Rev. Roum. Sci. Techn.– Électrotechn. et Énerg.
Vol. **71**, *1*, pp. 133–138, Bucarest, 2026

Automatique et ordinateurs
Automatic Control and Computer Sciences

ENHANCING SMART BUILDING RELIABILITY AND DATA UCONTINUITY USING EDGE COMPUTING

MARIAN-COSMIN DUMITRU[1], RADU-NICOLAE PIETRARU [2], MIHNEA-ALEXANDRU MOISESCU [3]

Keywords: Smart building; Edge computing; Internet of Things (IoT).

Starting from the increase in the number of IoT devices inside buildings, devices that are integrated are the basis of the smart building concept, thus resulting in the exponential growth of the volume of data generated. Such an implementation is feasible but presents certain limitations in terms of the time required for data processing and due to the management of data queues, thus resulting in data losses that can lead to erroneous decisions inside buildings. The article proposes an integration between smart buildings and Edge computing, an integration that aims to demonstrate the benefits of this approach. The system is implemented through the ThingsBoard Edge platform and facilitates access to information generated by several buildings on the campus of the National University of Science and Technology Politehnica Bucharest. To present the benefits of this implementation, a two-month reference period was considered, during which we simulated, in parallel, a smart building infrastructure with and without an integrated edge node. Results confirm superior data continuity for the platform with an integrated edge node; the dependence on an internet connection no longer limits the system's capacity in the absence of a stable connection.

1. INTRODUCTION

ThingsBoard is an open-source IoT platform that provides data collection and real-time analytics, offering compatibility with both Cloud and Edge Computing. It is the portal through which the collected information from various sensors located in the buildings within the campus of the university can be accessed

By fostering the use of Edge computing, the goal is not to replace the Cloud architecture, but rather to add a new layer that will strengthen the system's capacity [1]. This platform constitutes the environment in which the tests were conducted for this article, which aims to show the benefits of edge computing in smart buildings compared to a classic smart building infrastructure, in which the data is transmitted directly to the platform that processes them. In our case, the data processing and display mechanism is the open source Thingsboard platform, the platform that gives us the ability to integrate IoT devices and visualize collected data.

The concept of smart building refers to the introduction of necessary devices inside buildings, so that the building responds to the user's needs without the need for repeated interventions from the end user. [2]

In the context of Smart building, Edge Computing can be a viable solution that brings new capabilities to the system [3], having multiple areas of use such as resource management or electricity consumption management.

A main problem that can arise in the case of a classic architecture is the moment when internet connection problems occur for certain periods of time, whether the discussion concerns an outage or the use of the entire available bandwidth, the result is the same: data loss. In these cases, for an Edge architecture, the edge nodes become the main nodes, having direct connections between them, continuous and process [4]. In the absence of such an implementation, the tracked metrics will have certain intervals in which data was not collected, thus resulting in the alteration of the available information, decisions made based on this information being erroneous over the time intervals considered, also affecting graphs over longer time intervals or daily averages.

Starting from this issue, the article proposes to analyze the case where a new architectural approach is pursued based on the addition of a new component, the Edge node. This is responsible for collecting and managing the data generated by the sensors; subsequently, the information is synchronized with the main server, thus eliminating the problem of data loss due to the large volume generated in a unit of time. Further discussion considers certain specific cases, such as the lack of a direct internet connection; communication is carried out directly between the local edge node and the sensors inside the building.

The approach chosen for this work is one that focuses on the practical side, analyzing the information generated by various sensors. Those sensors can follow both the principles of a classic infrastructure and the edge ones, thus highlighting the importance of the presence of the edge node within a smart building system.

In the next chapter, starting from existing implementations and research, an analysis will be conducted on the existing work, the benefits brought to a classic infrastructure, as well as the areas that can be improved. In the third chapter, the proposed architecture will be detailed along with the components used to create the system, hardware and software. The study continues in the fourth chapter with the comparative part and the analysis of the performances between the classic and edge infrastructure, comparing the results obtained over a period of 60 days, the period in which various scenarios were simulated, such as the interruption of the internet connection. In the last chapter, the conclusions will be presented, the argumentation of the decision to use edge computing in smart buildings, and future research directions, which tend towards the automation of decision-making processes through machine learning.

