Design of a SCADA System Based on Open-Source Tools

1st Fernanda M Q Silva
Programa de Pós-Graduação em Engenharia Elétrica
Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
fquintaosilva@gmail.com

2nd Braz J Cardoso Filho
Departamento de Engenharia Elétrica
Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
braz.cardoso@ieee.org

3rd Igor A Pires
Departamento de Engenharia Eletrônica
Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
iap@ufmg.br

4th Thales A C Maia
Departamento de Engenharia Elétrica
Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
thalesmaia@cpdee.ufmg.br

2021 14th IEEE International Conference on Industry Applications (INDUSCON) | 978-1-6654-4118-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/INDUSCON51756.2021.9529709

***Abstract*—This paper presents the development of a supervisory system that uses Internet of Things (IoT) devices that can be implemented in any type of power plant. By applying the new concept of Internet of Things, it is possible to create an information sharing environment that simplifies and increases the effectiveness of supervisory and control systems. The goal of this paper is to present a supervisory system based on open-source tools for a photovoltaic power plant, using IoT devices and adapting non-IoT devices to the new generation of current automation (the trend of incorporating IoT concepts to processes). This new supervisory system improves the current supervisory system adding new features and increasing acquisition time. Besides the development of the new supervisory system, this paper also provides a comparison between the current supervisory system and the new supervisory system in order to highlight the main benefits of the latter.**

***Index Terms*—Internet of Things, supervisory system, solar power, Raspberry Pi, Node-Red, Emoncms**

I. Introduction

The main problems with renewable energy sources are cost and availability, for instance, wind and solar power are not always available when and where needed – the energy is not "dispatchable". Daily and seasonal changes affect greatly the generation of energy [1]. Control, supervision, and energy storage have been used to mitigate this type of problem and significant progress has been made throughout the years.

Traditionally, the supervisory system and data acquisition (SCADA) is a central computer that consists of inputs/outputs, communication devices, and software. It's a system used to monitor processes by collecting data from instruments, sensors, PLCs (programmable logic computers), and remote terminal units in order to present in a computer with a human interface an overview of the whole system. The supervisory system acts as a link between the control room and the field devices [2].

In [3]- [5], the authors develop a SCADA system by using a very traditional and common structure with data loggers for measurement and data acquisition. Then, they use a serial port to transmit the data to a user terminal, where the data is processed, analyzed, and used to develop a supervisory system that is only accessible in the specific portable computer. Due to some limitations related to serial communication, such as limited data transmission when data acquisition at high sampling rates is required, with technological advancements, it was possible to replace the serial interface with the establishment of a local network. In [6], the authors develop a supervisory system by implementing the more popular approach nowadays of establishing a local network. All the field devices are connected to remote terminal units or PLCs; these devices are connected to the local network and are able to make the data available for the network to which the monitoring terminal is also connected. This structure shows progress in regards to the other discussed here since it considers a LAN (local area network), which can be wired or wireless, for the building of a communication path between the field devices (RTU, Local Terminal, PLC) and the SCADA terminals. In order to be able to access the monitoring screens, it's necessary to have the supervisory installed in the machine and also to be connected to the LAN. In [7], the supervisory system developed shows a step towards innovation in this field. The authors also established a LAN to develop the communication interface as seen in [6], but the supervisory system is made available to remote users by using a publishing service; it's a step towards the implementation of the Internet of Things.

Improving and renovating the capabilities of monitoring and supervision can lead of a better usage of generated power, and also allow micro-grid operation. Nowadays technological advance in the industrial automation field plays a vital part to the monitoring and control of photovoltaic power plants, adding capabilities such as data processing and sending commands to the plant [8].

The advancement of technology in wired and wireless networks are a tendency nowadays. These changes are also present for industrial automation which incorporates the concept of Internet of Things. The Internet of Things (IoT) is an information sharing environment, in which the devices are connected to wired or wireless networks, making easy remote access viable. Considering the necessity of constant monitoring of variables related to the application (environmental and electrical data), applying the power of the Internet of Things to power plants seems extremely promising [9].

The goal of this paper is to present the control and monitoring of a photovoltaic plant. Applying the concept of the Internet of Things to the plant, automation, acquisition, and processing of data can be performed. Thus, this concept is used for the supervisory system development.

