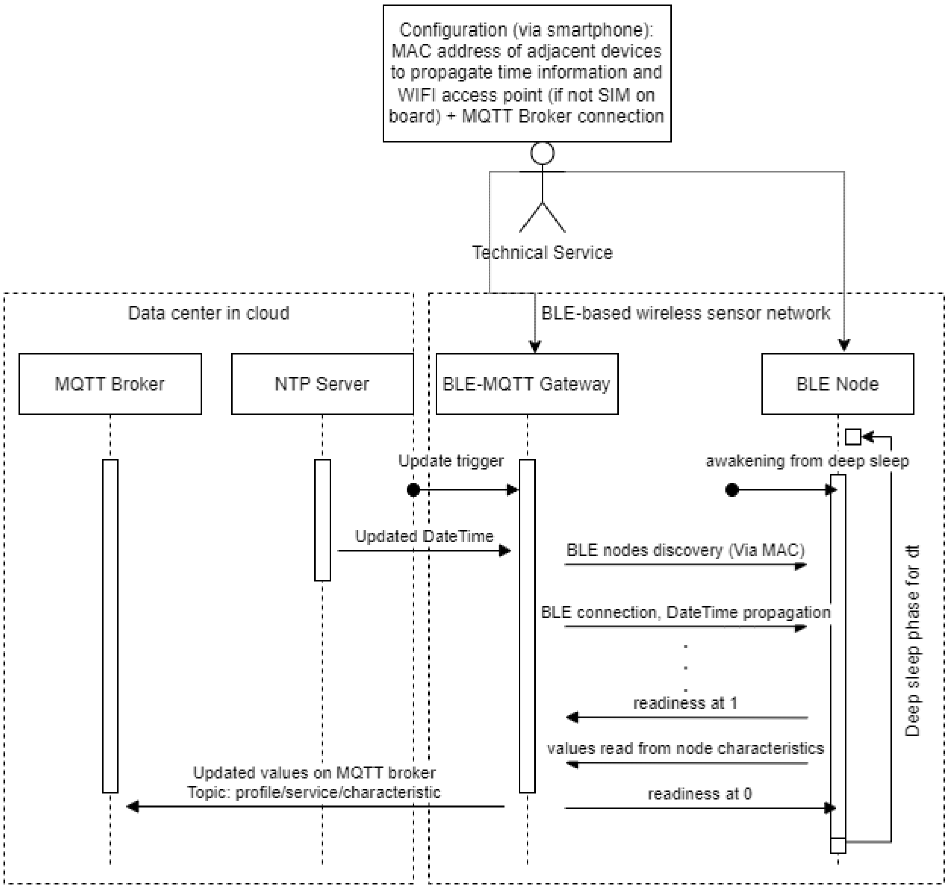

Figure 15: Physical architecture for use case 4

4.5 Detailed design principles

For the third use case the relevant physical modules from a detailed design point of view are:

4.5.1 UC3: Wireless sensor network using MQTT

Wireless sensor networks are IoT applications where sensor nodes can collect and exchange data using internet protocols to implement communication policies, but not necessarily to move data on the internet, potentially deployable also in the local network of the plant/infrastructure of interest [44]. In the design of the WSN we kept into consideration the guidelines of OpenFog Consortium, adopting a multi-layered IoT infrastructure [45] with:

- Edge devices: implemented using ESP8266 boards with a Wi-Fi module
- Fog nodes: implemented using Raspberry PI boards for simple analytics, alarm management and exposing a web server for data visualization in a web application. Real-time operations and requests are managed in this intermediate layer and

generally are computationally simpler operations than the ones performed in the data centre node.

- Data centre: a server with higher computational resources, where it is also hosted the BIM model to realize the digital twin, updating in real-time the values coming from the fog nodes.

The communication among these 3 layers was implemented via an event-bus orchestrator, using the choreography architectural pattern. We decided to use an asynchronous pattern leveraging on peer-to-peer data exchange, because it is aligned with the idea of communication platform described by FIPA guidelines exceeding the traditional client/server communication (more evident in modern contracts like RESTful APIs) and moving towards a service-oriented architecture (SOA) design pattern.
To implement the peer-to-peer platform we used the Message Queue Telemetry Transport (MQTT) protocol. The reasons behind this choice are:

- It is an ISO standard protocol (ISO/IEC PRF20922), [46]
- There are implementations both for object-oriented software development, agent-oriented software development and embedded devices programming languages based on microcontroller without an operating system (platform independent)
- It is simple and used for lightweight IoT applications
- It provides cybersecurity support in terms of authentication and authorization to access the message broker
- It is a common standard in different industries: manufacturing, automotive, smart buildings, logistics, transportation
- It provides the possibility of one-to-many communication, using the topic that can be used to design the system ontology, since it represents the identifier of message entity
- It decouples the modules in the system with event-triggered pattern, that creates loosely coupled modules, improving the overall failure containment in case of local unavailability of specific software modules.

The MQTT communication is established between:

- Edge devices (the ESP8266 controllers) having sensors connected to sample air quality parameters and the fog node (a Raspberry PI, representing the IoT gateway for each room or area of the production plant).

- Fog nodes and centralized data centre (the server hosting the data analytics and digitalized plant model), where each IoT gateway publishes the set of updated values for updating the plant digital twin.

MQTT messages are used both as a trigger and as a payload, in fact the IoT gateway initiates the request for updated values by sending trigger messages to the broker and the IoT nodes send subsequent messages to the MQTT broker with updated values for the sensors they host.

Thanks to the threshold rules defined in a configuration file hosted on each gateway, a data analytics module on the Raspberry PI is capable to identify anomalies, writing a triggering message of alert on the MQTT broker that causes the Alarm handler service to send automatic alerts to the mailing list that can be defined for each area of the plant.

As regards the development tools, both for the sensor nodes and for the software development on the IoT gateway we used Microsoft Visual Studio Code as IDE, using the integration plug-ins for Arduino (for the purpose of development in language C ++ of microcontroller platforms) and Python extensions for software development for the IoT gateway. As regards the MQTT clients and servers, the open source projects Eclipse Mosquitto (server) and Eclipse Paho (client) were used, [47] [48].

4.5.2 UC3: Ontology design

In the distributed systems (or multi-agent systems), the system ontology is the set of metadata that represent the system information model used to interpret the information exchanged among the different software units, in this specific use case the data stream arriving on the MQTT brokers. Since the ontology is not only used to interpret the data coming from the edge devices, but also to map them into the data structure of the BIM model to achieve the real-time updated model, we decided to use the topic of the MQTT message bus to organize the representation in an object-oriented way.

In the specific implementation the topic structure is:

$$\{room\ gateway\}\_\{parameter\}\_\{parameter\ instance\}$$

where:

- {room gateway} is used to represent the physical area covered by the IoT gateway in terms of data collection from edge devices,
- {parameter} is used to understand the physical measurement coming from the environment (humidity, pressure, etc),

- {parameter instance} is used to distinguish the specific edge node, in case in a plant area there are more nodes providing the same information.

This ontology is used by the BIM model updater module to search for corresponding items in the IFC file and modify the parameters (Figure 16).

```
IFCPROPERTYSINGLEVALUE('IoT-1_HUMIDITY_LAST-VALUE',$,IFCREAL(52.88),$);
IFCPROPERTYSINGLEVALUE('IoT-1_IAQ-INDEX_LAST-VALUE',$,IFCREAL(25.0),$);
IFCPROPERTYSINGLEVALUE('IoT-1_PRESSURE_LAST-VALUE',$,IFCREAL(100828.0),$);
IFCPROPERTYSINGLEVALUE('IoT-1_TEMPERATURE_LAST-VALUE',$,IFCREAL(19.27),$);
IFCPROPERTYSINGLEVALUE('IoT-1_URL',$,http://localhost/timeseries_frontend/,$);
```

Figure 16: IFC items representing a sensor node with values from IoT system

4.5.3 UC3: Data filtering and Alarm management

In the Fog node (the IoT gateway) two services that have the purpose to pre-filter and store the updated data coming from the edge nodes in the SQL database (MySQL technology was chosen) are present. This is not the only purpose of the filtering module, but it is also capable to detect anomalies with respect the rules defined for each parameter in the configuration file of the gateway, in order to write an alarm message on the MQTT broker to notify all the nodes interested in the anomalies management to proceed with their tasks.

This message is used by edge nodes in order to perform physical actions if needed (acoustic alarms, or visual feedbacks for operators in the plant), but is also collected by a service called “Alarm handler” that in our proof of concept sends automatically an email to the mailing list in scope of the area monitored with the specific gateway.

4.5.4 UC3: Dashboard system

The dashboard system (Figure 17) is implemented in form of a web application hosted on the room gateways. The dashboard system is designed to perform:

- Real-time graphs with the trend of the measurements of the available sensor nodes publishing the data on the broker of the gateway.
- Historical graphs, using the data stored in the SQL database hosted in the gateway node and retrieved via a Rest API.

In both graphs the system computes statistical parameters useful to analyse the trends, like the average value and the standard deviation, computed as normalized parameters depending on the time persistency of the values, using time weights $w_i = \frac{t_i - t_{i-1}}{t_{n-1} - t_0}$. The time weighted parameters are necessary since the pre-filtering module has the scope to extend the duration time of a specific value into the SQL database, in case a parameter does not change in the successive iteration.

$$\mu = \sum_{i=1}^{n-1} w_i \, x_i \qquad \text{[Eq6]}$$

$$\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n-1} w_i \, (x_i - \mu)^2} \qquad \text{[Eq7]}$$

The dashboard system is exposed via a web server hosted on the gateway; it is reachable via a QR code printed on each IoT gateway (encoding the URL to reach the endpoint) or via a link available in the BIM model (one parameter sent by each room gateway to the BIM model updater module is the URL to access the web application in the intranet of the plant).