2. RELATED WORK

The introduction of Edge computing in smart buildings is a much-discussed topic that has a wide applicability, whether the reference is to monitoring and optimizing electricity consumption[5], analyzing thermal images to estimate the number of occupants/users inside buildings [6], smart grid [7] or even managing an HVAC system [8], in all these cases edge computing plays a very important role both as a technical solution that helps improve performance, but also in improving the sustainability of the targeted buildings.

Consistent with the approach in [9], our study does not focus on a specific feature of smart building, regardless of the type of data generated, being more oriented on the benefits offered by the overlap between Edge Computing,

[1] National University of Science and Technology Politehnica Bucharest, Romania.
E-mails: marian.dumitru@stud.acs.upb.ro, radu.pietraru@upb.ro, mihnea.moisescu@upb.ro

DOI: 10.59277/RRST-EE.2026.1.22

IoT devices, thus being "the optimal approach "[10]

Regardless of the size of the buildings and the intended use of the building (office, industrial, or residential), adding the edge component to the system aims to improve the quality of services compared to a classic cloud infrastructure [11].

For all types of buildings, Edge computing comes with several major advantages, one of which is represented by reduced costs [12], an area that can be materialized by introducing protocols such as Zigbee [13]. Moreover, it enhances the system's flexibility in the ease of integrating artificial intelligence and machine learning technologies [14], thereby contributing to improvements in both overall performance and security. In case of implementations that are in advanced stages of development, only the introduction of the new node is necessary. The rest of the components remaining unchanged, managing to help with the fastest possible data processing and returning information to the user for decision-making, at the same time eliminating the need for a constant high-speed internet connection, data being processed closer to the source [15], data synchronization between the cloud server and the edge server being achieved when the internet connection allows it.

Regardless of the nature of IoT devices and the parameters generated by them, there may be several technical solutions that can improve the presented system performance such as Fog Computing, having "a fully distributed, multi-tier cloud computing system" [16], Intelligent Edge Computing "representing a natural evolution of network" [16] or multi-access edge computing (MEC), which is distinguished by its tight integration with 5G network standards [17]. Another technique that can be considered is based on cloud resources, using load-balancing mechanisms that allow dynamic scaling of resources based on load [18]. As introduced earlier, in this article, the ability of Edge computing to provide better granularity is analyzed, reducing errors due to data that is not transmitted correctly, bringing other benefits such as improving response times in systems that require real-time monitoring.

Installing edge nodes at building or floor level, depending on the size of the buildings and the volume of data generated, offers an improvement in response time compared to a classic centralized implementation through a cloud provider, mainly by reducing latency. Such a comparison of the performances between a centralized and an Edge node was addressed in [19], where the authors managed to obtain notable performances, obtaining average response times 250ms faster for small buildings and up to 700ms for large buildings [19].

The most important capabilities in using Edge computing are its scaling capacity, being a sure way for a scalable system by adding new edge nodes [20] as well as reducing data redundancy, which has the benefit of optimizing storage space [21], providing isolation capacity in the event of a failure.

At the same time, Edge implementations offer the possibility of future integrations, with multiple possibilities for improvement, such as integrations with Federated Learning "is a decentralized ML that trains an algorithm across multiple devices" [22] or the introduction of artificial intelligence by creating Edge AI platforms that help us collect and analyze data generated by sensors in real time [23].

3. PROPOSED ARCHITECTURE

Starting from the already existing architecture [24] located in the buildings on the campus of the National University of Science and Technology Politehnica Bucharest, it is necessary to highlight the advantages and improved performances offered by the new solution proposed in the article.

In addition to the IoT devices that generate data from the entire building, the presented architecture is based on two critical components, with well-defined tasks: the main server that underlies the digitaltwin.upb.ro platform and the edge node, the two being connected.

The main server (virtual machine) is based on an Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz, 4 cores with 4GB RAM, its operating system is CentOS Stream Linux 8 with kernel-4.18.0-553.6.1.el8.x86_64. The collected information is stored in the Postgres database, which uses the PostgreSQL version 12.22. On this system, the ThingsBoard interface is configured, which uses ThingsBoard version 3.8.1 together with Java OpenJDK 17.0.12, which is one of the tools required to install ThingsBoard. This server is accessible from outside the network; a stable internet connection is required to access the platform.

