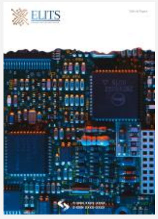
Electronics and Informatics Journal Vol. 1 No. 1 (2026)

P-ISSN: XXXX-XXXX
E-ISSN: XXXX-XXXX
ELITE Journal
journal.diginus.id/ELITS



Design and Simulation of an IoT-Based Library Noise Monitoring System Using ESP32 and ThingSpeak

Mutia Hikmah[1*], Dhiyaa Ul Haq M. Shadiq[2], Abdul Aziz[3]

[1*, 2] Department of Informatics Engineering, Faculty of Engineering, State University of Makassar, Jl. A. P. Pettarani, Makassar 90222, Indonesia,

[3] Faculty of Engineering, Hasanuddin University, Jl. Poros Malino Km. 6, Bontomarannu, Gowa Regency, South Sulawesi 92171, Indonesia.

E-mail correspondence: *thyaa2206@gmail.com*[1*]

ABSTRACT

This research details the creation and simulation of a noise monitoring system for libraries that is based on the Internet of Things (IoT), making use of an ESP32 microcontroller, a KY-037 sound detection device, the ThingSpeak cloud service, and a Mamdani fuzzy inference system. The main aim of this study is to constantly keep track of how loud it is and sort the levels into different noise conditions described in words, so that library supervisors can assess the acoustic setting and any possible disruptions in the areas where people read. In the designed system, the ESP32 acts as an IoT point that takes analog signals from the KY-037 sensor, changes these signals into decibel (dB) values, and sends the data to a safe ThingSpeak channel using a way of sending data that depends on events, which happens when noise levels go over certain set limits. Old noise data is taken out in CSV format and handled in MATLAB, where a Mamdani fuzzy model with three types of input (quiet, moderate, and noisy) and three levels of output (low, medium, and high) is put in place to make a noise index that goes from 0 to 100. The simulation results show that the ESP32–KY-037 setup sends noise data to ThingSpeak in a trustworthy manner without any breaks in communication, and the fuzzy output tracks the general pattern of the sensor data, giving a smoother way to sort noise. The suggested system can tell the difference between calm and potentially disruptive conditions, which backs up the idea of a smart library space with abilities for remote monitoring. This research shows that it is doable to put together ESP32, ThingSpeak, and fuzzy logic for noise monitoring that adjusts itself, and it lays the groundwork for future work that includes more sensor points and warning systems that work in real time.

ARTICLE INFO

Article History:

Received:
December 16th, 2025
Received in revised form:
December 28th, 2025
Accepted:
January 11th, 2026

Keywords:(10pt)

ESP32;
Fuzzy logic;
Internet of Things;
Library noise; ThingSpeak

1. INTRODUCTION

The library is one of the facilities that plays a very important role in supporting academic activities, especially in terms of reading and learning, so a quiet environment is needed so that visitors can concentrate effectively. Quiet is an important element in maintaining the comfort and concentration of visitors when doing activities in the library [1]. Noise itself can be interpreted as noise that is not expected to have a bad impact on people's health, well-being, and quality of life. In the realm of education, noise is a crucial factor that can affect the success of the learning process. Studies have proven that low noise levels can create a supportive atmosphere and make it easier for students to stay focused, while noise can increase the likelihood of hearing loss, reducing noise productivity is an important factor that affects the effectiveness of the learning process. Studies show that minimal noise can create a supportive environment for students to more easily improve concentration, while high noise levels can result in hearing damage and reduce work efficiency and even reduce concentration [2].

According to a decree issued by the Minister of Environment in 1997, the permissible noise level in a library environment is between 45 to 55dB. When the sound is in this range, the atmosphere in the library is considered balanced. However, in reality, the noise from visitors' activities often exceeds the set limit, resulting in discomfort for other visitors and even affecting concentration when reading [3] .

A previous study by Al Matsany Ahmad Najih & Martiani Ahmad, 2025 entitled *"Design and Build a Library Visitor Noise Detection Tool Using an Internet of Things (IoT)-Based Microcontroller"* states that systems that use ESP8266 NodeMCU and KY-037 sensors have the ability to detect noise in real time, provide precise response, and send automatic alerts via LCD, speakers and Blynk apps when noise exceeds 55dB. Despite the good performance, the method still carries out continuous data transmission with deterministic logic based on fixed thresholds, making it less efficient in terms of transmission and not sufficiently adaptive to changes in noise in dynamic environments [4].

To overcome these limitations, the study proposes an IoT-based noise monitoring system in libraries using ESP32 and the KY-037 sensor connected to Thingspeak, and implements two major improvements. First, using an event-driven threshold-based delivery method, data is only sent when noise levels exceed 55 dB or 70 dB to improve network efficiency without sacrificing detection reliability. Second, the application of the fuzzy logic of the Mamdani model to overcome uncertainties in acoustic signals and gradually classify noise conditions (e.g. "quiet", "moderate", "very noisy"), so that system decisions can strengthen the implementation of smart libraries through efficient *real-time* monitoring in network usage as well as more accurate noise level assessments [5].