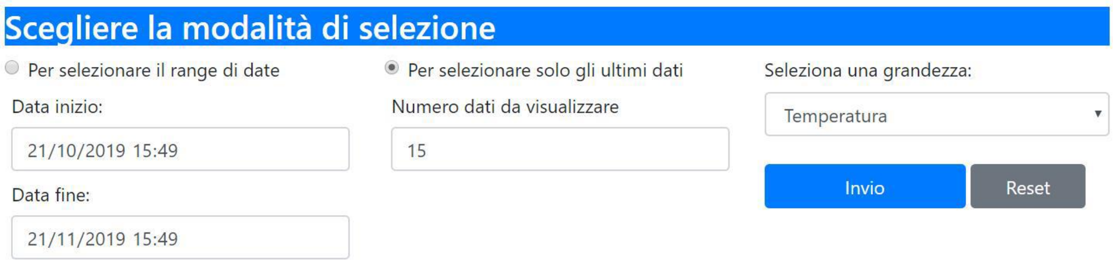

Figure 17: Dashboard selection form

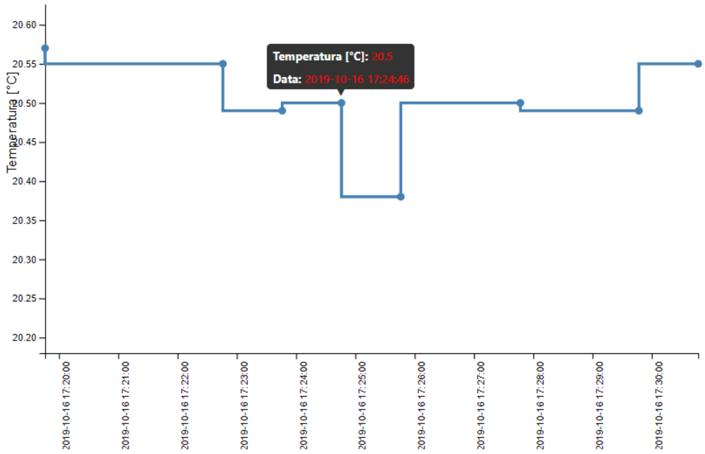

Figure 18: Detail of the time series graph

For the fourth use case the relevant physical modules from a detailed design point of view are:

4.5.5 UC4: BLE network

We decided to implement the Wireless sensor network considering as low energy consumption technology the Bluetooth Low Energy (BLE). The choice is because most mobile devices are currently delivered already with Bluetooth module and, according to official forecasts of Bluetooth Special Interest Group (SIG), there will be a linear growth of Bluetooth-enabled peripheral devices, reaching 6.4 billion of annual shipments in 2025 [49]. This protocol is mainly used for beaconing and indoor positioning systems, but it starts to be used also for smart home automation thanks to its power efficiency compared with Wi-Fi and Zigbee [50].

BLE is based on two different application layers with respect Bluetooth classic:

- Generic Attribute (GATT): for connection-based data exchange between a client node and a server node.
- Generic Access Profile (GAP): for advertising mechanism between central node and peripheral node.

Using the GATT service structure, it is possible to design multi-agent oriented WSN, since the structure is service-oriented. This structure is used as a client-server protocol when the connectivity between two BLE devices is established, and it is conceptually like an API [51]. The possible interactions for the client are defined by the interaction property specified for each characteristic exposed by the BLE Server.

The GATT data structure is composed by 3 layers (Figure 19):

- Profile: it represents the device owning the GATT data structure (the server application).
- Service: it groups a set of characteristics (variables exposed by the service) if they are linked semantically.
- Characteristic: it is the real mean to enable BLE data exchange when a connection between client device and server device is established. Each BLE characteristic provides a descriptor (in terms of Universally Unique identifier – UUID, [52]), an interaction property (broadcast, write, read, notify etc.) and the value field.

The idea is to use the GATT data structure to organize the ontological model of the multi-agent BLE system, where the service identifier is represented by the Service UUID to be unique and distinguishing the possible services present in the WSN, similarly for the characteristic UUID to semantically normalize the content of the different characteristics available in the distributed system and the type of interaction allowed during the client-server connection.
UUID for BLE protocol is a 128-bit unique number. Since BLE protocol is already used for standard applications, it is important to secure that the ontology definition is not overlapping the standard UUIDs recommended by Bluetooth SIG, [53].

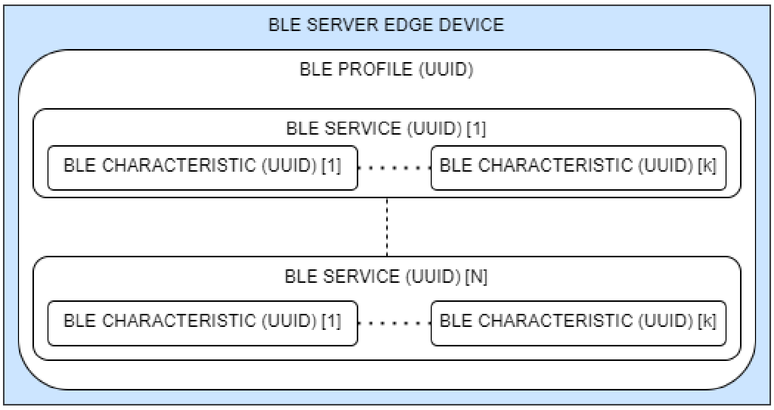

Figure 19: BLE GATT Data structure

4.5.6 UC4: BLE node design principles and BLE-MQTT Gateway

In order to implement a BLE network capable at the same time to move the data towards the BLE-MQTT gateway (in order to publish them on MQTT broker hosted on cloud) and to guarantee optimized energy consumption, we designed edge devices capable to:

- Implement both the behaviour of BLE server and BLE client, so that they can propagate information in two directions of the possible graph configured during commissioning phase
- Propagate an updated synchronization time coming from the BLE gateway (in communication with a Network Time Protocol (NTP) Server)
- Synchronize the data request phase, the data providing phase and the deep sleep phase, having the gateway the role of main node of the multi-agent system aimed to enable the data collection

During the commissioning phase, the field service technician configures the BLE graph by setting on each BLE node the MAC addresses of the downstream BLE nodes with which to connect to collect the data to be transferred to the cloud platform. On each device, a BLE service is reserved for the configuration procedure, setting the list of MAC addresses. The last node to be configured is the BLE-MQTT gateway, so the configuration is performed starting from the node with highest hop level (i.e., the highest distance in terms of node connectivity in the graph to reach the gateway). To enable the connectivity with internet, the BLE-MQTT gateway is configured to connect to a Wi-Fi access point if not equipped with Sim card module; then the MQTT broker connection parameters and the NTP (Network Time Protocol) server connection parameters are also to be configured.

The data streaming to MQTT broker is done with the following ontological schema:

{BLE_WNS_ID}/{mac-address-gateway}/ewqewq-ewq-ewq-{mac-address-ble-device}, {timestamp}, {raw-measurement}, {%-measurement}

Example:

BLE_connect/f008d1d2724a/beb5483e-36e1-4688-b7f5-10521c62af62,1634757845,47,47

To implement both the BLE nodes and the BLE-MQTT gateway, the ESP32 microcontroller was used and programmed with Arduino IDE using C++ programming language and available libraries for BLE stack implemented by Espressif.

4.5.6 UC4: BLE data propagation policy

After the configuration during the commissioning phase, the data propagation policy is achieved considering these sequential steps:

1. The BLE devices of the WSN, in a synchronized way, awake from deep sleep phase thanks to an Interrupt generated by an internal timer
2. The MQTT-IoT gateway synchronizes with NTP server for collecting an updated timestamp in UNIX format
3. The MQTT-IoT gateway cascades the updated time to downstream BLE nodes configured as MAC addresses on the reserved characteristic
4. The nodes, after receiving updated datetime from upstream nodes, switch behaviour from BLE server to BLE client since they need to propagate the datetime and start monitoring the readiness characteristic of downstream connected nodes
5. The downstream BLE nodes cascade the updated time until the highest hop layer is updated
6. Reversing the data flow (i.e. from highest hop to lowest), the sensor data values are updated and transferred thanks to the fact that the BLE nodes can set their own readiness characteristic to 1 for upstream nodes. Note: The readiness value is set to 1 by the BLE node when all the values are updated or when a threshold time has elapsed; it is used as a flag for upstream nodes to understand if the new values are ready or not.
7. The upstream BLE device on cycle time (configurable) connects to downstream nodes of interest to check the readiness characteristic (i.e. the characteristic that asserts that the updated values to be transferred on cloud are available). This data propagation policy is repeated in the same way between all the hop levels in the graph, until the updated values arrive to MQTT-IoT gateway. The BLE client after reading the downstream node values, set the readiness characteristic of the server to 0.
8. The BLE node that receives the readiness 0 from the upstream node enters the status of "deep sleep phase" and the time to sleep is computed (Figure 20) as
   Time_to_sleep = $[\text{BLE\_behaviour\_frequency}]^{-1}$ – Time(active_status)
9. The MQTT-IoT gateway reads and stores the updated values and writes them to MQTT broker.

10. Repeat the same cycle described in these steps.

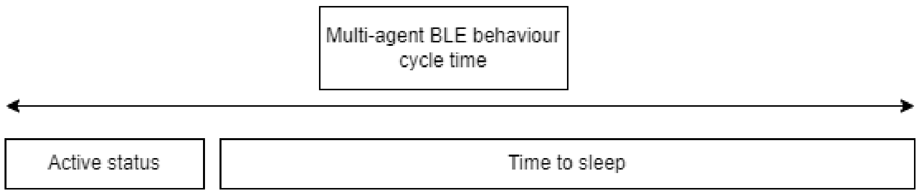

Figure 20: Cycle time of BLE distributed system

The main difference between the BLE-MQTT gateway and the BLE nodes is that the first one is always active, while the BLE nodes are designed to optimize energy consumption alternating active status phases to deep sleep phases. For this reason, it is important during the commissioning phase to set the same value of BLE_behaviour_frequency by means a characteristic exposed by BLE devices, in order to create the common "sleeping" behaviour for all the nodes of the platform.

The overall system can also deal with localized failures, since each node has an internal threshold that limits the amount of time to wait for updated values coming from the downstream nodes: indeed, in case of failure this time would be infinite. After the threshold is exceeded, the characteristic of readiness is set to -1 by the node so that the upstream node can know that the information is available but limited.