The Edge server is represented by a Raspberry Pi 2B development board based on an ARM Cortex-A7 @ 900MHz processor with 1GB RAM. The operating system used is Raspbian Linux 12 bookworm, used together with Kernel 6.6.62+rpt-rpi-v7. The server has been provisioned with the ThingsBoard Edge 3.8.0 Platform together with Java OpenJDK 17.0.13, and the database is PostgreSQL 15.9. This server can only be accessed via the local network, without an open connection to the internet.

As a data source, two sensors were also used within the system: the first, AIR811_PR205 was used to transmit data directly to the cloud, and the second AIR811_PR205_edge was used together with the Edge component.

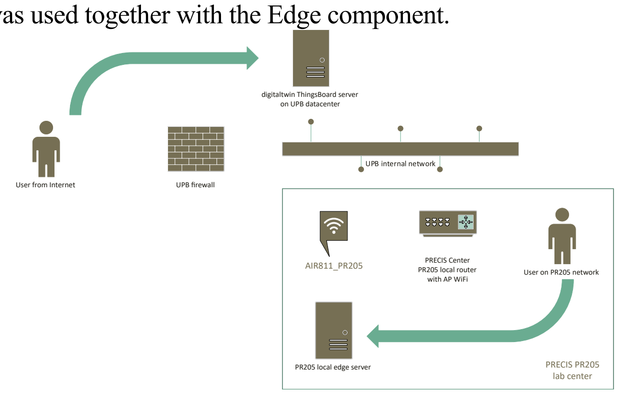

Fig. 1 – Edge computing solution diagram.

Figure 1 presents the proposed network architecture based on the edge component, the platform being accessible both locally and from outside the network. To establish the connection, users connect to the web interface via the HTTPS protocol, using the standard port of the ThingsBoard platform 8080. Communication between the sensors and the Edge platform is achieved via the MQTT protocol with QoS 1(at-least-once delivery), using port 1833, and between the Edge node and the server, there is bidirectional communication, using the RPC protocol together with port 7070.

Any user has access to the information collected from the sensors and their visualization with minimal delay. Also, if there are problems with the internet connection and the data can no longer be transmitted to the Thingsboard server running in the UPB datacenter, the edge node continues to receive the data generated by the IoT devices, synchronizing with the main server when the connection is restored, leveraging the edge configuration of the platform. At the same time, regardless of the status of the internet connection, the Edge server is available and accessible by users using the local network in PR205, and the information transmitted by the IoT devices is accessible at any time. Data buffering was implemented exclusively on the edge node, which performs local storage of telemetry during outages, while sensor-side buffering was avoided to limit device memory usage and power consumption.

An edge component used is the AIR811_PR205_edge sensor that was integrated into the ThingsBoard edge platform; its characteristics are detailed in Fig. 2.

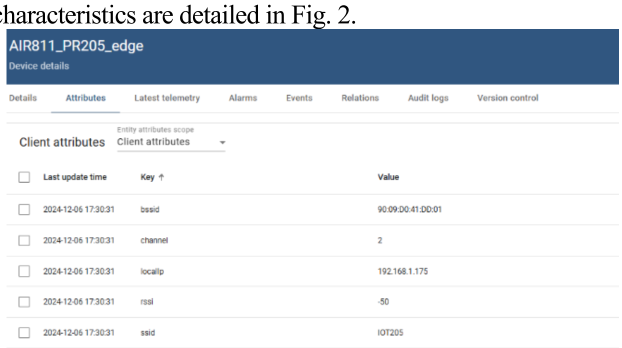

Fig. 2 – AIR811_PR205_edge sensor configuration.

The flow chart below presents the data transition process in the two scenarios: the case where the sensor communicates directly with the cloud provider and the case of integrating an edge node that processes the data sent by the sensor and communicates further with the cloud node.