Referring to that explanation, a method is needed that can not only identify noise, but also connect with the idea of a smart library based on the Internet of Things (IoT).

In addition to the challenge of physical noise, modern libraries now face the demands of transforming into an environment that is able to meet the needs of users in the digital age. The integration of Internet of Things (IoT) technology is one of the main solutions because it allows libraries to monitor environmental conditions automatically, in real-time, and continuously, including in the context of monitoring noise levels that often change dynamically all the time [6].

IoT provides an opportunity to connect various sensors and actuators into the internet, enabling surveillance that is not only effective but also provides deeper insights through data visualization on cloud platforms like ThingSpeak. The implementation of ThingSpeak in creating a monitoring system has been proven to make it easier to process data, create graphs, and store historical data that can be accessed from any location [7].

The ESP32 device used in this study has advantages such as low energy consumption, reliable Wi-Fi and Bluetooth connections, and higher data processing speeds compared to the previous generation, namely NodeMCU ESP8266. These advantages make the ESP32 ideal for small to medium-scale IoT applications, including noise monitoring systems in libraries that require fast response and power consumption efficiency [8].

In addition, the KY-037 microphone sensor used has the ability to capture various levels of sound volume in a broad spectrum, although additional signal processing is still required to make the noise data obtained more precise and can better reflect environmental conditions. The effectiveness of these sensors in recognizing sound changes has been proven in a number of previous studies that have used them for early warning systems, engine sound detection, as well as audio-based safety systems [9].

However, threshold-focused methods have fundamental drawbacks, especially when environmental conditions change rapidly or when noise levels are at thresholds that are difficult to clearly categorize. In these circumstances, the application of fuzzy logic becomes a more responsive and more flexible option, as it can overcome uncertainty in sound signals and offer gradual grouping of noise levels, rather than simply distinguishing between noisy and non-noisy [10].

The Mamdani fuzzy model, in particular, has become one of the most commonly used approaches in decision-making systems that rely on sensor data thanks to its ability to produce easy-to-understand results. Its use in environmental monitoring has proven to be effective in improving classification accuracy, including in IoT-based systems that require quick and responsive decisions [11].

The combination of methods that rely on event thresholds and fuzzy logic in this study is expected to provide a smarter and more efficient approach to monitoring noise levels. The system will transmit information only when needed, which will reduce network usage and device power, while still maintaining proper detection capabilities when library conditions change significantly [12].

In addition, the collaboration with ThingSpeak offers significant advantages through the graphical presentation of information that can be observed by library managers to take preventive or corrective measures. This is in line with the idea of smart libraries that prioritize convenience for users and increased efficiency in the management of internal facilities. The implementation of an IoT-based voice monitoring system is in line with the needs of educational institutions to reduce disruptions in the learning process and maintain the quality of the academic environment. Previous research has shown that sound that remains stable and low has a direct effect on improved attention, concentration, and learning efficiency in library rooms and other learning places [13] , [14].

As a result, this study not only presents technical innovations through the creation of more responsive and efficient noise monitoring systems, but also makes a significant contribution to improving the quality of educational facilities through automation powered by IoT. Mamdani's blend of ESP32, KY-037, ThingSpeak, event-based approach, and fuzzy logic is expected to present a case study that is relevant and applicable to the development of smart libraries in today's digital age [15].

This research presents an innovative noise monitoring system, a departure from typical IoT setups that constantly send data and use rigid classifications. Instead of the usual methods, it uses event-triggered data sending paired with a noise classification system based on fuzzy logic. Sending data only when noise levels change considerably helps cut down on unneeded network use, which makes communication work better. Furthermore, using Mamdani fuzzy logic allows for a more flexible and step-by-step way to classify noise, which handles the ambiguity of sound signals better than methods that depend on strict limits. This pairing provides a useful upgrade to current IoT noise monitoring techniques, especially in the context of smart libraries.

2. METHOD

The fuzzy system was evaluated using sensor reading data to ensure that the resulting noise classification corresponded to actual environmental conditions. This evaluation process involved data cleaning, noise pattern analysis, and comparison between raw sensor values and fuzzy system outputs. The results were used to assess the performance and effectiveness of the proposed IoT-based noise monitoring system [16].

2.1 Research Flow

In this study, the approach used is Research and Development (R&D), because this research not only conducts measurements, but also designs, develops, tests and evaluates system prototypes before building fuzzy simulation models to measure noise levels [17], [18]. This R&D research model is considered relevant because it can generate new innovations in the form of adaptive and cloud-based noise monitoring systems, which differentiates it from previous research that only applied deterministic methods or carried out constant data transmission [19]. The following are the steps taken in this study.