Currently, the analyzed solar power plant is monitored by a supervisory system developed by a third party company. This system is capable of presenting an overview of environmental and electrical variables of the system. It is possible to monitor graphically these field variables along any specified period of time. However, since this system is not easily customizable, such as adding or deleting variables, hardware, and self-processing algorithms further expansions would require trade agreement with this company. This scenario is detrimental for research. This system does not provide any information in regards to the current state of the devices nor does it show a schematic of the circuitry. Another setback when considering the current system is that, due to its structure, it does not allow a full implementation of a SCADA system.

The main idea is to provide a custom-made monitoring and control system of easy remote access of all environmental and electrical variables available. This supervisory system should be flexible – easy to incorporate expansions and/or changes -, adaptable to research, and more cost-effective, i.e. a system that surpasses the limitations cited above.

This paper is divided in five sections. Section 2 provides an overview of the methodology used in the development of the supervisory system. Section 3 describes the studied power plant (Usina Experimental Fotovoltaica TESLA Engenharia de Potência, Experimental Photovoltaic Power Plant TESLA Power Engineering in free translation) - its operation and its measurement system. Section 4 provides the comparison between the new and the current supervisory system. Section 5 presents the main conclusions of the paper.

II. Development of the Supervisory System

The drawback of self-developing a monitor and supervisory system is that computer service comes along with the system. Designing hardware besides software can become a tricky task. It involves a plethora of tasks: implementing backups tools, security monitoring, updating software, using UPS systems, and keeping considerable amount of hardware resources available for expansion if cloud computing is considered.

In order to overcome this challenge, a similar approach of container computer is used. Each individual service of this supervisory is allocated in a container like system. In other words, there is one resource responsible to acquire data from hardware, other to process data, other for user interface, and so on. Thus, a singular failure does not compromise the entire system. The main difference of the presented proposition over container computing is that dedicated hardware for each service is used. Instead of building services in a single multi-core hardware, several Raspberry Pi are used.

There are plenty of advantages when employing this modular architecture. By using individual servers for each service, in case of a failure in one of them, the error could be easily traced to the failed server, the failure could be analyzed, and the maintenance could be performed without affecting the other services. That means that an individual failure would not translate in a whole system failure. Besides this advantage, by using this structure, it's possible to allow parallel processing, meaning that all the services can happen concurrently and one process does not interfere in the others. Since the services are done in separate servers, if there is a blockage in one of the processes, this blockage does not halt the other services. Using this modular architecture, it's possible to see that each server is powered by an individual UPS (uninterruptible power supply), which increases the efficiency and reliability of the whole system.

The Raspberry Pi is a series of compact single-board computers. There are many advantages of using a Raspberry Pi. They are cost effective (low cost with a significant processing power), present a significant processing power in a compact board, are integrated with various interfaces (HDMI, USB, Ethernet, onboard Wi-Fi, Bluetooth), and they support Linux and Python, facilitating the development of applications [10].

To establish the supervisory system, it is necessary to implement some services in order to gather data, process it, and display it in the screens of the supervisory system. To this end, it is necessary to establish communication between the field equipment and the application (supervisory system), data storage, and a backup system. Three servers are established using the Raspberry: communication server, database and application server and the backup server as seen in Fig. 1.

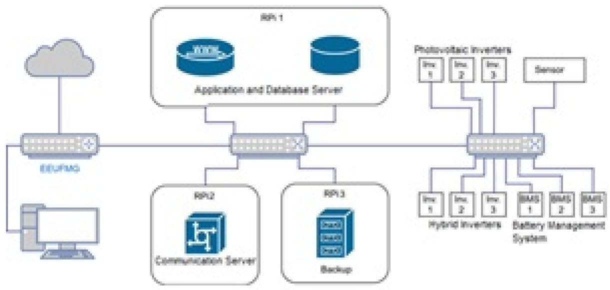

Fig. 1. Schematic of the proposed server architecture for the supervisory system.

Due to the limitations listed previously, another set of tools has to be used in order to provide the power plant with a flexible supervisory system that is easily expandable. Since it is interesting to use the concept of the Internet of Things when developing the new supervisory system, the goal is to find tools that are already integrated to IoT.

There is an array of open-source software options that would fulfill the requirements. The one that is the most popular and the most versatile is Node-Red [11]. Another tool that would be a good complement to Node-Red to provide the power plant with a complete supervisory system is Emoncms [12], that is frequently used to monitor generated and consumed energy and to develop dashboards [13]. Both are free Web-Service platforms already integrated with the IoT concept. Node-Red is a tool based on visual programming flow to connect hardware devices, APIs and online services. Emoncms is an application to process, register and visualize energy and other environmental and electrical data.

