

Design and Implementation of an IoT-Based Household Electricity Monitoring and Control System with Auto Cut-Off and Data Logger Features

Muhammad Fadhl Ramadhan^{1*}, Ahmad Rofiq Hakim², Fara Triadi³

^{1,2,3}Computer Engineering Technology Study Program, Departemen of Information Technology, State Polytechnic of Samarinda
mfadhl2610@gmail.com^{1*}, rofiq_93@yahoo.com², fara_triadi@polnes.ac.id³

Abstract

The growing demand for household electricity frequently leads to energy waste and overload risks. This study presents the design and implementation of an IoT-based Smart Energy system employing the ESP32 DevKit V1 microcontroller as the central controller. The system integrates an ACS758 current sensor, a ZMPT101B voltage sensor, an RTC DS3231 module, and a MicroSD card for timestamped historical data logging. Overload protection is achieved through a relay and a Miniature Circuit Breaker (MCB) that automatically disconnects the power supply when the load exceeds the configured threshold. The research adopts a Research and Development (R&D) methodology encompassing problem identification, system design, implementation, testing, and evaluation. Test results demonstrate that the system reliably monitors voltage, current, power, and energy consumption in real-time, records data to the SD card with accurate timestamps, and executes the auto cut-off function effectively. Comparative testing against a calibrated multimeter produce an average measurement error below 5.5%, confirming the system's suitability for household-scale energy monitoring applications.

Keywords: Auto cut-off; Data logger; ESP32; Internet of Things; Smart energy monitoring

1. Introduction

Rapid technological advancement has driven a significant increase in household electricity consumption. The proliferation of electronic devices at the individual level often results in uncontrolled energy usage, which not only raises electricity costs but also poses risks of overload and equipment damage [1]. At the same time, the declining reserves of fossil fuels—still the primary source of electricity in Indonesia—make energy efficiency an increasingly urgent matter. Unfortunately, public awareness regarding energy conservation in domestic settings remains relatively low.

The Internet of Things (IoT) offers an innovative approach to address these challenges. By integrating microcontrollers with voltage and current sensors, it becomes possible to build a system that monitors electricity consumption in real time and provides automatic protection against overload conditions [14]. Prior studies have demonstrated the viability of this approach from various perspectives. Himer et al. [2] reported that an IoT-based monitoring system using NodeMCU and PZEM-004T sensors could record voltage, current, and power with measurement errors below 0.6%. Garcés et al. [3] developed a smart meter using non-invasive sensors that achieved a Mean Absolute Percentage Error (MAPE) of less than 7% for all electrical parameters. In a similar vein, Hadi et al. [7] designed an IoT-based monitoring system that enabled remote observation of household power consumption through a web interface. Condon et al. [8] further advanced this direction by integrating cloud computing with IoT for comprehensive home energy management, while Dinmohammadi et al. [9] demonstrated that smart monitoring systems could significantly improve energy efficiency in building environments. Khan et al. [10] addressed the cost barrier by developing a low-cost monitoring system with universal compatibility and real-time visualization capabilities, and Hasan et al. [11] showed that IoT-based systems using usage data could effectively monitor and control electricity in real time.

The control dimension has received growing attention as well. Pratama and Kiswanto [4] showed that ESP32 combined with a solid-state relay could automatically disconnect the electrical flow when the load exceeded the MCB capacity. Yusuf, Rosyidi, and Saptono [5] demonstrated that integrating ESP8266 with an online platform for household load monitoring improved monitoring accuracy to 100% and provided daily consumption pattern insights. El-Khozondar et al. [6] developed an ESP32-based energy monitoring system with real-time notifications and historical data storage that exhibited high measurement accuracy. Hendrawan and Agustini [13] simulated ESP32-based control and monitoring of household appliances, confirming the microcontroller's suitability for domestic energy applications. Priyono and Sumardi [23] further validated the ESP32 platform for combined monitoring and control of electrical loads through IoT integration. Beyond

the household scale, Adyana Putra et al. [12] applied a comparable monitoring approach to electric vehicles, measuring voltage, current, and speed parameters with similar sensor architectures.