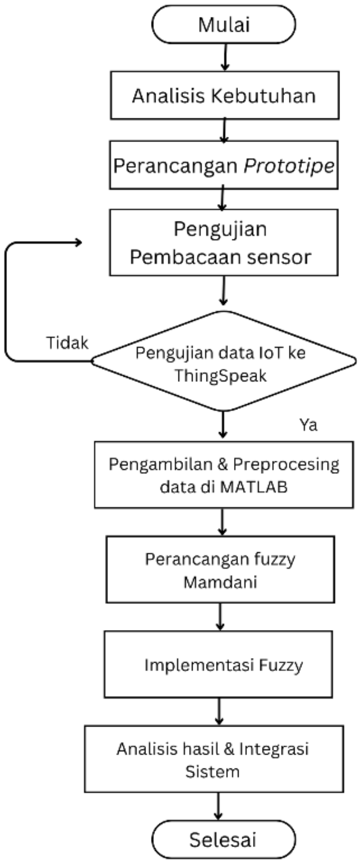


Figure 1. Research Flow

Generally, the research begins with an analysis of the need to identify pentig parameters in sound monitoring, such as the noise level applicable to the library (45-55 dB), the hardware specifications (ESP 32, KY-037, LED indicators), as well as the ThingSpeak configuration to store the data. Once the requirements analysis stage is complete, the next step is to design the electronic circuit, pin selection, and programming of the ESP32 to monitor the sound level as well as transmit the data to ThingSpeak at regular intervals [18] , [20].

The next step is to build a fuzzy model of Mamdani to overcome the uncertainty in sensor readings and provide a linguistic classification of noise conditions (quiet, moderate, and noisy). This fuzzy model was developed with three membership categories for input (dB) and three categories for output (noise level), as well as a set of fuzzy rules that are tailored to the noise level standards in the library. The fuzzy system is then evaluated using sensor reading data to ensure that the resulting classification corresponds to the actual conditions that occur [21], [16], [22] .

The fuzzy system is further evaluated using data from sensor readings to ensure that the resulting classification is in accordance with the actual conditions that occurred. This evaluation process involves data cleaning, noise pattern analysis, and comparison between raw sensor values and fuzzy system results. In the final stage of the assessment, an integration is carried out between the results of the fuzzy system measurements, and the noise level interpretation results, resulting in conclusions regarding the performance and effectiveness of the IoT-based noise monitoring system that has been developed. The final stage of this study includes the integration of measurement results, fuzzy systems, and noise level interpretation, so that conclusions are drawn about the performance and effectiveness of the IoT-based noise monitoring that has been designed [23], [24] , [25].

2.2 Hardware Planning

The hardware architecture created in this research is intended to gauge the level of sound within a library setting and show the gathered information using both LED lights and the ThingSpeak online platform. The KY-037 sound sensor is used as the system's input device, while the ESP32 DevKit V1 module operates as the primary computational element, handling tasks such as transforming analog signals to decibel measurements, managing the LED display, and sending information to the ThingSpeak service using a Wi-Fi link.

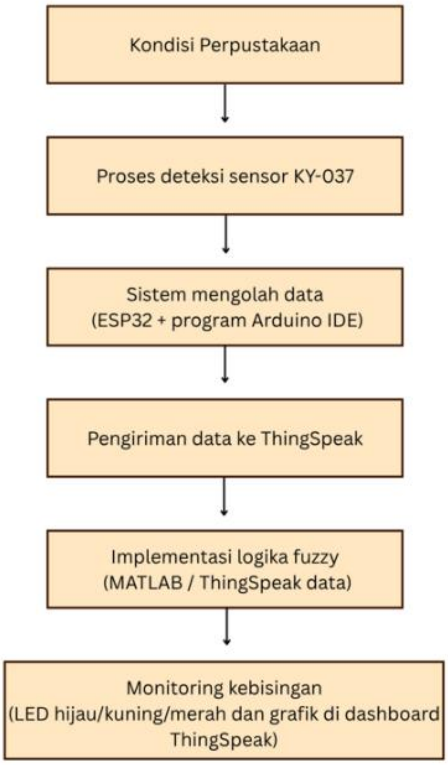


Figure 2. System workflow

The main components used can be seen in Table 1.

Table 1.Hardware components

Yes	Component Name
1	ESP32 DevKit V1 Module
2	KY-037 sound sensor
3	Green LED
4	Yellow LED
5	Red LED
6	Breadboard
7	Jumper cable
8	USB Cable

The following is the design scheme of the noise detection system that will be made in its design using sound sensors and several other components. The schematic of the plan can be seen in Figure 3.