A. Communication Server

The communication protocol used for the communication between the computer and the inverters and the meter is the Modbus TCP/IP. The Modbus protocol is a highly used flexible open message structure used for the communication between master (client) and slave (server) [14]. The way this communication is performed guarantees that the final recipient receives the message: the master knows the request has reached the slave and the slave knows the master has received the response. One important disadvantage of Modbus is that it does not incorporate security resources [15].

Besides the meter and the power inverters, there are gauges in the power plant that are responsible for measuring the relative and absolute humidity, the relative and absolute pressure, the ambient temperature, the wind direction and the global irradiance – the solar meter station. There are also gauges to measure the temperature and the radiance of a reference cell. Since these devices are not integrated to the IoT concept, they communicate with an intermediary device via Modbus RS-485 (Modbus via serial communication). Via the file transfer protocol (FTP), the data from these gauges is also available in the communication server.

In order to post the data to the application server, the protocol used is the MQ Telemetry Transport (MQTT) protocol. MQTT is a communication protocol based on a publish-subscribe architecture. This protocol has implemented security layers; it requires login and password for broker access and for message exchange. And, even though it can implement complex security measures, it is a lightweight protocol [15].

All the protocols are implemented via the Node-Red environment. Node-Red is equipped with nodes to perform the communication necessary in a steady and reliable way.

B. Application and Database Server

The processed data is sent to the Emoncms dashboard to be monitored. The Emoncms environment gives the possibility to create custom dashboard in order to provide flexibility regarding the type of dashboard to be generated.

The idea is to generate a dashboard in Emoncms that represents the single-line diagram of the system, including the photovoltaic solar power plant. Also, this dashboard has to be able to provide further information about the electrical variables in a graphic format if the user wants to analyze real-time and historical data.

The dashboard is primarily developed in the Emoncms platform since it provides a user-friendly interface to build a custom-made supervisory system, providing some built-in gadgets and allowing customization. Besides, due to the features that Emoncms offers to its users related to its inputs and database storage, the platform is extremely adaptable to fulfill any requirements and it is easily expandable – to insert new variables, devices and/or features.

This versatility of Emoncms is an important benefit of the platform specially when comparing it to the current supervisory system available in the solar power plant. For the latter, it is cumbersome to add any new features and/or variables to it, since the system is developed by a third party company and the source is not available. Besides, the available structure of the current supervisory system is more rigid while Emoncms provides an environment that allows for any type of required dashboard.

The new supervisory system developed in Emoncms allows the inclusion of this overview besides allowing the creation of any necessary graphics and visual gadgets required to fully inform the user of the current status and operation of the power plant.

The input data from the communication server is configured to be stored in the built-in database. The Emoncms environment provides a database storage tool that does not rely on MySQL in order to minimize the processing time when retrieving historical data.

III. Case Study

The photovoltaic (PV) plant analyzed in this paper is the Usina Experimental Fotovoltaica TESLA Engenharia de Potência (Experimental Photovoltaic Power Plant TESLA Power Engineering in free translation). The capacity of the power plant is 37kWp and it is located in Belo Horizonte, Brazil. Fig. 2 shows the control room of the power plant. Fig. 3 is the single-line diagram of the solar plant. There are two power inverters in the plant: a Fronius IG Plus 150V-3 (10 kW) and a SMA Sunny Tripower 12000TL (12 kW). The power plant has 152 PV panels (Yingli 245P-32b, 245W). The output voltage of the power plant is 400Vac. The loads fed by the solar power plant are illumination and computers.

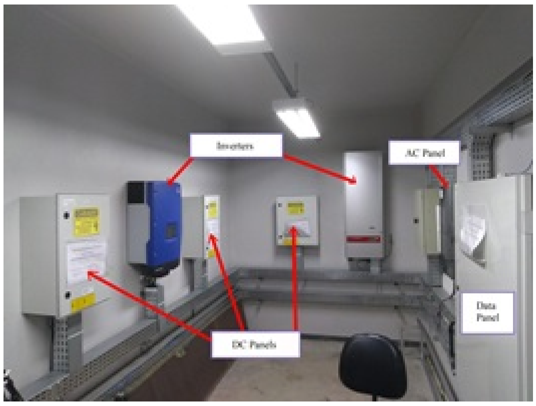

Fig. 2. Control Room.