Despite these advances, many existing systems address either monitoring or control in isolation, and few combine both with reliable local data storage. Building upon these findings, the present study aims to develop an integrated Smart Energy system that combines real-time monitoring, automatic overload protection (auto cut-off), and historical data logging using a MicroSD card and RTC module. The system is designed to be accessible through a web dashboard, enabling remote monitoring via smartphone. This integrated approach addresses the gap identified in several prior studies that typically focused on monitoring or control separately, without combining both with local data storage capabilities.

2. Methods

2.1. Research and Design

This study adopts a Research and Development (R&D) methodology adapted from a simplified Borg and Gall model to systematically develop and validate a functional Smart Energy prototype. The R&D framework was selected because the primary objective is to design, build, and evaluate a new technology product rather than merely test hypotheses [16, 17, 18]. Haq et al. [16] and Ekowati et al. [17] demonstrated the effectiveness of this approach in developing technology-based educational products, while Abdurrahman [18] provided a comprehensive overview of R&D methodology applications in educational research contexts. The present study follows the same developmental logic: iterative design, prototype construction, and empirical evaluation.

The research was conducted at the Department of Information Technology, State Polytechnic of Samarinda, from June to October 2025. The development process began with problem identification through observation and literature review, followed by data collection and comparative analysis of existing IoT-based monitoring systems. Based on these findings, the system architecture was designed—including hardware schematics, software flowcharts, and database structure—before proceeding to hardware assembly and software programming. The completed prototype then underwent functional and accuracy testing, followed by evaluation and refinement of identified issues.

2.2. System Architecture

The system architecture centers on the ESP32 DevKit V1 microcontroller, selected for its dual-core processor, built-in Wi-Fi and Bluetooth connectivity, and 12-bit Analog-to-Digital Converter (ADC) essential for precise sensor data acquisition. Hilmansyah et al. [24] previously demonstrated the ESP32's reliability in wireless monitoring applications, reinforcing its suitability as the core controller for this system. The overall hardware architecture is illustrated in the following block diagram.

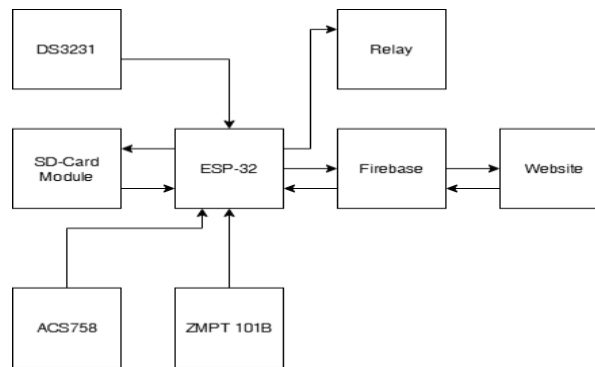


Fig. 1: Blok Diagram Architecture

The current sensor selected for this system is the ACS758, which Antara et al. [25] had previously employed for household electricity capacity control, confirming its accuracy and suitability for domestic load measurement. For voltage sensing, the ZMPT101B module was used, while the RTC DS3231 provides timekeeping with an accuracy of ± 2 ppm over an I2C interface. Iqbar and Riyanti [19] had validated the use of RTC modules with Arduino-based systems for time-dependent automation tasks, supporting the selection of DS3231 in this design. The MicroSD card module communicates via the Serial Peripheral Interface (SPI), a protocol that Baskoro et al. [15] demonstrated to be effective for output pin expansion and peripheral communication in Arduino-based systems. The hardware components and their specifications are summarized in Table 1.