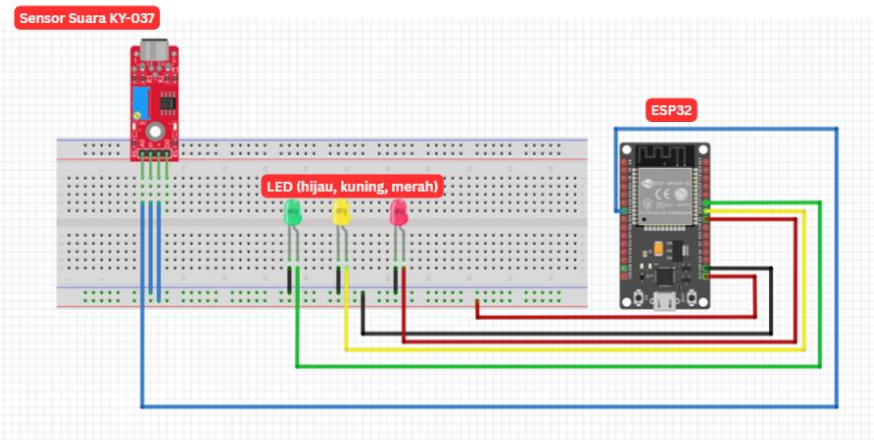


Figure 3. Hardware design scheme

The main components used can be found in Table 1. The KY-037 sensor connects to the ADC pin on the ESP32 (such as GPIO34) and the 3V3-GND power source. While green, yellow, and red LEDs are connected to ESP32 digital pins (e.g. GPIO19, GPIO18, and GPIO5) with a cathode connected to GND. All components are mounted on the breadboard using a jumper cable, and the ESP32 obtains power as well as programming connections from the laptop's USB port. The complete connection can be seen in Figure 3 as the basis for hardware implementation before being combined with IoT programming and fuzzy logic.

2.3 IoT System Planning and ThingSpeak

In this study, ESP32 served as an IoT node that connected the library's atmosphere to the ThingSpeak cloud platform. The structure of the IoT system is depicted in general in Figure 2, where the KY-037 sound sensor measures the noise level and transmits an analog signal to the ESP32. The ESP32 then processes the information and sends details about the noise to ThingSpeak over an internet-connected Wi-Fi connection.

A ThingSpeak channel is created under the name "ESP32 Noise Monitor" and is private, having a Channel ID 3102630. This channel consists of two fields, namely field 1 to store the noise value in dB (dB level), and field 2 to record the noise state (noise state) represented by discrete numbers (e.g. 0 = quiet, 1 = medium, and 2 = noise), ESP32 is set using the write API key so that only the device can send to the channel, while the analysis and data retrieval in MATLAB is carried out using the read API key.

In terms of programming, the ESP32 connects to a Wi-Fi network by leveraging an already available library, and then sends data to ThingSpeak at specific intervals (e.g. every 15 to 20 seconds). Each time the sensor read is performed, the analog voltage value of the KY-037 is converted to a dB value, then an HTTP GET URL is formed according to the format required by ThingSpeak and sent to the server. ThingSpeak automatically records data on each field and presents it in real-time graphs on web pages and mobile apps, allowing users to monitor the progress of noise levels in libraries from various locations.

The historical data stored in fields 1 and field 2 will then be used in the application of fuzzy logic. MATLAB accesses the data via the ThingSpeak CSV/API endpoint, performs pre-processing such as, removing empty values and abnormal values), and uses the noise time series as inputs for the Mamdani fuzzy model developed in this study.

2.4 Fuzzy Implementation

This research determined fuzzy membership function ranges and noise thresholds based on well-known library noise benchmarks and the results of previous investigations. Environmental noise rules state that acceptable noise levels for libraries usually fall between 45 and 55 dB; these values served as benchmarks for establishing the limits of quiet, moderate, and noisy environments. The Mamdani fuzzy inference model was chosen because it is easy to understand and works well for classifying language, allowing noise levels to be communicated using understandable language that library administrators can easily grasp. The Mamdani method offers more understandable rule depiction and smoother changes between noise categories than other fuzzy models like Takagi–Sugeno, making it a better choice for environmental monitoring uses.

In this study, Mamdani-type fuzzy logic was applied to group noise levels based on decibel values (dB) obtained from the KY-037 sensor and sent via ESP32. The input of a fuzzy system is the noise level in decibels, while the output is a noise condition category represented on a scale of 0 to 100. This category is used to set the LED indicator and the status value sent to ThingSpeak.

The fuzzy process consists of four stages, namely fuzzification, rule application (inference), rule composition, and defuzzification. The determination of variables, speaker universes, fuzzy sets, and rules used is shown in Table 2–Table 4.

Table 2. Determination of variables and speaker universe

Function	Variable Name	Speaker Universe	Information
Input	Noise level	30 – 90 dB	Measured dB

ELITE Journal
E-ISSN: XXXX-XXXX, P-ISSN: XXXX-XXXX

			values of the KY-037 sensor
Output	Noise level	0 - 100	Fuzzy score for noise conditions (low–high)

Table 3. Fuzzy Set

Function	Variable Name	Fuzzy set name	Interval (dB)
Input	Noise level	Calm	0, 0, 30, 45
		Medium	40, 55, 70
		Noisy	65, 75, 90, 90
Output	Noise level	Low	0, 0, 50
		Medium	25, 50, 75
		Noisy	50, 100, 100

In the output variable, the noise level is expressed by three linguistic categories, namely low, medium, and high in the range of values 0 to 100. The low category is used to indicate a quiet atmosphere in the library, when the decibel value is well below the noise limit. The intermediate category reflects the switching state when the noise level is close to the acceptable limit. On the other hand, the high category indicates a noise situation that is already disturbing and requires corrective action from the library manager. Based on several scenarios in fuzzy determination, the application of the implication function (rule) for noise level interference in the library results in the following number of rules.