4.6 Verification and Validation

For the third use case it was performed a modular testing activity in laboratory, dividing the three main functional modules present in the platform:

1. Request of updated values and new values published on the MQTT broker by edge node
2. Verification of the filtering and storage of new values in the SQL DB
3. Previous datasets retrieved from the web app via REST API

Verification ID	Module	Description	Goal
Ver1	MQTT platform layer 1: edge node and fog node	1. Check that, when a scheduled job on the gateway requires an updated parameter to	Check the sequence of request and response

		edge nodes, a message is sent on the MQTT broker to the corresponding topic 2. Check that this message is received by the edge node with physical feedback (led) implemented for testing purpose 3. Check that the updated value is published on the MQTT broker (using a fake subscription to monitor the message) 4. Check that in case of unavailable node response, the gateway is fault-tolerant guaranteeing the service availability for the other edge nodes	between the room gateway and the room edge node.
Ver2	Alarm generation and handling	1. Check that in case an alarm rule defined in the configuration file of the gateway is activated by updated values, the alarm message is published on the MQTT broker 2. Check that the alarm handler module sends an	Check the alarm generation and management capabilities

		email to alert the configured mailing list 3. Check that the message is collected by the edge nodes interested in alarm management, with visual feedback (led)	
Ver3	Dashboard with webapp	1. Check that the graphs are correctly generated with the dataset stored in the SQL DB and compare the trend with a manually created graph using the same dataset 2. Check that the real-time graph is updating with values compared in parallel with the samples computed with other sensors 3. Check that historical data are correctly retrieved with datetime selection, comparing with manually executed SQL scripts on the same database 4. Check that mean and standard deviation are correctly computed comparing with manually computed values for sample trends	Check the correctness of dashboards and the usability of the selection features

		5. Check that the webapp is accessible also from the URL written in the IFC file of BIM model	
Ver4	BIM model updater	1. Check that the values inserted in the IFC module are always the last inserted in the DB of the room gateway 2. Check that the association between values of edge node, MQTT topic and IFC items are corresponding 3. Check that the unmapped object data in the IFC file does not create problems and that it is simply ignored by the software module	Check the ability to update the BIM model with updated values

The advantage of this type of verification methodology, guided by checking the physical modules before full system integration, is the parallelization of verification activities and better identification of the root cause in case of problems. Another benefit of the modular approach is the version and configuration management of the solution, which you can easily scale by adding new modules to add functionality over time.

For the fourth use case we performed these verification activities:

Verification ID	Module	Description	Goal
Ver1	BLE Network	1. Verify the data consistency when a BLE	Check the functionality of

		node reads and stores information from downstream node characteristic to internal characteristic 2. Verify the capability of updating the readiness characteristic to 1 when all the downstream nodes configured in MAC address list have updated the data and this information has been transferred to internal characteristics 3. Verify the capability of updating the readiness characteristic to -1 when the internal time threshold to wait for updated characteristics from downstream nodes is exceeded 4. Verify the capability of upstream node to set the readiness characteristic of downstream nodes to 0 when the propagation policy has been finished and the data stored on internal characteristics 5. Verify the capability of synchronized awakening	data collection in the BLE WSN

		is not lost during several iterations (checking all the data arriving on MQTT Broker)	
Ver2	BLE-MQTT gateway	1. Check the capability to propagate the updated time downloaded from NTP server 2. Check the capability to cascade the internal time of BLE-MQTT gateway if connectivity with NTP server is not available 3. Check the capability of publishing the updated data on MQTT broker, using a MQTT logger on the broker created for testing purposes	Check the functionality of publishing the data of BLE nodes on cloud using MQTT
Ver3	BLE Nodes configuration of MAC address list	1. Check that the configuration steps are performed following the same strategy on all the platform nodes, using an opensource application for Android and iOS (nRF) 2. Check that the BLE nodes to be added to the list of Mac addresses of the current node are reachable through the	Check the configuration strategy

		application used in the configuration phase to ensure the reachability of the nodes during the communication phase	
Ver4	Optimization of energy consumption	1. Check that the system has an energy efficiency improved with respect similar setup in case of traditional http communication and BLE communication without deep sleep phase. Note: during test activities in laboratory, we observed that the Wi-Fi connectivity to publish on a MQTT broker requires 10 seconds, while BLE connectivity 1-3 seconds. Moreover, it was observed a difference in terms of current consumption: during deep sleep phase 10 μA current consumption against the 30-68 mA when the device is in idle state. Thanks to the energy formula for electricity ($E = V \cdot I \cdot \Delta t$), it is possible to declare that this	Check the energy efficiency of the solution

		configuration of the network allows energy savings.	

It is worth to mention that in both the use cases a long run validation in laboratory was performed, without observing performance degradation in terms of data losses.
An important evaluation was performed on the configurability of the system, verifying that the system in the second use case can be installed and configured by operators not specialized in the BLE technology, simply using their smartphones to create the network of devices, configuring the MAC addresses in each device characteristic.

In future research it will be interesting a comparative analysis on the latency and energy performance of the different protocols in order to understand the differences between low energy consumption technologies. In such research it would also be necessary to explore the possibility of using the advertising mechanisms offered by BLE through GAP and try to define an optimal integration between low energy impact systems that use low level protocols and modern 5G cellular technologies that enable a high bandwidth and optimized latency.

3.7 Conclusions

With these two use cases chosen as samples for the area "Digital twin and IoT multi-agent systems" it was possible to support SMEs involved in the initiative in understanding:

1. How to design a wireless sensor network, structuring a common ontology for data collection and interpretation in centralized analytics platform.
2. How to choose different protocols and technologies for the embedded devices and data capturing/analytics modules, depending on the system requirements about power supply, flexible positioning etc.
3. How to structure an IoT system leveraging on the three layers recommended by the standard architectures: edge, fog, data centre.
4. How to create a digital twin using real-time data to create dashboards, automatic alerts and updating a digital plant model.

Chapter 5: Big Data and Decision Support Systems

The third research topic of I4.0 analysed during the digitalization initiative was for “Big Data and Decision Support Systems”.

The selected proof of concepts for this area come from manufacturing industry but they implement digitalization enablers in scope of different organizations: in the fifth use case the goal is to leverage on big data technologies to optimize the production efficiency of production lines with typical layout in ceramic industries. In the sixth use case the goal is to improve the quality of internal production and to support the troubleshooting of delivered products, leveraging on technical issue reports found on the field from a robotics and automation enterprise.

We decided to focus on these two use cases thanks to the opportunity to investigate the benefits and the challenges for Decision support systems based on Big Data applied in two different manufacturing contexts and with two different approaches: analytics and pre-defined metrics computation in the first case, data exploration and clustering in the second case.

- Use case 5 - “Analysis of the causes of downtime in ceramic plants”: decision support system based on MES and PLC data to identify the main causes of production stoppages on each equipment, to allow production optimization activities.
- Use case 6 - “Similarity analysis in unstructured texts for non-compliance reports”: decision support system based on free text reports, in order to identify similar descriptions grouped by equipment in scope.

5.1 State of the art

Big Data is one of the key technologies for smart industry, but it is a complex task to provide a unique definition. According to different researchers in the domain, we can define “Big Data” the applications and technologies described by three keywords, also known as the “Three V’s”: Velocity, Volume and Variety [54], [55].

- Velocity refers to the data rate at which the dataset is produced and the speed for their analysis.

- Volume refers to the magnitude of the data and the technological implication in their management, since the same size could imply very different data manipulation, storage and extraction logics (e.g. tabular data vs video data).
- Variety refers to different data sources in inputs (e.g. XMLs, CSVs, data streams in message queuing, JSONs, SQL data), the Data Types inside the dataset (text, video, images, structured data), the type of analytics to be applied for the analysis.

A Decision Support System (DSS) is a technical enabler for Industry4.0 and it is closely related to the domain of Big Data; we define the DSS as an analytics software program used to collect, harmonize and analyse data to extract and correlate information contained in the unstructured dataset. The final goal of this digital data pipeline is to support the decision-making process in manufacturing companies, [56]. A DSS software has the purpose to support human operator daily activities, generating data-driven services, that can be model-driven or data-driven [57].

The DSS systems are mainly divided into two categories:

- Traditional DSS: are based on historical data, without the ability to drive real-time actions.
- Modern DSS: are based on real-time E2E data pipelines, aimed to trigger immediate actions and real-time insight on the new data arriving in the dataset.

The support systems are not expected to take autonomous decisions excluding human operators from the decision-loop but, conversely, are designed to elaborate huge amount of raw data to provide issue notification to operators via automatic alerts and correlate the data presented in dashboards, quick to understand and easy to use.

The study of Decision Support Systems will be fundamental according to academia in the future research, since repetitive tasks will be delegated to autonomous mechatronic systems, having the human operator as central orchestrator and supervisor of a complex data-driven process, [58].

5.2 Problem statement: Use cases and Stakeholder needs

In both use cases a large dataset was already present, in the fifth use case generated by the alarm logged by a MES system with raw information coming from equipment PLCs of a production line, in the sixth use case a dataset coming from an IT reporting system, where the field service engineers store information about non-conformities found during commissioning activities.

Stakeholder need ID	Stakeholder need definition
SN1	The big data analytics tool shall provide insights about critical alarms impacting the overall production line efficiency, clustered by equipment.
SN2	The big data analytics tool shall provide a grouping of non-conformities descriptions leveraging on text similarities, in order to organize the information in a structured database to be used as a decision support system.

5.3 System Requirements elicitation

Requirements elicitation for use case 5

System Requirement ID	System Requirement definition
SR1	The data analytics tool shall be installable on the computer of production analysts (OS: Microsoft Windows)
SR2	The data analytics tool shall be capable to extract the data from Microsoft SQL Server Databases of Alarm history of MES system
SR3	The data analytics tool shall be capable to extract the detailed data from Microsoft SQL Databases of each equipment (with PLC logs) in order to collect production information to contextualize the alarm history of the overall production line
SR4	The data analytics tool shall be capable to recognize only alarm stops causing the downtime of the line, not the ones generated as consequence of another stop happened downstream or upstream in the same production process
SR5	The data analytics tool shall save the filtered alarm logs in a unique file containing only the alarms generating a production line downtime

SR6	The data analytics tool shall integrate the possibility to filter out statistical outlier stops, that represent uncommon values that are not relevant for performance improvement analysis
SR7	The data analytics tool shall provide a time chart with the trend of MTTR of stop alarm, in order to contextualize its steady state
SR8	The data analytics tool shall provide a pattern analysis module to highlight recurrent sequences of stop alarms to enable the search for any sequential correlations

Requirements elicitation for use case 6

System Requirement ID	System Requirement definition
SR1	The data analytics tool shall be installable on the computer of quality and issue resolution specialists (OS: Microsoft Windows)
SR2	The data analytics tool shall be capable to extract the data from CSV dataset in form of unstructured natural language descriptions (free text)
SR3	The data analytics tool shall be capable to group similar issues leveraging on text similarities
SR4	The data analytics tool shall provide the grouped datasets in CSVs
SR5	The data analytics tool shall provide dashboards organized as histograms to explore clusters of similar descriptions

General guidelines, we can summarize and extract from the elicitation for the use cases, are that it is important to define the receiving organization of the DSS and the expected optimization actions needed in their daily activities, in order to design not only the functionalities in scope but to collect information regarding the usability needed for the UX design of HMI and dashboards (non-functional requirements).
In both use cases we identified the need of creating a structured dataset to enable further analytics and to display the aggregate information in form of a visual dashboard. In general, the need of a real-time Decision Support System was not captured, while it was

in interest a tool to simplify, organize and structure historical information already collected but not yet used for human decision-making process, due to their unstructured nature.