Table 1: Hardware and Component

No	Component	Specification	Function
1	ESP32 Devkit V1	Dual-core 240 MHz, 520 KB SRAM, Wi-Fi, BLE, 30 GPIO	Main controller for data processing and IoT communication
2	ACS758 100A	Current range $\pm 50\text{A}/100\text{A}$, analog output, 3–5V supply	AC/DC current measurement
3	ZMPT101B	Input 0–250 VAC, output 0–5V analog	AC voltage measurement
4	RTC DS3231	Accuracy ± 2 ppm, I2C interface, CR2032 backup	Real-time timestamp for data logging

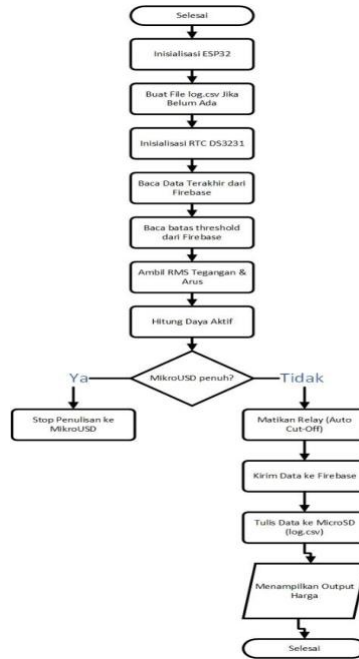


Fig. 4: Flowchart

2.4. Testing Protocol

The firmware was developed using the Arduino IDE and implements several key functions: ADC-based sensor acquisition with DC offset removal for accurate RMS computation, energy accumulation using numerical integration ($E_{wh} = P \times \Delta t$), Firebase Realtime Database communication for remote monitoring, periodic CSV data logging to the MicroSD card, and command handling for remote relay control and threshold configuration.

The data storage structure in Firebase Realtime Database was designed with main fields including V_{rms} (float), I_{rms} (float), timestamp (text), power in Watts (float), energy in Wh (float), and power factor (float). Data is published to Firebase at one-second intervals for real-time dashboard display, and recorded to the MicroSD card every 10 minutes with timestamps from the RTC module.

$$Error(\%) = \left| \frac{Nilai\ Sensor - Nilai\ Referensi}{Nilai\ Referensi} \right| \times 100\%$$

Calibration was performed by first obtaining a reference constant from the ratio of the multimeter reading to the raw sensor value. This constant was then applied as a multiplier in the program code to bring the sensor readings closer to the actual values.

3. Result and Discussion

3.1. Hardware Implementation

Hardware implementation involved realizing all schematic designs into a functional physical product by soldering components onto a Printed Circuit Board (PCB), where the ESP32 serves as the central controller connecting all sensors and modules.



Fig. 5: Installation of All Components

After the main circuit board was completed, the components were integrated with the MCB and organized into an enclosure for safety.

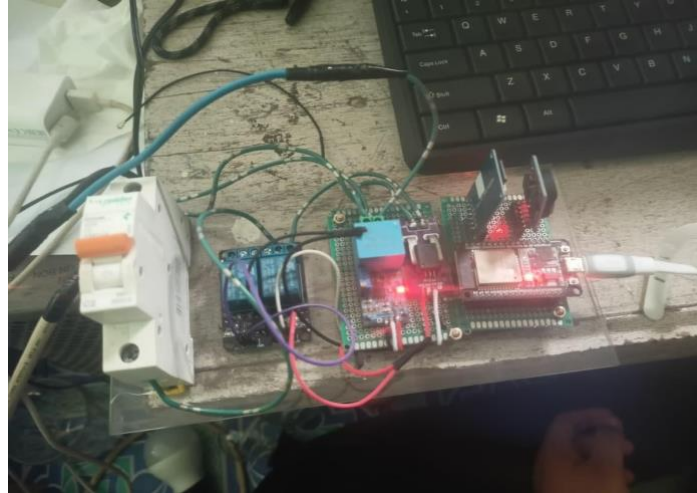


Fig. 6: Final Hardware Realization

3.2. Software Implementation

A web-based user interface was developed as the central data visualization hub, informatively displaying all electrical parameters in real-time. The dashboard comprises a device control panel for manually turning the relay on or off and synchronizing the RTC time, real-time data displaying voltage, current, power, and total energy, a live power chart alongside a forecast for the next 24 hours using single exponential smoothing [26], a raw data table showing data received from the device at specified time intervals, and a list of historical data files on the MicroSD card that users can access or download.