Table 4. Fuzzy Rule

Yes	Noise Level (Input)	Noise level (Output)
1	Calm	Low
2	Keep	Keep
3	Noisy	High

Thus, all the elements in the Mamdani fuzzy system used in this study have been described in detail, starting from the selection of variables, the determination of membership pools, to the preparation of IF–THEN rules. The dB values obtained from the KY-037 sensor first go through the defuzzification process into the Quiet, Medium, or Noisy categories, and then use a rulebase to determine the noise level that can be categorized as Low, Medium, or High. The defuzzification process with the centroid method is applied to translate the inference results into concrete values in the range of 0–100, which are then used as an indicator of noise level in the library and compared with sensor data in the results and discussion sections.

3. RESULTS AND DISCUSSION

3.1 System integration

The system test was carried out after all the elements were connected, namely the KY-037 sound sensor served as input, the ESP32 module acted as an information processor and transmitter, and the ThingSpeak platform was used as an IoT server to store noise data online.

When the system is activated, the KY-037 sensor detects the noise level in the library environment and converts it into an analog signal, which is then converted into dB values by ESP32. That dB value is routinely sent to the ThingSpeak channel in column 1 for the dB level, while column 2 is set up to store the noise state generated by fuzzy logic. The graph displayed in ThingSpeak shows that the relationship between ESP32 and the server is functioning well, as seen from the regular appearance of data points on the time axis.

3.2 Noise data acquisition results from ThingSpeak

The noise data stored in ThingSpeak is then downloaded in CSV format and analyzed using MATLAB. Based on the data snippet in the hasil_fuzzy_noise file. csv, the detected noise value ranges around 45 dB with slight variation in each time sample. This value reflects the condition of the library that is quite quiet and still within the permissible noise limit.

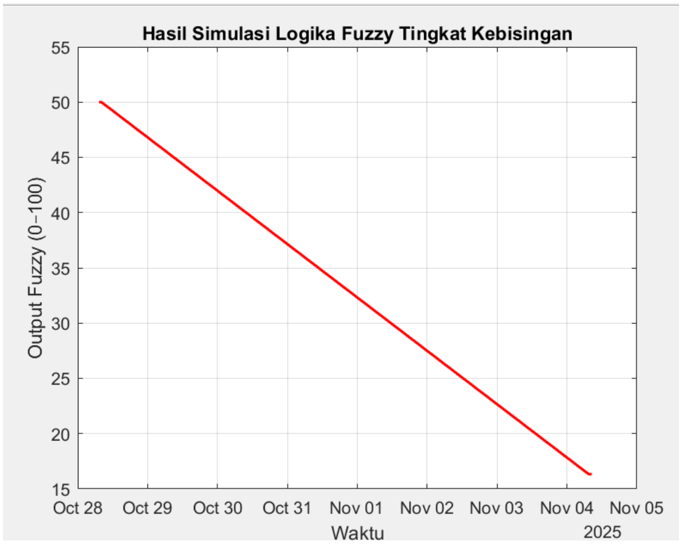


Figure 4. Noise level fuzzy logic simulation results

The graph "Noise Data from ThingSpeak" shows that during the observation period, most of the data points were in a fairly stable range, with occasional spikes in value when noise sources were added around the sensor. This pattern indicates that the KY-037 sensor can respond to changes in sound intensity and that the process of sending data to ThingSpeak does not experience significant data loss.

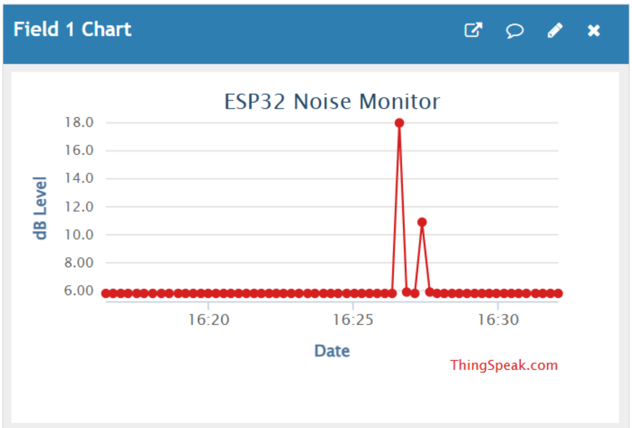


Figure 5. Field chart 1

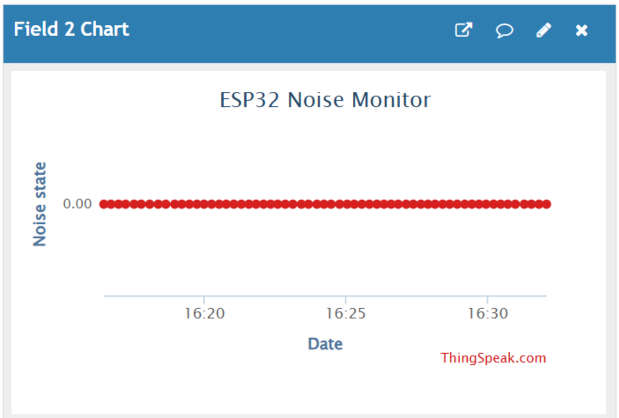


Figure 6. Field chart 2

In the graph shown in ThingSpeak (Chart Field 1), there are rows of red dots representing dB values over time, with some value peaks indicating increased noise. On the other hand, the Field 2 graph for noise status remains the same throughout the initial test because the fuzzy classification has not been updated in ThingSpeak, so at this stage, the field is more of an indication of channel availability for future fuzzy integration.