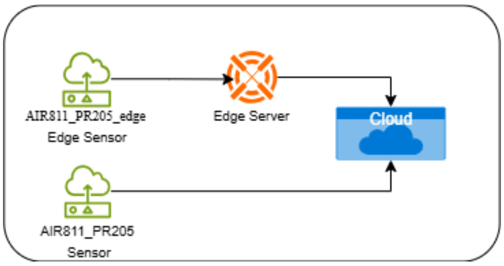

Fig. 3 – Data flow chart.

The figures below (Fig. 4 and 5) present the connection made between the edge node and the main server and the IoT device, without any lost packets in making the connections.

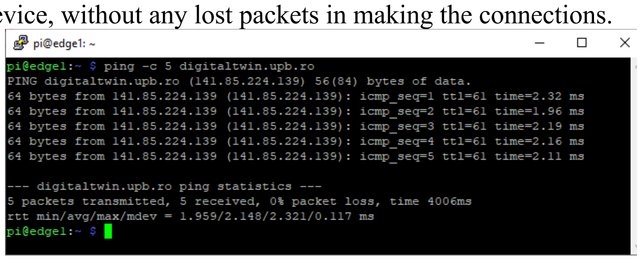

Fig. 4 – Ping edge to main server

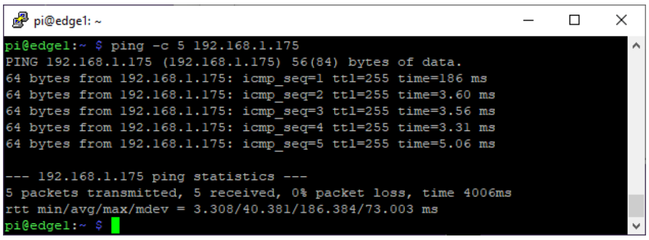

Fig. 5 – Ping edge to IoT edge device

In terms of communication performance for the tests in Figs. 4 and 5, in the case of the connection between the edge server and the main server, the average response time is 2.148ms, while for the ping case between the Edge server and the edge sensor, the average response time is 40.381 ms. Although the edge server and the edge sensor are in the same vicinity, the response time is higher than the connection between the edge server and the main server, this is due to the way the connection is made; in Fig. 5, it is a WiFi connection, while in Fig. 4, it is a high-speed fiber optic connection (the university's internal network).

4. EDGE COMPONENT EVALUATION

To evaluate the advantages of the proposed solution, a monitoring period of 60 days was considered a sufficiently long-time interval to be able to verify the differences between the classic architecture in which the sensors are connected directly to the main server and the proposed solution, the use of the edge node. To be able to perform this test, two sensors named AIR811_PR205 and AIR811_PR205_edge were used, the first being dependent on the connection to the UPB server, the second being connected to the edge server. Both sensors generate the following air quality metrics: Temperature, Humidity, Volatile Organic Compounds (VOC) Index, Nitrogen Oxides Index, Particle Matter 1, 2, 2.5, 4, and 10. The total message size transmitted by the sensor is approximately 220-350 bytes per telemetry message. To compare how the two IoT devices behave in case of network problems, the internet connection being affected, a connection problem between the PRECIS PR205 laboratory network and the UPB network was triggered by disconnecting the PRECIS PR205 laboratory network from the UPB network on 06.02.2025 between 14:30 and 15:00.

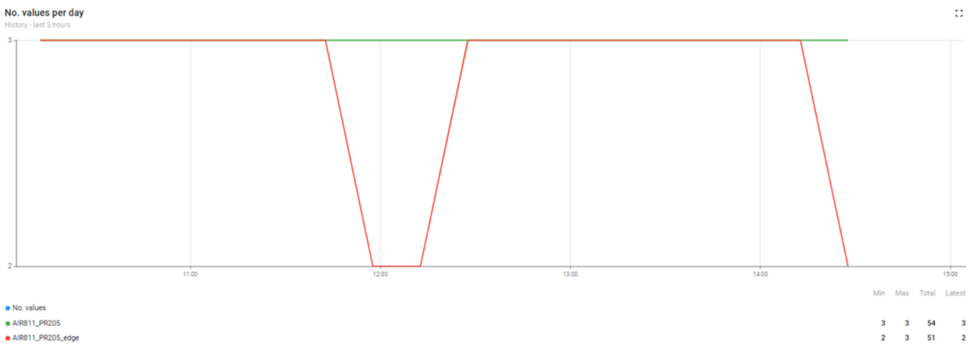

Fig. 6 – Screenshot with number of values sent by the two sensors in the central interface.