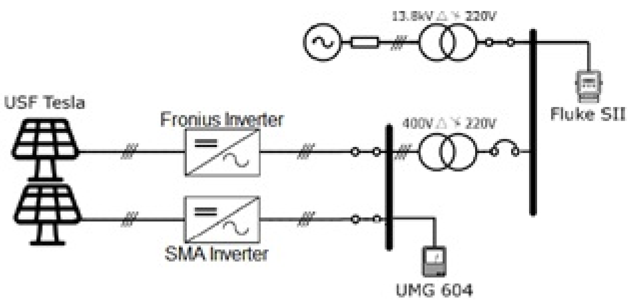

Fig. 3. Single-line diagram of the solar power plant.

All the power inverters are IoT (Internet of Things) ready. Using standard industrial TCP/IP Modbus protocol, the inverters are easy to be incorporated to the network and display their measurements in the Web when properly configured. They all communicate with a device that stores and makes the data available via a private portal. Every parameter related to each inverter can be recovered via this cloud.

There is also a Janitza energy meter, UMG 604, capable of measuring voltage, current, power and energy installed in the output of the power plant before the 400/220V transformer.

There is also a Fluke SII energy meter capable of measuring voltage, current, power, energy, frequency and harmonics installed in the 220V busbar before the 13.8k/220V transformer.

IV. Results and Analysis

The current supervisory system can be seen in Fig. 4. It depicts an overview of the current main electrical and environmental measurements of the solar power plant. It can be seen that the structure of the current dashboard is rigid, not allowing for changes in the type of information that is shown and how it is shown. There is also an Evaluation page, seen in Fig. 5, that provides the option of plotting different electrical and environmental variables. The new supervisory system has already been assembled and has been in operation for more than a year. Fig. 6 shows the box containing the servers (Raspberry Pi 3 B+), the router, and the communication connections. The new supervisory system shows voltage, current, power, and energy measurements from the AC and DC sides besides the environmental measurements from the solar meter station. The screens of the new supervisory system are seen in Fig. 7, Fig. 8, and Fig. 9. As it can be seen, the structure of the Emoncms dashboard is much more flexible than the structure of the current supervisory system, allowing for any type of data and format to be shown for the user and also by allowing the user to develop any type of dashboard without structural limitations. Besides the fact that Emoncms provides more flexibility and adaptability than the current supervisory system as seen by the previous

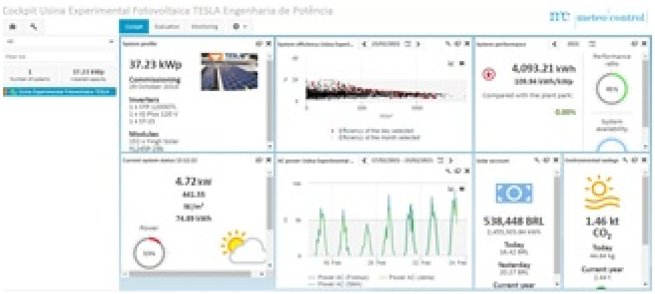

Fig. 4. Main screen of current supervisory system.

Fig. 5. Graphs page of current supervisory system.

Fig. 6. Supervisory system hardware box.

Fig. 7. Main screen of the new supervisory system: Overview of the solar power plant.

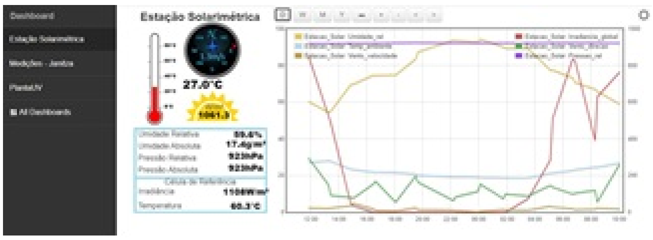

Fig. 8. Screen for the measurements of the solar meter station.