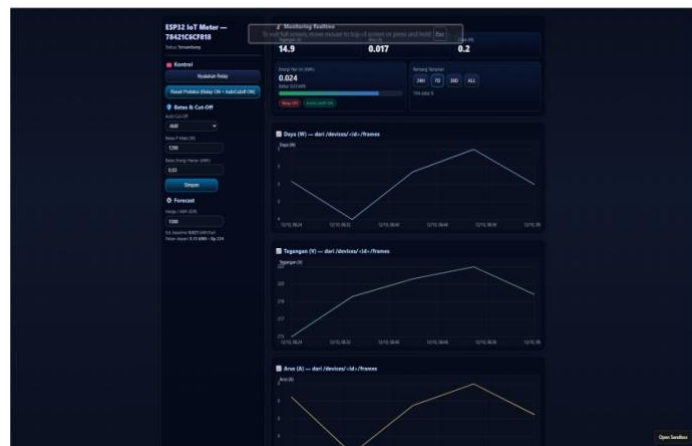


Fig. 7: Web Dashboard

3.3. Function Testing Results

Functional testing verified that all core system features operate as designed. Each function was tested using specific scenarios to compare expected results with actual outcomes. Table 2 summarizes the test scenarios and results.

Table 2: Functional Testing result

No	Function Tested	Test Scenario	Expected Result	Actual Result	Status
1	Electrical parameter measurement	System connected to an active electrical load (lamp)	Device accurately measures voltage and current	System successfully reads voltage (V) and current (A) accurately	Pass
2	Data Logger	System left operating over an extended period	Electricity consumption data recorded periodically to a file on the SD card	Consumption data successfully recorded and saved to a .csv file at the specified time intervals	Pass
3	Web Dashboard display	Data from ESP32 sent to Firebase and accessed via browser	Device displays electricity usage data in real-time on the web dashboard	Web dashboard successfully displays all electrical parameters in real-time and device status matches actual conditions	Pass

4	Auto cut-off feature	System given a load exceeding the specified capacity	System automatically disconnects the electrical flow for safety	Relay successfully disconnects the electrical flow automatically when the load is detected to exceed the set threshold	Pass
---	----------------------	--	---	--	------

All four tested functions passed their respective scenarios. The system demonstrated reliable end-to-end data communication from sensor acquisition through cloud-based visualization.

3.4. Sensor accuracy results

Following calibration with a reference constant of 360 (derived from the ratio of the multimeter reading of 205.2 V to the raw sensor value of 0.57), sensor accuracy was evaluated against a calibrated multimeter using a 60W incandescent lamp as the test load. The calibration constant was applied as a multiplier to the sensor value to approach the actual reading ($360 \times 0.57 = 205.2$ V). The test results are presented in Table 3.

Table 3: Sensor Accuracy after Calibration

Test load	Parameter	Reference value (multimeter)	System reading	Difference	Error%
60W incandescent lamp	Voltage	205.2 V	205.2 V	0 V	0%
	Current	0.26 A	0.25 A	0.01 A	3.84%
	Power	58.5 W	56.1 W	2.4 W	4.10%

The voltage measurement demonstrated excellent performance after the calibration process, achieving perfect accuracy when compared with the reference measurement instrument. This indicates that the calibration procedure was effective in minimizing measurement deviations and improving the reliability of the voltage sensor output. Meanwhile, the current and power measurements still showed small deviations, with error values of 3.84% and 4.10%, respectively. However, these error percentages remain below the acceptable threshold of 5.5% for household-scale energy monitoring applications. Therefore, the measurement results can be considered sufficiently accurate and reliable for monitoring electrical parameters such as voltage, current, and power in residential environments. These findings also indicate that the developed system is capable of providing stable and consistent measurement performance. Furthermore, the results are in line with the study conducted by Adyana Putra et al. [12], who reported comparable levels of accuracy in voltage and current monitoring for electric vehicle systems using similar sensor configurations.