In Fig. 6, in the reference interval, both AIR811_PR205 and AIR811_PR205_edge devices are affected at the central server level. The ThingsBoard platform does not receive and cannot display data from either device.

Although the connection is interrupted in Fig. 7, the IoT device AIR811_PR205_edge sends data to the edge server via the PR205 laboratory's local network, as shown by the constant rate at which data is received, thus guaranteeing much better data consistency and continuity. By having the full set of data available the described analysis won't be affected by any downtime caused by incidents. Once the issue is resolved, all the data will be synchronized with the main server.

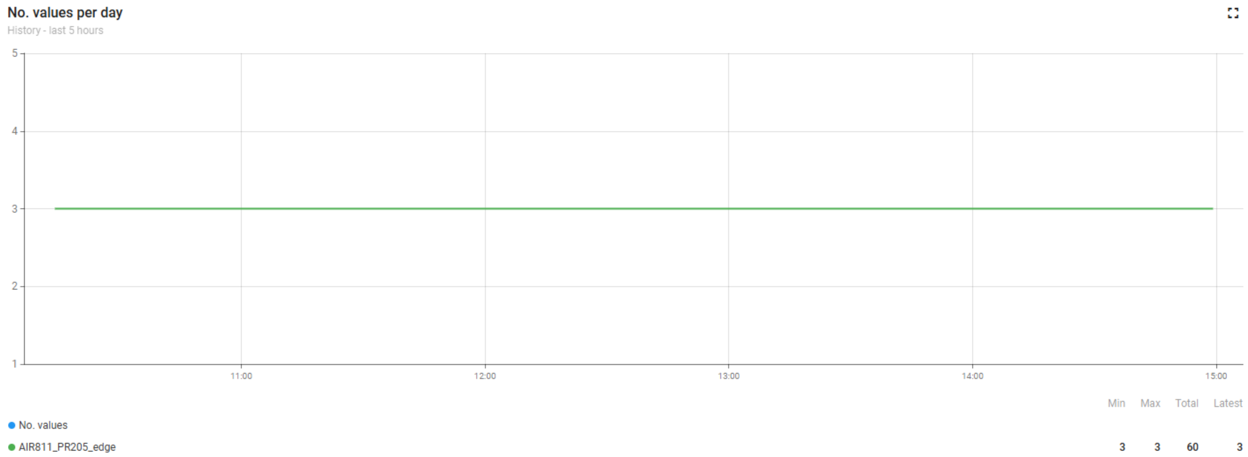

Fig. 7 – Screenshot with the number of values sent by the AIR811_PR205_edge sensor in the edge interface.

The generated data is transmitted at a standard interval of 5 minutes; the maximum number of values collected in a day without incidents is 288 values per day. The graphs presented show the total number of values transmitted in a day of the reference interval.

All data from the AIR811_PR205_edge device is accessible through the local dashboard of the edge device, even if the connection to the main server has been interrupted. Figure 8 shows the data read from the edge interface at the time of the test.

Although MQTT QoS 1 ensures at least once delivery, there may still be gaps in telemetry for the sensor that communicates directly with the cloud node. These gaps typically occur due to network disruptions during message transmission or brief platform restarts. The introduction of the Edge node mitigates this issue by storing messages locally and forwarding them when connectivity is restored.

After restoring the connection to the UPB network, the data stored by the edge device related to the AIR811_PR205_edge sensor is transmitted to the ThingsBoard central server, behavior shown in Fig. 9. Since the sensor generates data every 5 minutes, this results in a backlog of 12 messages per hour for a single connection, resulting in a backlog of approximately 3.6 kilobytes of data per hour. Even for a 24-hour outage, the maximum backlog will not exceed 100 kilobytes, with the time required for synchronization being a few seconds, without impacting ongoing data collection or local operations.