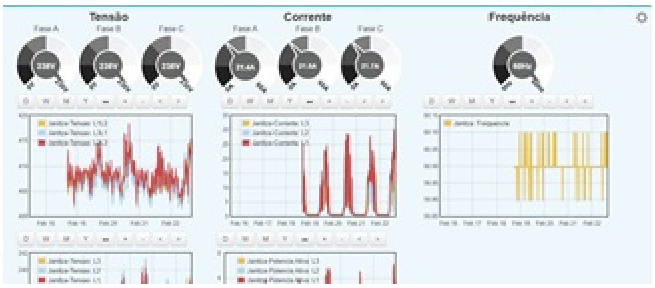

Fig. 9. AC-Grid measurements for the solar power plant.

images, the Emoncms environment allows for a finer resolution. The user can configure the new system to update its database every 5 seconds if necessary, whilst the current supervisory system only allows for new readings every 5 minutes. This polling time of 5 minutes could mean loss of measurement data especially in days with a high variability in weather conditions. Fig. 10 shows the plot of the SMA active power stored in the database of the new supervisory system (developed system) and of the SMA active power stored in the database of the current supervisory system (original system). Fig. 11 depicts a zoomed view of the data displayed in Fig. 10; it depicts the morning of this particular day in order to highlight the difference in accuracy regarding polling time. The data shown is from a window of time during the day. As it is seen in the graph, the new supervisory system is capable of detecting higher values of peaks of power and lower values of valleys of power since the polling time is lower than in the current supervisory system; 10 seconds versus 5 minutes. The original system also loses some variations that happen during the time frame.

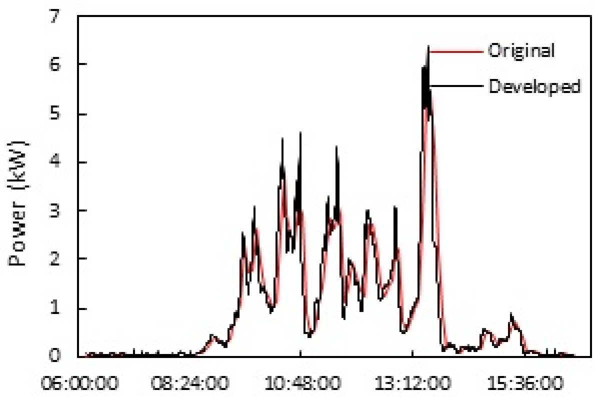

Fig. 10. Active AC power measured at the SMA Inverter.

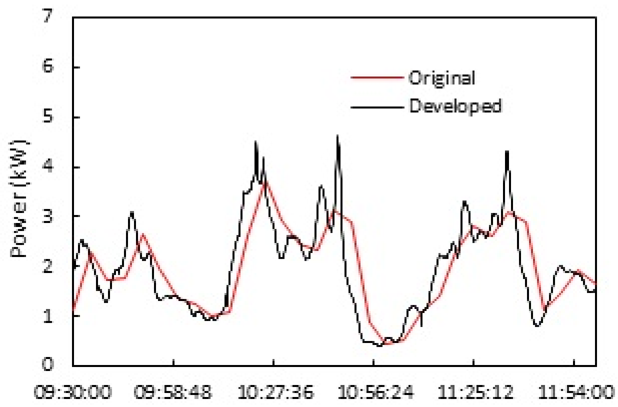

Fig. 11. Zoom of the active AC power measured at the SMA Inverter during the morning.

V. Conclusions

The main goal of this work was to develop a supervisory system for a power plant using the concepts related to the Internet of Things, employing devices that were network ready and adapting the ones that were not. In order to exemplify the usage of this supervisory system, it was applied to an existing solar power plant installed in the Escola de Engenharia at the Universidade Federal de Minas Gerais. However, it is important to notice that all the knowledge and the techniques used during the implementation of the supervisory system could be easily transferred not only for any power plant (with any type of source) but also for any industrial/residential plant as long as it is possible to establish a network in the plant.

Using devices integrated to IoT (such as the power inverters used in the power plant) and adapting the devices that are not integrated (the solar meter station connected to an intermediary device for example), it was possible to collect all the available data of the solar power plant. With this data accessible in the network, making it available for the selected microcomputer, Raspberry Pi, it was possible to develop different servers: communication and database servers that could help assembly the application server that houses the supervisory system per se.

Comparing both supervisory systems, it was possible to see that the new one that employs Emoncms is a more flexible solution; it's possible to adapt the developed supervisory system to accommodate any changes and/or requirements. That means that if there is a need to add new variables and/or features in the dashboard, it is easier, more cost-effective, and more efficient to implement the changes in the new supervisory system. Also, the new supervisory system allows the user to configure smaller polling times than the current one. The user is able to configure polling times more suitable to each electrical and environmental variable.