5.4 Definition of functional architecture and physical architecture

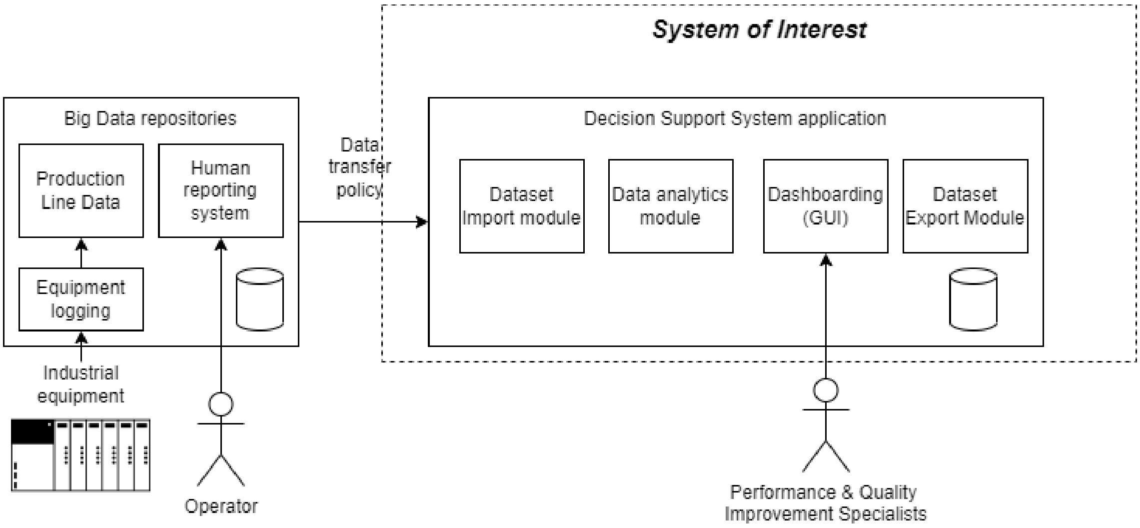

Figure 21: Functional Architecture for Data processing and visualization systems

The functional architecture (Figure 21) is important to visualize the main interfaces to be considered in the System of Interest development according to Systems Engineering practices. The functional architecture deriving from the 2 use cases is common, even if the physical architecture developed was different due to the different non-functional requirements collected during the system requirements phase.

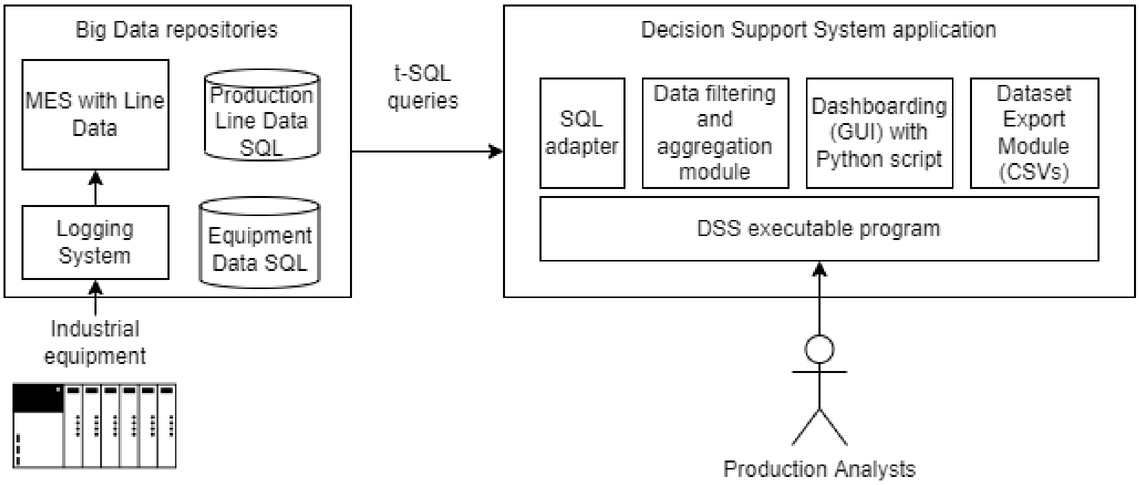

Figure 22: Physical architecture for use case 5

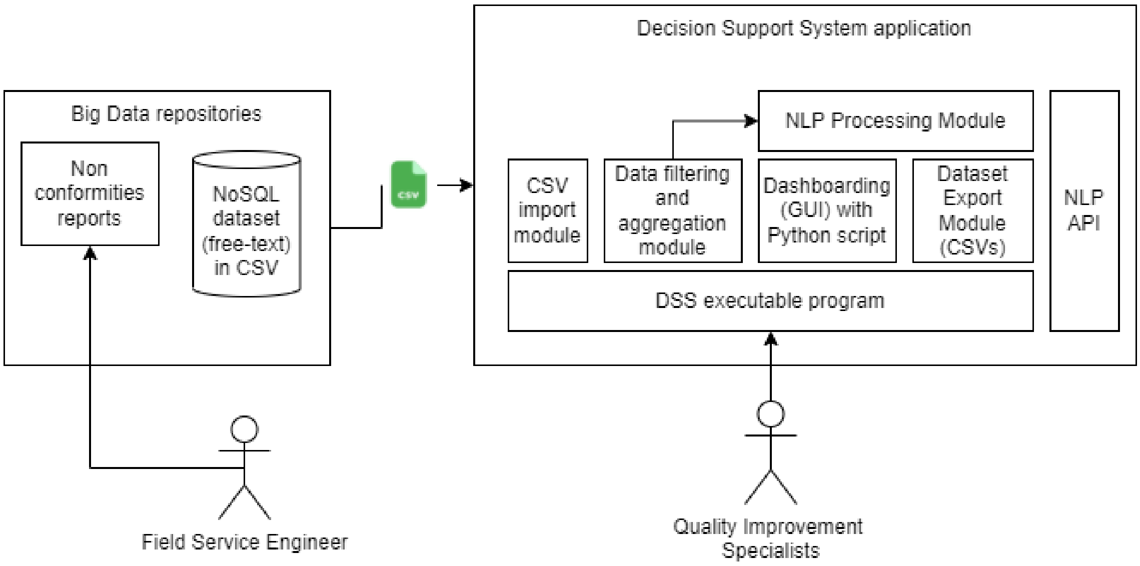

Figure 23: Physical architecture for use case 6

5.5 Detailed design principles

For the fifth use case the relevant physical modules from a detailed design point of view are:

5.5.1 UC5: Data filtering and aggregation module

The "data filtering and aggregation" module of the Decision Support System was realized with a Python script, using the Pandas library in order to manage the dataset in form of a

dataframe (allowing similar operations when developing queries in DBs with SQL structure).
First task implemented in the module was the data filtering, in order to keep only the logs about stop alarms of the equipment causing the production line downtime: this task was possible correlating the full Alarm history present in the MES database with the system logs present in the SQL DB of each equipment in the production line, where was reported the information about how much time the line stopped due to an equipment alarm.

As parameter for alarm impact assessment, the MTTR KPI was chosen and computed, [59]

$$MTTR = \frac{Total\ Repair\ Time}{N\ of\ failures} \qquad \text{[Eq8]}$$

This parameter is computed in the data filtering and aggregation module in function of each production line, each equipment and each alarm, since the objective is having the information necessary to plan performance optimization identifying main impacting issues on each equipment.
In the filtering module is also present an outlier filtering function, leveraging on the fence technique based on the box-plot representation, [60]. In this specific technique it is necessary to order the dataset of a specific alarm (i.e. the time series about its activation log) in ascending order basing on the total duration of the alarm.
We define:

- Median, the central record of the dataset. In case the number of data is even, it is chosen the average number of the two records.
- Lower interquartile Q_1, the value found taking the record at the 25% of the series
- Superior interquartile Q_3, the value found taking the record at the 75% of the series
- Interquartile range IQR, the difference $Q_3 - Q_1$

We can define a value as "outlier" if it is located outside the Inner Fence range.

$$Lower\ Inner\ Fence = (Q_1 - 1.5\ \cdot IQR) \qquad \text{[Eq9]}$$

$$Upper\ Inner\ Fence = (Q_3 +\ 1.5\ \cdot IQR) \qquad \text{[Eq10]}$$

5.5.2 UC5: Dashboard module

The dashboard system was designed in order to provide incremental insight on the dataset for a specific equipment located in a production line. The open source matplotlib library was used to implement the dashboards in the prototype, [61].

- The first chart displayed is useful to evaluate which stops are potentially not to be considered in the performance improvement activity, since they are considered as outliers (Figure 24).
- The second chart is used to understand if the MTTR used for the impact analysis of the alarm can be considered stabilized with current dataset or it is better to include larger dataset for evaluating a steady state value, (Figure 25).
- The third chart and fourth chart are used to evaluate the impact of each alarm causing a production line stop on a specific equipment, leveraging on total duration, occurrences and MTTR, (Figure 26) and (Figure 27).
- The fourth and last type of chart designed to support the Optimization process is a graph that recognizes recurrent alarm sequences that could support in finding failure modes relationships in the production machine, (Figure 28).