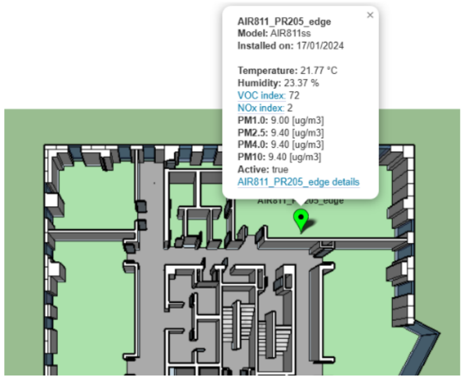

Fig. 8 – Screenshot of the local edge sensor dashboard AIR811_PR205_edge accessed from the PR205 lab network.

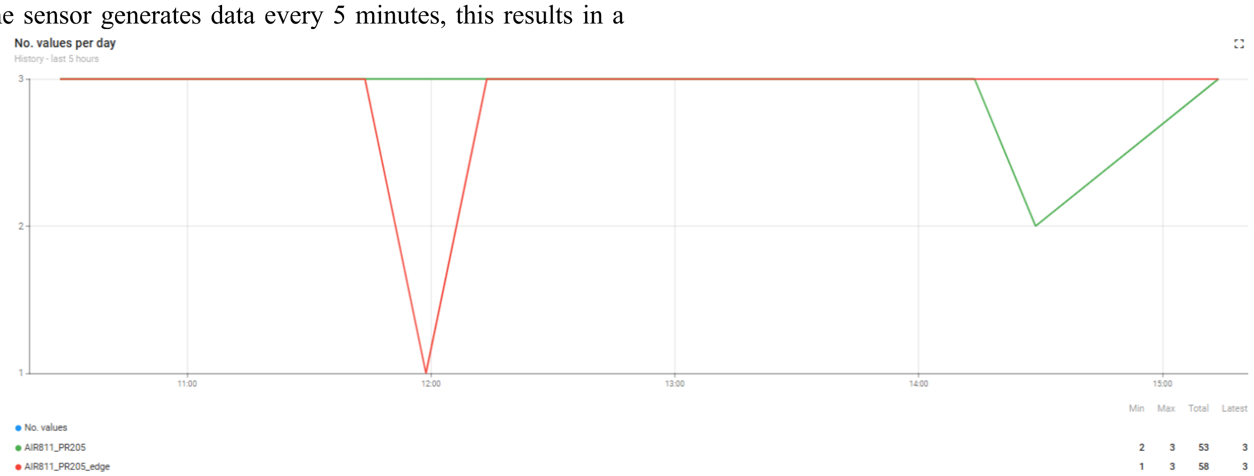

Fig. 9 – Screenshot with the number of values sent by the two sensors in the central interface after restoring the connection

Over 60 days, the edge device has a total of 17256 values sent, unlike the direct connection device, which has a total of 17243 values sent (Fig. 10). The missing values are entirely caused by errors during the process of transmitting them to the cloud node and cannot be categorized as random noise or sensor instability as the data was generated by the sensor, visible in the sensor logs and collected correctly by the Edge node. In the implementation presented, data loss can be seen; the scenario presented considers the comparison between the two sensors. Extrapolating to a case with more sensors or more communication failures, data loss can be much more serious, even leading to major losses in the recording of time series. However, it is essential to consider that during the considered period, there were no major incidents that affected the sensor's inability to transmit data to the main server; otherwise, the differences would have been very large.