In terms of improvements for the developed supervisory system would be to integrate a prediction of the energy generation in order to be able to schedule the operation of a future energy storage system. That way, it would be possible for the software to predict the power output of the system based on the weather forecast for the next day and send commands to the field devices for when to charge/discharge the battery bank based on the forecast, fine-tuning in real time during the actual operation.

It would also be a valuable improvement to develop an optimization tool for the automatic control of the plant, with the goal of maximizing the efficiency of the energy generation and minimizing costs and losses.

Acknowledgment

The authors gratefully acknowledge the financial support from the RD project "Desenvolvimento de Sistema Smart Battery e Planta Piloto de Armazenamento de Energia Associado à Geração Distribuída de Energia Elétrica" (development of a smart battery system and a pilot plant of energy storage associated to distributed generation of electric energy in free translation), ANEEL 21/2016, PD-00553-0046/2016, Petrobras.

References

[1] E. Camacho, T. Samad, M. Garcia-Sanz, and I. Hiskens, "Control for renewable energy and smart grids," The Impact of Control Technology, v. 6, pp. 19-25, 2011.

[2] N. Raghvendra, Blog Post: "SCADA system – components, hardware, software architecture, types," Electricalfundablog. 2019. Electronic Archive. Available: https://electricalfundablog.com/scada-system-components-architecture/. Access: 12 January 2021.

[3] N. Forero, J. Hernández, G. Gordillo, ”Development of a monitoring system for a pv solar plant,” Energy Conversion and Management, v. 47, n. 15-16, pp. 2329-2336, Jan. 2006.

[4] R. Mukaro, and X. Carelse, ”A microcontroller-based data acquisition system for solar radiation and environmental monitoring,” IEEE Transactions on Instrumentation and Measurement, v. 48, n. 6, pp. 1232-1238, Dec. 1999.

[5] M. Benghanem, and A. Maafi, ”Data acquisition system for photovoltaic systems performance monitoring,” IEEE Transactions on Instrumentation and Measurement, v. 47, n. 1, pp. 30-33, Feb. 1998.

[6] C. Dumitru, and A. Gligor, ”SCADA based software for renewable energy management system,” Procedia Economics and Finance, v. 3, pp. 262-267, Oct. 2012.

[7] P. Papageorgas, D. Piromalis, K. Antonakoglou, G. Vokas, D. Tseles, and K. G. Arvanitis, ”Smart solar panels: in-situ monitoring of photovoltaic panels based on wired and wireless sensor networks,” Energy Procedia, v. 36, pp. 535-545, 2013.

[8] N. Forero, J. Hernández, and G. Gordillo, ”Development of a monitoring system for a PV solar plant,” Energy Conversion and Management, v. 47, n. 15-16, pp. 2329-2336, Jan. 2006.

[9] S. Adhya, D. Saha, , A. Das, J. Jana, and H. Saha, ”An IoT based smart solar photovoltaic remote monitoring and control unit,” International Conference on Control, Instrumentation, Energy Communication (CIEC), v. 2, pp. 432-436, 2016.

[10] E. Chen, G. Chow, C. He, and J. Tien, ”Bus 237 E102 group 1 Raspberry Pi”. 2020. Electronic Archive. Available: https://sites.google.com/site/bus237raspberrypi/home. Access: 12 January 2021.

[11] O. Foundation, Node-red, 2016. URL: https://nodered.org/.

[12] OpenEnergyMonitor Project, Emoncms, 2011. URL: https://emoncms.org/.

[13] M. Benammar, A. Abdaoui, S. Ahmad, F. Touati, and A. Kadri, ”A modular IoT platform for real-time indoor air quality monitoring,” Sensors, v. 18, n. 2, pp. 1-18, Feb. 2018.

[14] C. M. Freitas, ”Protocolo Modbus: fundamentos e aplicações”. 2014. Electronic Archive. Available: https://www.embarcados.com.br/protocolo-modbus/. Access: 12 January 2021.

[15] F. C. de S. Silva, Artigo: ”Modelos cliente-servidor do Modbus TCP e publish –subscribe do MQTT: quando utilizar cada um deles e quais suas vantagens e desvantagens?” 2019. Electronic Archive.