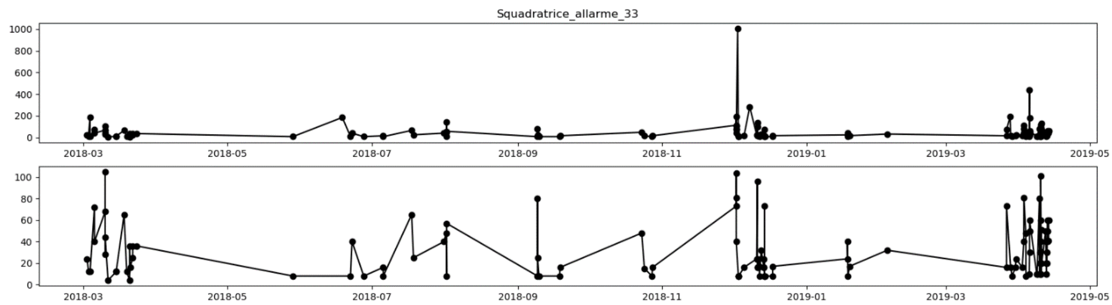

Figure 24: Outliers filtering with Inner Fence technique

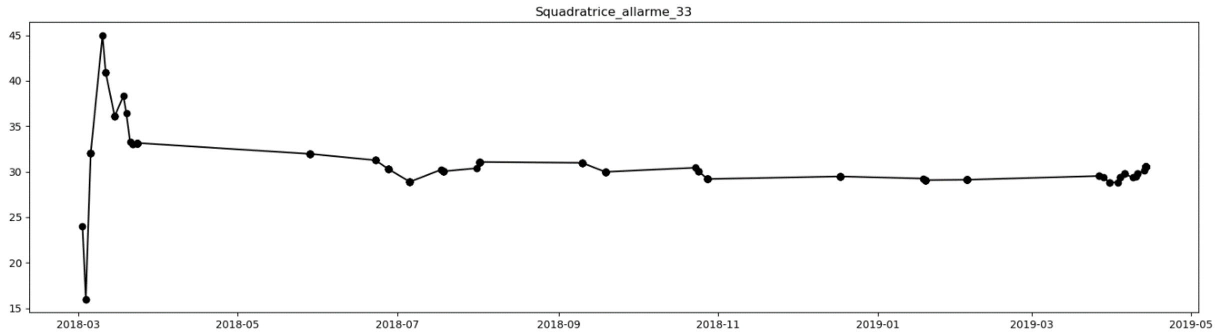

Figure 25: Dashboard for MTTR steady state evaluation

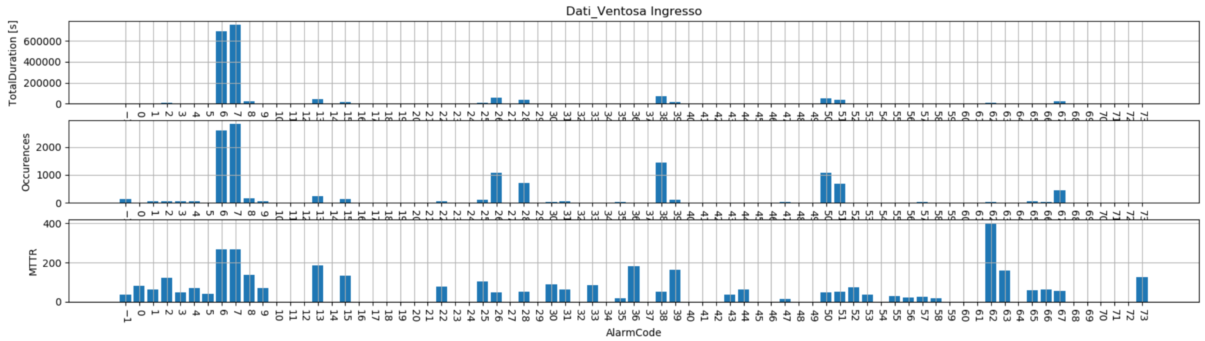

Figure 26: Alarm impact dashboard for a specific Equipment

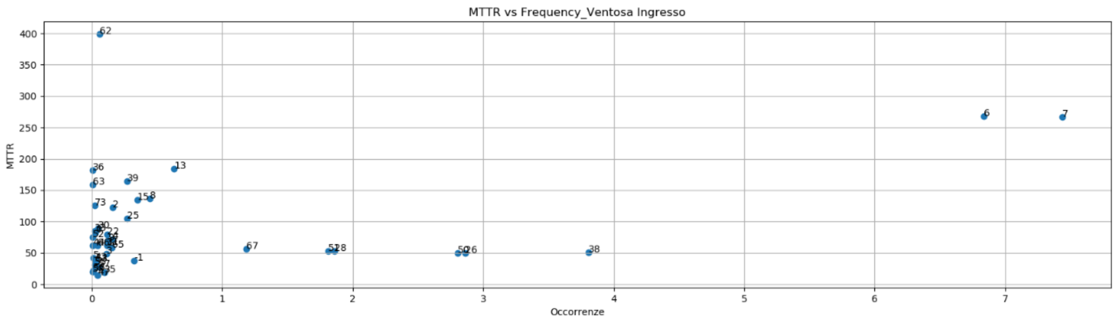

Figure 27: MTTR vs Frequency chart

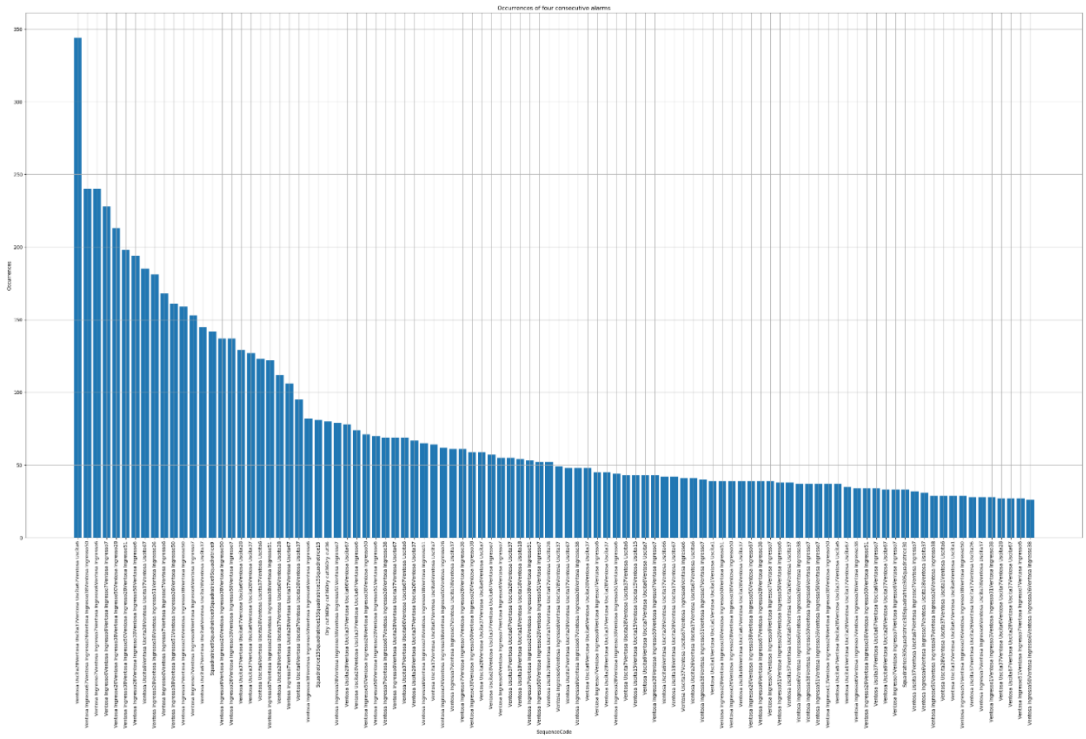

Figure 28: Recurring alarm sequence

For the sixth use case the relevant physical modules from a detailed design point of view are:

5.5.3 UC6: Data filtering and aggregation module

To design the data filtering and aggregation module for the sixth use case, the techniques of Natural Language Processing (NLP), the Machine Learning area for unstructured text analysis, were applied.

Summarizing, the module applies the following functionalities:

1. Dataset partitioning according to the equipment on which the defect was detected.
2. Stopwords definition: predefined stopwords + customized stopwords. We define “stopword” a word that is not relevant for the semantic analysis of the text.
3. Punctuation removal.
4. Conversion of strings to lowercase characters.
5. Tag, special characters and digits removal.
6. Tokenization of sentences in single words.
7. Removal of stopwords.

Final objective of the filtering module is to produce a "Corpus", that in NLP represents the complete list of semantically relevant words, used to describe the different texts present in the dataset.

5.5.4 UC6: NLP Processing module

The module of NLP has the purpose of grouping semantically similar descriptions of non-conformities inserted in the dataset by test engineers during internal validation activities or by field service engineers during commissioning phases.

During the concept design phase, we thought to perform a clustering via unsupervised learning techniques using k-means, leveraging on "term frequency–inverse document frequency" (TF-IDF) [62] to determine the weight of the word in the vector to be used for the machine learning clustering.

After internal review with Technical Issue experts, it was observed how the clusters found via this technique were difficult to interpret. It was preferred to move to the "frequentist approach" using the n-grams, since this methodology produced clusters that were evaluated by specialists more correlated and connected.

The n-grams are histograms where on the x-axis are reported groups of n consecutive words taken by the dataset corpus and on the y-axis are reported the frequencies [63], [64]. The development of the NLP Processing module was done using the Python programming language, using the Pandas library for the part of interacting with the dataset and its conversion into an object-oriented structure and, for the processing part of the language, the libraries sklearn and nltk.

5.5.5 UC6: Dashboard module

The dashboard system was designed in order to provide incremental insight on the dataset.

- Distribution of the non-conformities reports of the machines (Figure 29).
- Given in input the value n for computing the n-grams, the histograms are produced for each equipment/mechanical module present in the dataset (Figure 30).
- Selecting a specific column of the histogram, it is possible to visualize the dataset defined by the n-gram in scope, (Figure 31).