3.3 Noise level fuzzy logic simulation results

The dB data obtained from measurements using ThingSpeak is then used as input for the Mamdani fuzzy system in MATLAB. Each dB data will be fed into the FIS (Fuzzy Inference System) that has been developed, which has three fuzzy groups for the input variables, namely Quiet, Medium, and Noisy, and three fuzzy groups for the output variables, namely Low, Medium, and High in the range of 0 to 100. The inference process generates a "fuzzyOutput" time series that describes the noise level in the library based on the fuzzy scale.

The graph "Noise Level Fuzzy Logic Simulation Results" shows that the fuzzy output is in the average number when the noise level is close to a predetermined limit. When the dB value remains close to the limit, the fuzzy output also tends to be stable in the middle number (e.g. around 50), indicating the presence of noise conditions in the medium category. If in subsequent tests the dB value is boosted higher than the limit considered Noisy, the fuzzyOutput value is expected to rise close to 100, so it can be easily understood as a situation that requires warning.

3.4 Noise level fuzzy logic simulation results

The graph entitled "Comparison of dB Sensor vs Fuzzy Output" in Figure 7 presents a comparison between the dB data obtained from the sensor reading results and the fuzzy output results. In the graph, the axis located on the left side represents a dB curve of the sensor, while the axis located on the right side represents a fuzzyOutput curve in a scale range between 0 and 100. Broadly speaking, the configuration of these two curves shows an identical trend: when the dB value decreases gradually, the fuzzy output value also decreases, and when the dB increases slightly, the fuzzy curve responds in harmony with the unidirectional increase. This indicates that a pre-designed Mamdani fuzzy model is capable of converting various changes in sound intensity into multiple noise levels continuously.

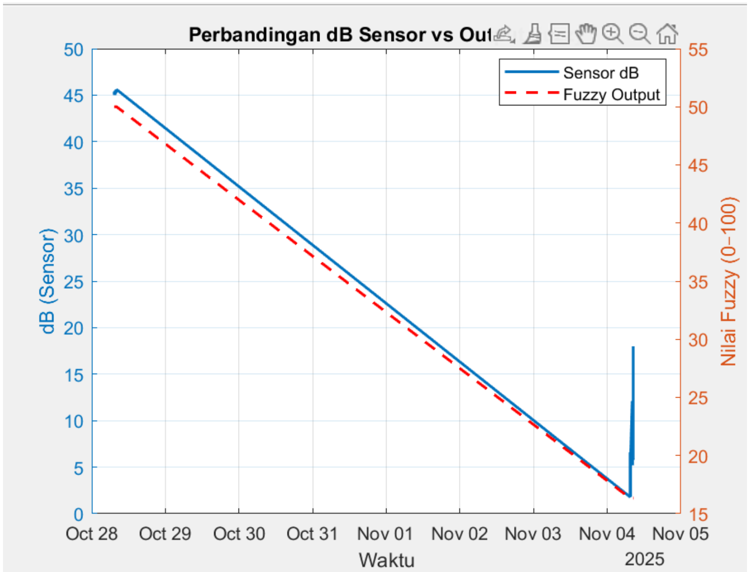


Figure 7 Comparison of dB Sensor vs Fuzzy Output

There is a small difference between dB and fuzzyOutput values at certain points due to the form of the membership function and the defuzzification process which is specifically designed to provide a smooth transition between the various linguistic categories. In other words, even if the dB value changes significantly enough, the fuzzy output will even out the changes that occur so that the classification of noise conditions does not change very extremely just because of a momentary fluctuation in a sensor signal. This characteristic is so important to prevent a system from giving an excessive warning in the event of a brief noise that is not too annoying.

3.5 Discussion

It's important to recognize that the assessment carried out in this research mostly depends on emulated systems, alongside confined real-world application using just one sensor device during a comparatively brief monitoring timeframe. As a result, we didn't quantitatively confirm the sensor's precision by directly contrasting it with a correctly calibrated sound measurement tool. This constraint could have an impact on how widely the identified noise intensities can be applied; nevertheless, the main goal of this study is to illustrate that combining the system, sending information when events happen, and classifying sound using fuzzy logic inside an Internet of Things arrangement can be achieved. Subsequent studies will work on resolving this restriction by adding the use of multiple device applications and quantitative fine-tuning to make measuring more precise and dependable.

The experiments conducted led to the conclusion that the noise monitoring system, which utilizes the ESP32 and KY-037 sensor, successfully transmitted noise data to ThingSpeak in real-time and created data patterns that reflected the environmental conditions of the library. The implementation of Mamdani fuzzy logic on this data effectively categorized noise levels into low, medium, and high, aligning with the decibel values that were recorded. This integrated approach bolsters the smart library concept, allowing managers to use easily understandable noise information—both in decibels and fuzzy levels—to decide when actions like addressing visitors or rearranging the layout are needed. In general, these findings suggest that the proposed system has the potential for additional development, such as sending the fuzzy output back to ThingSpeak as a noise state, and incorporating actuators like buzzers or light indicators to provide automated alerts within the library setting.