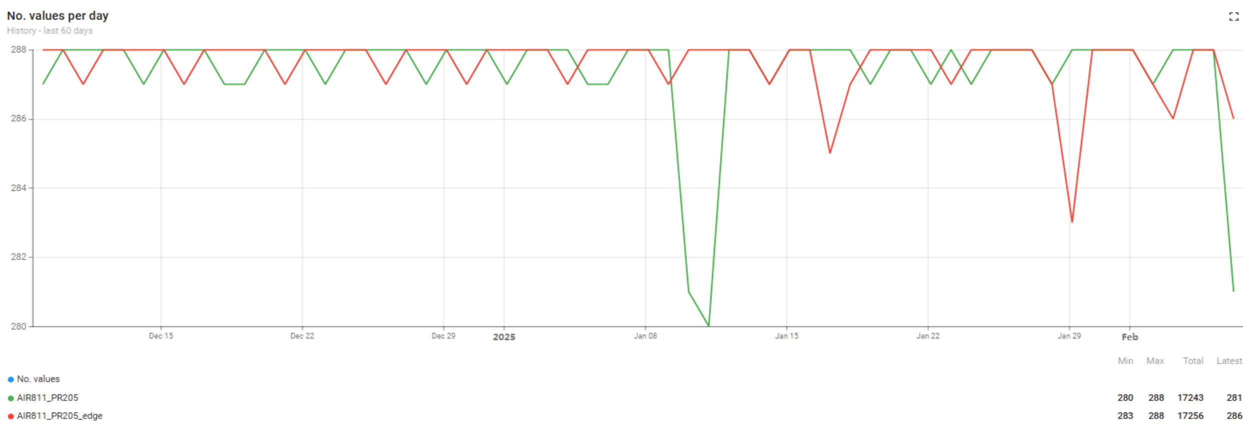

Fig. 10 – Screenshot with the number of values sent by the two sensors in the central interface over a period of 60 days (December 2024 – February 2025).

5. CONCLUSIONS

By evaluating the proposed architecture, a detailed examination of the use of Edge computing in smart buildings was conducted, focusing on the benefits brought by this implementation, as well as an extensive comparison between an edge architecture and a cloud server-based one.

Analyzing the results of the tests carried out for the proposed scenario, the main key points of using Edge computing in smart buildings can be observed: improving the data collection rate in a certain unit of time, the ability to receive information even in the absence of an internet connection, and the data being accessible immediately after restoring the internet connection.

By running the two types of connections in parallel over a defined period of 60 days, the quantitative differences in data generated, transmitted and processed between the two IoT devices were captured, differences that, depending on the granularity required for the information transmitted by the sensors, can translate into alterations in the data over certain time intervals, this scenario being very well covered by the introduction of the Edge node. These vary depending directly on the nature of the outage and the time required to return to the initial state.

As future research directions, this work seeks to add new functionalities and performance evaluations to the presented edge architecture, introducing testing the distribution of computing effort regarding data processing on the central cloud platform on the edge component. Another very important directive is studying the efficiency of distribution in the case of introducing machine learning algorithms in data processing that, based on the information collected, can make decisions automatically, based on user behavior, streamlining processes and resource consumption inside the building, thus making the transition to autonomous buildings.

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Author_1: 60%
Author_2: 30%
Author_3: 10%

Received on April 15, 2025

REFERENCES

1. A.S. George, A.H. George, T. Baskar, *Edge computing and the future of cloud computing: a survey of industry perspectives and predictions*, Partners Universal International Research Journal, **2**, *2*, pp. 19–44 (2023).
2. D. Kim, Y. Yoon, J. Lee, P.J. Mago, K. Lee, H. Cho, *Design and implementation of smart buildings: a review of current research trend*, Energies, **15**, *12*, 4278 (2022).
3. X. Chang, W. Li, C. Xia, J. Ma, J. Cao, S.U. Khan, A.Y. Zomaya, *From insight to impact: building a sustainable edge computing platform for smart homes*, 2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS), pp. 928–936 (2018).
4. S. Nayak, R. Patgiri, L. Waikhom, A. Ahmed, *A review on edge analytics: issues, challenges, opportunities, promises, future directions, and applications*, Digital Communications and Networks (2022).
5. A.S. Shah, H. Nasir, M. Fayaz, A. Lajis, A. Shah, *A review on energy consumption optimization techniques in IoT based smart building environments*, Information, **10**, *3*, 108 (2019).
6. A. Metwaly, J.P. Queralta, V.K. Sarker, T.N. Gia, O. Nasir, T. Westerlund, *Edge computing with embedded AI: thermal image analysis for occupancy estimation in intelligent buildings*, Proceedings of the INTelligent Embedded Systems Architectures and Applications Workshop 2019, pp. 1–6 (2019).
7. K.S.K. Veni, J.C.S. Thomas, J.P. Paul, J. Joyce, *IoT-based voltage control in smart grid with feed-forward control in automatic voltage regulator*, Rev. Roum. Sci. Techn. – Électrotechn. et Énerg., **70**, *1*, pp. 75–80 (2025).
8. P.K.D. Pramanik, S. Pal, M. Mukhopadhyay, P. Choudhury, *Sustainable edge computing with mobile crowd computing: a proof-of-concept with a smart HVAC use case*, The Journal of Supercomputing, **80**, *16*, pp. 23911–23994 (2024).