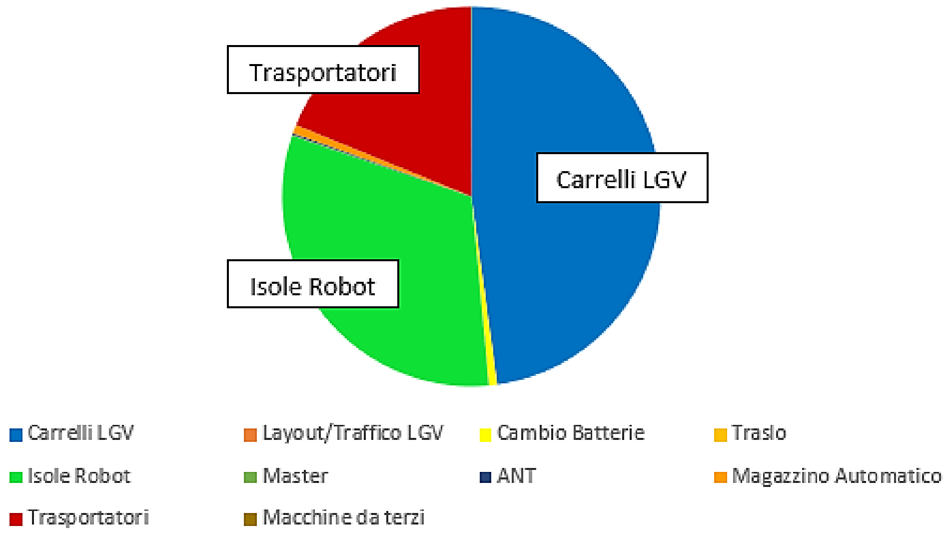

Figure 29: Non-conformities distribution over the machines

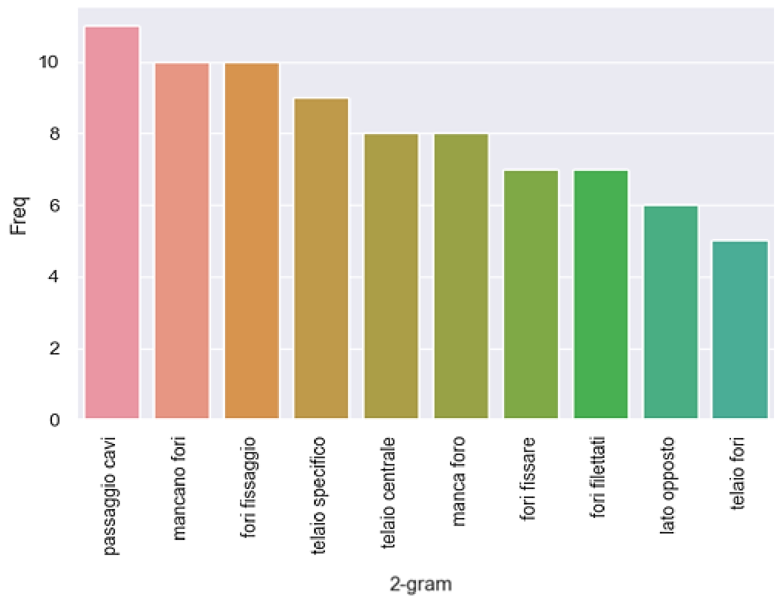

Figure 30: N-gram dashboard with histogram

NC	Descrizioni	Codici
18/3945	mancano n°2 fori per consentire il passaggio dei cavi sul pezzo in oggetto, inoltre r	b982245013
18/3109	sul cod.776234119 bisogna inserire un foro di Ø10 per il passaggio di tutti i cavi, sul	b343577864
18/2875	è necessario allargare il foro per passaggio cavi, aggiornare il disegno del telaio in	b113457112
18/2657	Bisogna aggiungere un foro sul cod.188231678 per passaggio cavi	B16744389
18/2612	Sul supporto cod.81651134 è necessario aggiungere n°4 fori per il passaggio dei cav	b225683421
18/2541	necessario inserire a disegno n°2 fori per il passaggio dei cavi, vedere allegato. La	b144760026
18/2498	su cod. 14577284 bisogna allargare il foro passaggio cavi perché non è sufficientem	b113457112
18/2456	il foro passaggio cavi va allargato da Ø8 a Ø11, il telaio cod.133981356 non risulta e	B83348315
18/1967	occorrono n°2 fori per il passaggio dei cavi. vedi foto allegate.	B44715689
16/0574	sul cod.4471435 non sono presenti i fori per il passaggio cavi, occorre modificare il	b35992056

Figure 31: Dataset of a specific n-gram

5.6 Verification and Validation

The internal verification was mainly designed to verify the technical steps applied on the dataset import and manipulation (filtering and analytics).

Use case 5

Verification ID	Module	Description	Goal
Ver1	SQL adapter	1. Providing the connection string to the MS SQL DB and the user credentials, it is possible to establish the connectivity with SQL DB Server 2. It is possible to parse the SQL DB in a Pandas Dataframe	Check the connectivity with Big Data sources and the dataset parsing with Pandas Dataframe
Ver2	Data filtering and aggregation module	1. The outliers are removed and compared with manually computed values for inner fence ranges 2. The parameters of interest (total duration, MTTR and occurrences) are compared with sample values manually precomputed on the same dataset 3. Some sequences of alarms are manually checked and compared	Check the main functionality of removing the outliers and computing the KPI values for DSS

		with the frequencies reported by the UI	
Ver3	Dashboard module	1. The data visualized in the graphs are compared to the technical values computed by the “data filtering and aggregation” module	Check the data consistency in the User interface
Ver4	Dataset export module	1. The format of the file produced in output by the analytics tool is the expected one (CSV) and can be opened with an open source reader (like CSV using Notepad) 2. The file content is compared with the values present in the UI graphs	Check the correctness of structured datasets obtained by the DSS application

Use case 6

Verification ID	Module	Description	Goal
Ver1	CSV import module	1. Check the possibility to import and parse as Pandas Dataframe the CSV with a specific format predefined 2. Check the capability of the system to reject different file formats,	Check the possibility to import the Dataset

		without generating a system failure	
Ver2	Data filtering and aggregation module	1. Check the capability of removing stopwords and other elements not relevant for the text similarity analysis 2. Manually check for one description that the remaining words are all present in the corpus of the dataset	Check the functionality of data filtering and corpus preparation
Ver3	NLP Processing Module and Visualization Module	1. Check the presence of the words in the n-gram using the sub-dataset 2. Check the correct refactoring of the histograms changing the value "n" for n-grams computation 3. Check the right order of the columns in the histogram, compliant with the frequency in y-column	Check the functionality of histogram visualization for the DSS application
Ver4	NLP API module	1. Check the exposed methods can accept both the POST of the dataset and the GET of a specific n-gram	Check the functionality of REST API focusing on data contracts of HTTP methods

		2. Check that http requests with different contracts are rejected	

In these use cases we were more interested in the validation phase performed together with the engineers of SMEs involved in the definition of the Decision support systems. It was important to refine and define incrementally new requirements for the visualization module, to facilitate the data exploration and a rapid switching among different visualizations to understand the ongoing status of the production process or of a particular issue group.

In future research it will be useful to deepen the unsupervised clustering methodologies for the analysis of unstructured texts, possibly correlating the textual information with multimedia contents to be attached to the reporting system (such as photographs and videos) to further guide the system in finding similarities in the situations reported by the engineers. It will also be interesting an investigation on how to exploit and integrate the analysis module on historical data with a new real-time observation module, for the real-time recognition of possible patterns already found in previous production batches, in order to anticipate possible events to operators and guiding the resolution in an optimized way.

5.7 Conclusions

With these two use cases chosen as examples for the area "Big Data and Decision Support Systems" it was possible to support SMEs involved in the initiative in understanding:

1. How much value can be extracted from unstructured datasets, defining incrementally the desired outputs in terms of analytics and correlations.
2. How modularization can support in adding incrementally functionalities without huge impact in the overall architecture (like API module added to NLP processing module in later stages for the required integration into an external web application).
3. How service-oriented design and modularization can facilitate asynchronous evolution of the UI with respect the back-end services.

Chapter 6: Conclusions

With the research activities performed during the PhD course in the domain of Industry 4.0, the following results have been achieved:

- Assessment of the SCRUM agile framework as proper product development process facilitating the knowledge sharing between Academia and small-medium enterprises willing to introduce new technologies in their business.
- Generalization of a structured design flow, following Systems Engineering principles, in order to manage the complexity of distributed systems, both from the technological and configuration management perspectives.
- Definition of system architectures with modular approach, to improve the scalability of the application, to facilitate the feature incremental approach required by modern software services.
- Technological scouting to solve common I4.0 engineering problems, with the possibility of choosing different solutions depending on the technical and cost requirements in scope of the use case under investigation.
- Organized project planning for I4.0 proof of concepts, providing inputs and suggestions to the SMEs management for the project governance in this innovative domain.

6.1 Holistic review of the integrated approach

From the various proof-of-concepts analysed and developed with the ARSControl research group during the 3 years of the PhD, it is possible to extract transversal and integrated guidelines concerning research and development activities related to Industry 4.0 and, in general, to digitalization in industries.

The first factor that unites digitalization activities is the speed of evolution of needs: the new products and services developed by modern industries tend to be oriented and adhering to the requests of the individual customer; we try to achieve this flexibility for different needs through standard but highly configurable products. For this reason a rigorous approach to configuration management and version control is required to keep track of the different modules developed using modern tools such as GIT, as explained in Chapter 1 and Chapter 2.

This speed of change, together with the emergence of new technologies in the IT field, is creating difficulties in small and medium-sized enterprises in adopting new enablers, in some cases both due to the immaturity of the skills required in the IT field and to the difficulty of understand the achievable business benefit. For this reason, unlike other engineering sectors where technological evolution or business needs are slower and more stable over time, companies operating in the digital and automation sector must be ready for frequent and rapid changes in needs. This approach is facilitated by Agile development methodologies (i.e. with an incremental implementation approach to obtain frequent feedback and confirmations on the released value) which require a very careful and structured approach to Systems Engineering. In Chapter 2 it is possible to find a mapping between the artifacts foreseen by the SCRUM methodology and those required by the Systems Engineering methodologies, with the advantage of being able to structure development methodologies (or research projects as in the case of this PhD thesis) incremental and suitable for the management of requirements changes.

As a result of rapidly changing requirements, end-to-end version control systems are needed that cover all development artifacts: requirements and specification documents, system architecture diagrams, detailed design diagrams, test suites and release documentation. The main goal is to achieve a clear traceability of the requirements, of the architecture to meet the requirements and a clear link between the verification approach and the initial request. In Chapters 3, 4 and 5 it is possible to find the presentation of different proof-of-concepts developed during the PhD course: for all the prototypes a common approach was used for the analysis and implementation. We start from the analysis of the state of the art to identify tools and methodologies, we analyse the problem to identify the use case and the needs of the stakeholders, we formalize the system requirements for the architectural design, we apply tools and methodologies for the implementation of the architectures and finally a verification and validation plan is identified to evaluate the technical and conceptual benefit of the system.