The outcomes underscore that merging event-driven data transfer with fuzzy logic categorization creates a significantly more versatile and insightful way to keep tabs on noise levels when contrasted with standard systems relying on simple thresholds.

4. CONCLUSIONS

This research developed and tested a library noise surveillance system that uses IoT technology, which includes an ESP32 microcontroller, a KY-037 audio sensor, the ThingSpeak service, and a Mamdani fuzzy inference method. The suggested system demonstrated effective tracking of sound intensity, consistent transmission of information to the internet, and categorization of sound environments into relevant noise groups using fuzzy logic.

The experimental outcomes demonstrate the proficiency of the ESP32 and KY-037 component in capturing acoustic strength across the library vicinity and conveying decibel figures to ThingSpeak consistently, maintaining uninterrupted connectivity. The information preserved on ThingSpeak is accessible for retrieval in CSV format and subsequent manipulation within MATLAB, wherein the Mamdani fuzzy framework converts the input decibels into fuzzy output tiers, mirroring the patterns observed in the sensor data. This suggests the coherence of the crafted fuzzy regulations and membership roles, effectively portraying the shifts among silent, average, and loud settings in a more progressive fashion compared to utilizing a solitary static benchmark. The graphical representation of both decibel and fuzzy outputs across time reinforces the notion of a technologically advanced library, given the capability to oversee noise levels from afar and interpret them with greater instinctiveness.

Even though the system's operation has been consistent with its original blueprint, its current application is confined to only one detection location and a comparatively brief duration of monitoring. Further investigations could broaden this project by positioning several sensor units in various sections of the library, transmitting the fuzzy categorization back to ThingSpeak as a representation of noise conditions, and incorporating devices like warning signals or sounders to provide automated notifications whenever the sound intensity surpasses the suggested threshold. Furthermore, more thorough fine-tuning and assessment across diverse authentic library environments are essential for gathering measurable performance data like precision and reaction speed.

Currently, the fuzzy logic calculations are done separately using MATLAB, using information taken from the ThingSpeak system. Because of this, the results of the fuzzy logic are not yet used in the live IoT system. This method was chosen to make building the system easier and to concentrate on checking how the fuzzy classification works. Further studies will try to include the fuzzy logic directly on the ESP32 or make it so the fuzzy results on ThingSpeak change automatically, which would allow for seeing noise levels in real time and automatic reactions inside the IoT system.

5. AKNOWLEDGMENT

The authors would like to express their sincere appreciation to the Faculty of Engineering, State University of Makassar, for providing research facilities and technical support that contributed to the design, simulation, and analysis stages of this study. The authors also thank colleagues and staff who offered constructive feedback and administrative assistance during the research process.

6. REFERENCES

[1] H. K. Riyadi *et al.*, "PROTOTYPE OF SOUND NOISE DETECTION DEVICE USING NODEMCU ESP8266 IN THE STMIK WIDYA CIPTA DHARMA LIBRARY ROOM," no. 2, p. 3, 2024, doi: 10.46984/sebatik.v28i2.0000.

[2] Mirjaz Najib Naufal, Syathir Ansyarullah, Az-Zahra Shofiya, and Martana Salmon Priaj, "aditya,+PAPER+2-DESA+September+2024," Dec. 2024.

[3] B. Nodemcu *et al.*, "Implementation of GYMAX4466 Sensors in Noise Monitoring Systems Using the Internet of Things (IOT)," *Jurnal CyberTech Journal*, vol. 4, no. 8, 2021, [Online]. Available: https://ojs.trigunadharma.ac.id/

[4] Al Matsany Ahmad Najih and Martiani Ahmad, "(+648-661+)+DESIGN+BUILD+TOOLS+DETECTION+NOISE+VISITORS+LIBRARIES+USE+MICROCONTROLLER+INTERNET+OF+THINGS+(IOT)," Jun. 2025.

[5] Sitiawan, Bambang Apriyanto, and Gunawan T Hadiyanto, "PROTOTYPE OF NOISE DETECTION OF VISITORS TO THE UNIVERSITY OF BATAM LIBRARY," *Electrical Zone: Electrical Engineering Study Program (S1) University of Batam*, vol. 14, no. 3, Dec. 2024, doi: 10.37776/ze.v14i3.1619.