9. F.J. Ferrández-Pastor, H. Mora, A. Jimeno-Morenilla, B. Volckaert, *Deployment of IoT edge and fog computing technologies to develop smart building services*, Sustainability, **10**, *11*, 3832 (2018).
10. M. Chiozzotto, M.A. Ramírez, *What is the best solution for smart buildings? A case study of fog, edge computing and smart IoT devices*, Applied Sciences, **15**, *7*, 3805 (2025).
11. P. Kochovski, V. Stankovski, *Supporting smart construction with dependable edge computing infrastructures and applications*, Automation in Construction, **85**, pp. 182–192 (2018).
12. H. Yar, A.S. Imran, Z.A. Khan, M. Sajjad, Z. Kastrati, *Towards smart home automation using IoT-enabled edge-computing paradigm*, Sensors, **21**, *14*, 4932 (2021).
13. S.S. Madsen, B.E. Staugaard, Z. Ma, S. Yussof, B.N. Jørgensen, *A cost-effective edge computing gateway for smart buildings*, Energy Informatics Academy Conference, pp. 37–54 (2024).
14. A. Glavan, V. Croitoru, *Incremental learning for edge network intrusion detection*, Rev. Roum. Sci. Techn. – Électrotechn. et Énerg., **68**, *3*, pp. 301–306 (2023).
15. K. Rastogi, D. Lohani, *IoT-based indoor occupancy estimation using edge computing*, Procedia Computer Science, **171**, pp. 1943–1952 (2020).
16. M. Chiozzotto, M.A. Ramírez, *What is the best solution for smart buildings? A case study of fog, edge computing, and smart IoT devices* (2025).
17. A. Glavan, D. Gheorghica, V. Croitoru, *Multi-access edge computing analysis of risks and security measures*, Rev. Roum. Sci. Techn. – Électrotechn. et Énerg., **68**, *2*, pp. 206–211 (2023).
18. M. Jesi, A. Appathurai, M. Narayanaperumal, A. Kumar, *Load balancing in cloud computing via mayfly optimization algorithm*, Rev. Roum. Sci. Techn. – Électrotechn. et Énerg., **69**, *1*, pp. 79–84 (2024).
19. F. Lăcătușu, A.D. Ioniță, M. Lăcătușu, A. Olteanu, *Performance evaluation of information gathering from edge devices in a complex of smart buildings*, Sensors, **22**, *3*, 1002 (2022).
20. M.S. Aslanpour, S.S. Gill, A.N. Toosi, *Performance evaluation metrics for cloud, fog and edge computing: a review, taxonomy, benchmarks and standards for future research*, Internet of Things, **12**, 100273 (2020).
21. W. Yu, F. Liang, X. He, W.G. Hatcher, C. Lu, J. Lin, X. Yang, *A survey on the edge computing for the Internet of Things*, IEEE Access, **6**, pp. 6900–6919 (2017).
22. S. Márquez-Sánchez, J. Calvo-Gallego, A. Erbad, M. Ibrar, J.H. Fernandez, M. Houchati, J.M. Corchado, *Enhancing building energy management: adaptive edge computing for optimized efficiency and inhabitant comfort*, Electronics, **12**, *19*, 4179 (2023).
23. Y. Sahni, F. Xiao, S. Wang, *Edge AI platform for practical learning in introductory course on smart building technologies*, 2024 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE), pp. 1–5 (2024).
24. R.N. Pietraru, R.A. Crăciun, D.M. Merezeanu, *Rapid deployment of low-cost wireless monitoring solution for smart building*, 2024 International Conference on Automatics and Informatics (ICAI), pp. 377–380 (2024).