Another recurring and transversal paradigm for the new technologies of Industry 4.0 is the possible modular approach: the advantages are essentially 3.

- By dividing the system into modules, it is possible to limit the impact of changing requirements by reworking only the architectural modules affected by the changed requirement.

- Modularization allows the parallelization of design, development and testing of units, shortening development times and, consequently, providing a faster response to market needs.
- Especially in companies still immature in some Industry 4.0 technologies, with modularization there is also the possibility of defining well-confined and localized black-box subsystem requirements on specific modules, defining a verification approach oriented towards system integration. In the latter case, the advantage is to be able to study the most convenient solution even with the support of external companies or organizations (such as university research centers), without necessarily having to share the business need and the domain knowledge, which remains the main know-how and advantage of a company.

The use of the modular and decentralized approach is a recurring topic in Chapters 3, 4 and 5 of the thesis as it forms the basis of the new technologies of Industry 4.0: in particular in the part of the analysis of the architectures and of the verification and validation plans, I highlight the benefits due to modularization both for the management of the change of requirements and for the parallelization of the testing activities prior to the system integration activities.

Finally, a very important topic in Industry 4.0 remains the central role of the human operator in modern industry: from the studies conducted by numerous researchers, the unclear future role of the human operator and lack of clarity in establishing the boundary between tasks to be automated and tasks to be performed with human work create concern. For this reason, digital enablers must be designed primarily to enhance the intellectual role of the human operator by delegating repetitive tasks or health-threatening interactions to automation.

The issue of the readability of complex distributed systems plays a fundamental role in industry 4.0: IT enablers capable of analysing large amounts of data and aggregating its parameters in dashboards must be thought of as a central part of the user's journey during user experience (UX) analysis of products, processes and services. The human operator must be able to understand and make decisions based on what is proposed at a high level by the systems. The study of dashboard and HMI systems is certainly relevant within an accurate analysis of requirements and architecture: this is also demonstrated by the fact that this element is common to all the use cases proposed in this thesis.

I believe that chapter 5 is particularly relevant for the development of Decision Support Systems, which explains in detail the importance of integrating digital tools in the management process that continues to be coordinated and governed by human intelligence, supported by the calculation capacity of modern software systems.

6.2 Scientific contribution during the PhD course

During the three years of PhD course, my scientific contribution to Automation and Software Engineering research community was achieved thanks to these activities, additionally to the projects presented in previous chapters of the Thesis:

- Conference Papers
- Journal Article
- Workshop on "Industrial agents in Industry 4.0" at Technical University of Munich (TUM)
- Workshop on "Industrial distributed systems and AI" at the Reggio Emilia Technopole
- Workshop to present the achievements of "PID: Impresa 4.0" at Chamber of Commerce of Reggio Emilia with representatives of SMEs

6.2.1 Conference papers

Together with other researchers of the ARSControl research lab, I wrote and published three papers for relevant conferences of industrial Automation and IT sector. The main topics investigated in these papers are:

1. The state of the art of multi-agent applications in industrial domain, specifically in the logistics and manufacturing domain. In the paper we presented an innovative algorithm for the control of production lines based on AgentSpeak programming language with the Jason framework, based on belief-desire-intention pattern of intelligent distributed systems. This paper was written in cooperation with R&D engineers of an important multinational company in the packaging sector for food protection. The paper "Decentralized line equipment detection and production control by multi-agent technology" was presented during the conference "IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society", [40].
2. The Digital Twin applications to monitor production environment with wireless sensor networks interfaced with a BIM modelling system, to reach a quality

control of indoor air parameters. This topic has been presented in the PhD thesis in the chapter “Digital Twin and IoT multi-agent systems”. The paper “Air Quality Monitoring in a BIM model by means of a IoT Sensors Network” was presented during the conference “2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)”, [65].

3. The application of BLE technology for the low energy consumption WSN networks with synchronized deep sleep phases, to address applications with low connectivity coverage (both cellular and Wi-Fi). This topic has been presented in the thesis in the chapter “Digital Twin and IoT multi-agent systems”. The paper “Self-configuring BLE deep sleep network for fault tolerant WSN” has been accepted and will be presented during the conference “11th IFAC Symposium on Fault Detection, Supervision, and Safety of Technical Processes (SAFEPROCESS 2022), 2022”.

6.2.2 Journal article

Thanks to the great achievements reached as research lab in the domain of control and modelling of distributed IoT WSNs, we decided to write an article for a relevant and open access journal, “Sensors” of MDPI. The title of the paper is “Smart Node Networks Orchestration: A New E2E Approach for Analysis and Design for Agile 4.0 Implementation”, [39]. In the paper we described the systems engineering flow to analyse the end-to-end process for the ontology design, development, implementation, and verification of different architectures of IoT sensor networks used in industrial domain, both considering static sensor nodes, or nodes applied to mobile systems. The article is not only focused on the description of data collection systems on embedded level, but it describes also the modules needed for a data processing and analytics applications, as well as the visualization systems to implement web-based HMIs.

6.2.3 Workshop on “Industrial agents in Industry 4.0” at Technical University of Munich (TUM)

During the research on the topic of industrial agent systems, I had the opportunity to present my research activities and results regarding multi-agent control systems and IoT WSNs in a workshop held at TUM with other research fellows and PhD students. This experience was inspirational, because it was the opportunity of knowledge sharing about different applications based on the same technology and design pattern.

6.2.4 Workshop on "Industrial distributed systems and AI" at the Reggio Emilia Technopole

The results of the research on distributed systems and decision support systems carried out in collaboration between the ARSControl laboratory and automation companies were presented in a dedicated workshop held at the Reggio Emilia technopole. This was an opportunity to gather new inputs for research on topics from stakeholders of various companies operating in the industrial automation sector.

6.2.5 Workshop to present the achievements of "PID: Impresa 4.0" at the Chamber of Commerce of Reggio Emilia with representatives of SMEs

The application results and research achievements of the initiative "PID: Impresa 4.0" were presented at Chamber of Commerce of Reggio Emilia. In this workshop both the researchers involved in the activities and the SMEs representatives involved in the digitalization program were present. This was an opportunity for a final validation of the proof of concept and it was interesting to note how several SMEs decided to continue their research activities on the issues of digitization investigated with the "PID: Impresa 4.0" initiative. This choice highlighted the effectiveness of this project and the potential of technology transfer programs between academia and industry.

Chapter 7: Acknowledgments

My PhD studies at the Department of Engineering of the University of Modena and Reggio Emilia ends with this thesis work.

A heartfelt thanks is dedicated to my supervisor, Prof. Cesare Fantuzzi: he gave me the opportunity to deepen numerous important topics in the field of industrial automation and modern software engineering processes.

I also thank all the people with whom I have collaborated in the research topics for the great availability, team spirit and exchange of ideas and knowledge that have taken place over the years: Annalisa, Carlo, Marco, Nicola you have been a great reference point.

Thanks also to my Line Manager, Gaetano, who has recognized an opportunity for growth in this period of study and has always encouraged me to complete this path, without ever seeing it as a limitation in the work I carried out in parallel.

I thank my family for helping me to face this course of study serenely, despite all the difficulties of these years: the family is the fundamental point from which to start to build one's horizons.

Bibliography

[1] K. A. W. K. P. &. A. H. Jan Stentoft, “Drivers and barriers for Industry 4.0 readiness and practice: empirical evidence from small and medium-sized manufacturers,” *Production Planning & Control,* vol. 32, pp. 811-828, 2021.

[2] M. L. P. G. M. W. J. J. P. E. a. M. H. M. Rußmann, “Industry 4.0: The Future of Productivity and Growth in Manufacturing Industries,” *MA: Boston Consulting Group,* 2015.

[3] E. L. X. &. L. L. Li Da Xu, “Industry 4.0: state of the art and future trends,” *International Journal of Production Research,* vol. 56, pp. 2941-2962, 2018.

[4] S. M. P. H. Z. a. V. T. Raza, “Building the Internet of Things with Bluetooth Smart,” *Ad Hoc Networks,* pp. 19-31, 2017.

[5] E. O. P. L. Nelson Rodrigues, “Decentralized and on-the-fly agent-based service reconfiguration in manufacturing systems,” *Computers in Industry,* vol. 101, pp. 81-90, 2018.

[6] P. Leitao, “Agent-Based Distributed Manufacturing Control:A State-of-the-Art Survey,” *Engineering Applications of Artificial Intelligence,* vol. 22, pp. 979-991, 2009.

[7] J. Butt, “A Strategic Roadmap for the Manufacturing Industry to Implement Industry 4.0,” *Designs (mdpi),* 2020.

[8] A. G. R. S. Sachin S. Kamble, “Analysis of the driving and dependence power of barriers to adopt industry 4.0 in Indian manufacturing industry,” *Computers in Industry,* vol. 101, pp. 107-119, 2018.

[9] A. S. S. L. Chetna Chauhan, “Barriers to industry 4.0 adoption and its performance implications: An empirical investigation of emerging economy,” *Journal of Cleaner Production,* vol. 285, 2021.

[10] F. R. A.C. Pereira, “A review of the meanings and the implications of the Industry 4.0 concept,” *Procedia Manufacturing,* pp. 1206-1214, 2017.

[11] J. M. Müller, “Assessing the barriers to Industry 4.0 implementation from a workers’ perspective,” *IFAC-PapersOnLine,* vol. 52, no. 13, pp. 2189-2194, 2019.

[12] C. L. P. D. Erwin Rauch, “Anthropocentric perspective of production before and within Industry 4.0,” *Computers & Industrial Engineering,* vol. 139, 2020.

[13] CCIAA, “PID: Impresa 4.0,” [Online]. Available: https://www.puntoimpresadigitale.camcom.it/eventipid/workshop-applicazioni-tecnologie-impresa-40-camera-commercio-reggio-emilia-unimore.

[14] D. -. U. o. M. a. R. Emilia, “ARSCONTROL website,” [Online]. Available: https://www.arscontrol.org/. [Accessed 2021 12 19].