[6] S. Choudhary, A. D. R, H. Likith Sai Varma, and L. S. Kumar, "SMART SURVEILLANCE MONITORING SYSTEM USING MACHINE LEARNING AND RASPBERRY PI," *www.irjmets.com @International Research Journal of Modernization in Engineering*, vol. 585, [Online]. Available: www.irjmets.com

[7] P. Nababan and T. A. Andromeda and Joshua Alvin, "DESIGN OF AN INTERNET OF THINGS (IOT)-BASED HYDROPONIC NUTRIENT FILM TECHNIQUE (NFT) MONITORING SYSTEM USING THE WEB SERVER THINGSPEAK." [Online]. Available: https://ejournal3.undip.ac.id/index.php/transient

[8] F. Serepas, I. Papias, K. Christakis, N. Dimitropoulos, and V. Marinakis, "Lightweight Embedded IoT Gateway for Smart Homes Based on an ESP32 Microcontroller," *Computers*, vol. 14, no. 9, Sep. 2025, doi: 10.3390/computers14090391.

[9] N. A. Zen, S. Indriyanto, I. Permatasari, J. Sasmita, and L. Yuliantini, "Preliminary Study of Using KY-038 Sensor Based on Arduino UNO and LabView to Determine the Pulse Rate," *Journal of Physics and Its Applications*, vol. 16, no. 2, p. 87, Jun. 2020, doi: 10.12962/j24604682.v16i2.6092.

[10] Tm. Devi, "Environmental Noise Classification and Cancellation using Fuzzy Classifier and Fuzzy Adaptive Filters," 2012. [Online]. Available: www.IJCSI.org

[11] M. Yudha and S. Susilawati, "Analysis of Air Pollution Conditions Based on Time Changes Using IoT and Fuzzy Logic: Solutions to Prevent the Impact of Pollution on Health," *INCODING: Journal of Informatics and Computer Science Engineering*, vol. 5, no. 2, pp. 218–238, May 2025, doi: 10.34007/incoding.v5i2.862.

[12] V. Tiwari, A. Keskar, and N. C. Shivaprakash, "A reconfigurable IoT architecture with energy efficient event-based data traffic reduction scheme," *International Journal of Online Engineering*, vol. 13, no. 2, pp. 34–52, 2017, doi: 10.3991/ijoe.v13i02.6460.

[13] A. Shaikh, S. Gurav, A. Jagdale, P. Kamble, S. Chavan, and R. Scholar, "IoT-Based Connected Environmental Monitoring System Using NodeMCU." [Online]. Available: www.ijfmr.com

[14] P. Goldschagg, T. Bekker, and K. Cockcroft, "Perceived effects of background noise on the learning experiences of English first- and second-language female learners," *S Afr J Educ*, vol. 43, no. 1, 2023, doi: 10.15700/saje.v43n1a2153.

[15] D. Alfiyyah Ansar and I. Taufik, "The Design of a Noise Detection Automation."

[16] D. Dwi Sutia *et al.*, "Quiet Environment Optimization with Noise Monitoring Systems Using Fuzzy Logic," vol. 18, no. 1, pp. 1979–276, 2024, doi: 10.30998/factorexacta.v18i1.24869.

[17] R. Susanto, "JISEC_V6N2_2025_05," Oct. 2025.

[18] K. Valendra, S. Si, and C. Setiawan, "DEVELOPMENT OF AUTOMATIC NOISE DETECTION SYSTEM IN LIBRARIES USING GOOGLE ASSISTANT AND ESP32 BASED ON VOICE RECOGNITION."

[19] T. Ervina Sari and T. Siswandi Syahputra, "MAPPING AND MONITORING OF NOISE LEVELS BASED ON IOT (INTERNET OF THINGS) AT THE SUMATRA INSTITUTE OF TECHNOLOGY," *Original Article Journal of Science and Applicative Technology*, vol. xx, pp. xx–xx, doi: 10.35472/x0xx0000.

[20] A. Kurniawan, A. Ristiono, and S. Sulistiadi, "Microclimate Monitoring in Greenhouse in Real Time Using Thingspeak-Based Internet of Things (IoT)," *Journal of Agricultural Engineering*, vol. 10, no. 4, p. 468, Dec. 2021, doi: 10.23960/jtep-l.v10i4.468-480.

[21] A. Saleh, "IMPLEMENTATION OF THE MAMDANI FUZZY METHOD IN PREDICTING TRAFFIC NOISE LEVELS," pp. 6–8, 2015.

[22] M. D. Pop, D. Pescaru, and M. V. Micea, "Mamdani vs. Takagi-Sugeno Fuzzy Inference Systems in the Calibration of Continuous-Time Car-Following Models," *Sensors (Basel)*, vol. 23, no. 21, Oct. 2023, doi: 10.3390/s23218791.

[23] S. Purnama Jati and O. Melfazen, "EXHAUST NOISE MONITORING SYSTEM IN FUZZY MAMDANI-BASED MOTOR VEHICLES."

[24] N. H. Cahyadi, V. Nastiti, A. E. G. Audryadmaja, and D. O. Radianto, "Design of a Work Environment Quality Determination System Based on Multi Input Parameters Using the Mamdani Fuzzy Inference System (FIS) Method," *Journal of Health*, vol. 10, no. 02, pp. 119–128, Sep. 2025, doi: 10.30590/joh.v10n2.646.

[25] D. Hercog, T. Lerher, M. Truntič, and O. Težak, "Design and Implementation of ESP32-Based IoT Devices," *Sensors*, vol. 23, no. 15, Aug. 2023, doi: 10.3390/s23156739.