[15] INCOSE, “Systems Engineering,” [Online]. Available: https://www.incose.org/about-systems-engineering/system-and-se-definition/systems-engineering-definition. [Accessed 10 11 2021].

[16] K. F. Dick Fairley, “System Life Cycle Process Models: Vee,” Guide to the Systems Engineering Body of Knowledge (SEBoK), [Online]. Available: https://www.sebokwiki.org/wiki/System_Life_Cycle_Process_Models:_Vee.

[17] “Agile Manifesto,” [Online]. Available: https://agilemanifesto.org/iso/it/manifesto.html. [Accessed 20 01 2022].

[18] J. M. G.-C. J. G. J. M. M. N. N. S. Emilio L. Cano, "A Scrum-based framework for new product development in the non-software industry," *Journal of Engineering and Technology Management,* vol. 61, 2021.

[19] D. W. (Atlassian), " Sprint planning," [Online]. Available: https://www.atlassian.com/agile/scrum/sprint-planning. [Accessed 05 01 2022].

[20] D. R. (Atlassian), " Why code reviews matter (and actually save time!)," [Online]. Available: https://www.atlassian.com/agile/software-development/code-reviews. [Accessed 07 11 2021].

[21] J. Alblas, " Scrum from the Trenches - Product Backlog Refinement is a Scrum Team Responsibility," scrum.org, 13 11 2018. [Online]. Available: https://www.scrum.org/resources/blog/scrum-trenches-product-backlog-refinement-scrum-team-responsibility. [Accessed 12 12 2021].

[22] " What is a Sprint Review?," [Online]. Available: https://www.scrum.org/resources/what-is-a-sprint-review. [Accessed 06 01 2021].

[23] GIT. [Online]. Available: https://git-scm.com/. [Accessed 10 12 2021].

[24] Microsoft, "Adopt a Git branching strategy," [Online]. Available: https://docs.microsoft.com/it-it/azure/devops/repos/git/git-branching-guidance?view=azure-devops. [Accessed 20 11 2021].

[25] "Diagrams Net," [Online]. Available: https://www.diagrams.net/. [Accessed 01 02 2022].

[26] C. J. Hartmann W.M., "Acoustic Signal Processing," *Springer, New York, NY,* 2014.

[27] R. Analytics, "Model-Driven vs Data Driven methods for Working with Sensors and Signals," 20 07 2016. [Online]. Available: https://medium.com/@RealityAI/model-driven-vs-data-driven-methods-for-working-with-sensors-and-signals-98fc6ac8cc92. [Accessed 19 12 2021].

[28] G. Mendels, "How to apply machine learning and deep learning methods to audio analysis," 18 11 2019. [Online]. Available: https://towardsdatascience.com/how-to-apply-machine-learning-and-deep-learning-methods-to-audio-analysis-615e286fcbbc. [Accessed 04 12 2021].

[29] "Top 10 Programming Languages to Learn in 2022," 07 01 2022. [Online]. Available: https://www.geeksforgeeks.org/top-10-programming-languages-to-learn-in-2022/. [Accessed 10 01 2022].

[30] S. Asiri, "Machine Learning Classifiers," towardsdatascience, 11 06 2018. [Online]. Available: https://towardsdatascience.com/machine-learning-classifiers-a5cc4e1b0623. [Accessed 20 11 2021].

[31] P. Š. J. M. a. I. Z. T. Lojka, "Web Technologies in Industry HMI," *IEEE 19th International Conference on Intelligent Engineering Systems,* pp. 103-106, 2015.

[32] Bootstrap, "Build fast, responsive sites with Bootstrap," [Online]. Available: https://getbootstrap.com/. [Accessed 12 01 2022].

[33] "Data-Driven Documents," [Online]. Available: https://d3js.org/. [Accessed 12 01 2022].

[34] WIKIPEDIA, "Huglin Index," [Online]. Available: https://en.wikipedia.org/wiki/Huglin_index. [Accessed 2021 11 11].

[35] O. P. B. Š. Z. K. a. F. T. Slavko Bernáth, "Influence of Climate Warming on Grapevine (Vitis vinifera L.) Phenology in Conditions of Central Europe (Slovakia)," *Plants (mdpi),* vol. 10, 2021.

[36] A. Falaschi, April 2017. [Online]. Available: https://teoriadeisegnali.it/libro/html/html/libro-4.4.html. [Accessed 08 10 2021].

[37] W. Y. T. D. Z. M. a. M. F. Huang Xuemei, “Theoretical analyze and implementation method of reconfigurable assembly line based on agent and holon,” *Fifth World Congress on Intelligent Control and Automation (IEEE Cat. No.04EX788),* vol. 3, pp. 2830-2833, 2004.

[38] M. Wooldridge, “An Introduction to MultiAgent Systems, 2nd Edition,” *Wiley,* 2009.

[39] A. Bertoli, A. Cervo, C. Rosati and F. C., “Smart Node Networks Orchestration: A New E2E Approach for Analysis and Design for Agile 4.0 Implementation,” *Sensors ,* vol. 21, 2021.

[40] G. G. C. F. a. R. B. A. Cervo, “Decentralized line equipment detection and production control by multi-agent technology,” *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society,* pp. 2940-2945, 2019.

[41] R. Unland, “Chapter 1 - Software Agent Systems,” *Industrial Agents,* pp. 3-22, 2015.

[42] I. C. Society, “FIPA,” 2005. [Online]. Available: http://www.fipa.org/. [Accessed 10 11 2021].

[43] Safeatlast, “80 Insightful ‘Internet of Things’ Statistics (Infographic),” 25 11 2021. [Online]. Available: https://safeatlast.co/blog/iot-statistics/#gref. [Accessed 12 12 2021].

[44] M. R. A. D. B. M. G. Nacer Khalil, “Wireless Sensor Network for Internet of Things,” *IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP) Symposium on Public Internet of Things,* 2014.

[45] C. E. H. G. S. D. P. Viorel Mihai, “WSN and Fog Computing Integration for Intelligent Data Processing,” *Distributed, Parallel, and Cluster Computing,* 2019.

[46] ISO/IEC, “Information technology — Message Queuing Telemetry Transport (MQTT) v3.1.1,” 06 2016. [Online]. Available: https://www.iso.org/standard/69466.html. [Accessed 13 12 2021].

[47] E. Foundation, “Eclipse Mosquitto™ An open source MQTT broker,” [Online]. Available: https://mosquitto.org/. [Accessed 03 01 2022].

[48] E. IoT, “Paho,” [Online]. Available: https://www.eclipse.org/paho/. [Accessed 04 01 2022].

[49] J. Marcel, “New Trends and Forecasts in the 2021 Bluetooth Market Update,” SIG, 30 03 2021. [Online]. Available: https://www.bluetooth.com/blog/new-trends-and-forecasts-in-the-2021-bluetooth-market-update/. [Accessed 12 01 2022].

[50] S. D. S. J. M. G. H. G. a. S. M. M. Khan, “Context-aware low power intelligent SmartHome based on the Internet of things,” *Computers & Electrical Engineering,* vol. 52, 2016.

[51] L. S. M. H. N. d. L. John Harris, “A Summary of Bluetooth Low Energy,” Defence Technology Agency (DTA),, 04 2020. [Online]. Available: https://www.dta.mil.nz/assets/Publications/A-Summary-of-Bluetooth-Low-Energy.pdf. [Accessed 05 12 2021].

[52] WIKIPEDIA, “Universally unique identifier,” [Online]. Available: https://en.wikipedia.org/wiki/Universally_unique_identifier. [Accessed 19 01 2022].

[53] B. SIG, “Assigned Numbers,” [Online]. Available: https://www.bluetooth.com/specifications/assigned-numbers/. [Accessed 16 01 2022].

[54] G. Cheung, “What Exactly is Big Data in 2020?,” 10 11 2019. [Online]. Available: https://towardsdatascience.com/what-exactly-is-big-data-in-2020-9acee48e8dd7. [Accessed 02 01 2022].

[55] M. H. Amir Gandomi, “Beyond the hype: Big data concepts, methods, and analytics,” *International Journal of Information Management,* vol. 35, pp. 137-144, 2015.

[56] Qlik, “Decision Support System,” [Online]. Available: https://www.qlik.com/us/business-intelligence/decision-support-system. [Accessed 15 01 2022].

[57] M. Wolf, “Chapter 8 - Internet of Things systems,” *Computers as Components (4th edition),* 2016.

[58] R. M. J. C. a. J. O. C. J. Turner, “Human in the Loop: Industry 4.0 Technologies and Scenarios for Worker Mediation of Automated Manufacturing,” *IEEE Access,* vol. 9, 2021.

[59] L. C. M. P. G. I. a. S. L. M. Grisostomi, “Application of a wireless sensor networks and Web2Py architecture for factory line production monitoring,” *IEEE 11th International Multi-Conference on Systems,* pp. 1-6, 2014.

[60] R. d. S. Neil C. Schwertman, “Identifying outliers with sequential fences,” *Computational Statistics & Data Analysis,* vol. 51, no. 8, pp. 3800-3810, 2007.

[61] Matplotlib, “Matplotlib: Visualization with Python,” [Online]. Available: https://matplotlib.org/. [Accessed 25 01 2022].

[62] Y. Hamdaoui, “TF(Term Frequency)-IDF(Inverse Document Frequency) from scratch in python,” 10 12 2019. [Online]. Available: https://towardsdatascience.com/tf-term-frequency-idf-inverse-document-frequency-from-scratch-in-python-6c2b61b78558. [Accessed 10 11 2021].

[63] E. C. D. Dios, “From DataFrame to N-Grams,” towardsdatascience, 23 05 2020. [Online]. Available: https://towardsdatascience.com/from-dataframe-to-n-grams-e34e29df3460. [Accessed 09 12 2021].

[64] L. K. A. E. O. A. A. N. K. Nidal Nasser, “n-Gram based language processing using Twitter dataset to identify COVID-19 patients,” *Sustainable Cities and Society,* vol. 72, 2021.

[65] A. C. a. C. F. C. A. Rosati, “Air Quality Monitoring in a BIM model by means of a IoT Sensors Network,” *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC),* pp. 110-115, 2020.

[66] P. M. S. W. A. T. W. P. Robert Nishihara, “Real-Time Machine Learning: The Missing Pieces,” *Proceedings of HotOS ’17, Whistler, BC, Canada,* 2017.