3.5. Data Logger Performance

The data logger feature was validated by operating the system continuously over an extended period to ensure that the recording process could function reliably without interruption. During the testing process, the system successfully created and maintained a log.csv file on the MicroSD card as the main storage medium for measurement data. The recorded data included several important electrical and system parameters, such as voltage, current, power, cumulative energy consumption, relay status, and cut-off status. These parameters were automatically stored at regular 10-minute intervals, allowing the system to provide a consistent history of electrical usage and protection activity over time.

The successful logging process indicates that the MicroSD-based data storage system was able to operate properly and record data in a structured format. By storing the data in CSV format, the recorded information can be easily accessed, reviewed, and analyzed using spreadsheet software or other data processing tools. This feature is particularly useful for monitoring long-term energy consumption patterns, identifying abnormal electrical conditions, and evaluating the performance of the relay and cut-off mechanism. A sample of the recorded data obtained during the testing process is presented in Table 4.

Table 4 Sample data logger

Date	Time	Voltage (V)	Current (A)	Power (W)	Energy (kWh)	Relay	Cut-off
12/10/2025	08:24:09	214.98	0.022	4.66	0.020	1 (ON)	0
12/10/2025	08:34:11	219.14	0.020	4.37	0.021	1 (ON)	0
12/10/2025	08:44:14	220.97	0.021	4.74	0.022	1 (ON)	0
12/10/2025	08:54:17	222.21	0.022	4.91	0.023	1 (ON)	0
12/10/2025	09:04:22	219.36	0.021	4.64	0.023	1 (ON)	0
12/10/2025	22:19:59	218.11	0.022	4.85	0.025	1 (ON)	0

12/10/2025	22:29:59	216.30	0.015	3.26	0.025	1 (ON)	0
------------	----------	--------	-------	------	-------	--------	---

The consistent 10-minute interval between records and the presence of accurate timestamps confirm the reliability of both the data logging mechanism and the RTC module integration. A relay status of 1 (ON) indicates normal operation, while a cut-off status of 0 denotes that no overload event occurred during the recorded period.

3.6. Equations

The auto cut-off mechanism was tested by configuring a power threshold and applying a load that exceeded it. The relay responded promptly, disconnecting the electrical flow upon detecting the overload condition. The system correctly updated the cut-off status flag in both the Firebase database and the local SD card log. Hidayat et al. [22] implemented a comparable protection mechanism using Node-Red for current-based load disconnection, and the present system achieves similar protective functionality through a fully embedded approach that does not rely on external middleware. This confirms that the protection logic functions as designed, providing an additional safety layer beyond the conventional MCB.

3.7. Discussion

The testing results demonstrate that the Smart Energy prototype successfully achieves all stated research objectives. The real-time monitoring capability, realized through the integration of ACS758 and ZMPT101B sensors with the ESP32 and Firebase web dashboard, provides users with continuous visibility over their electricity consumption. The data logger, backed by the MicroSD card and RTC DS3231, enables reliable historical data recording for long-term consumption pattern analysis. The auto cut-off feature provides effective overload protection through relay-based automatic disconnection.

Compared to prior studies, this system offers a more comprehensive solution. Himer et al. [2] focused on monitoring without local data storage, while Yusuf et al. [5] emphasized visualization without load control. Garcés et al. [3] achieved high measurement accuracy but did not support automatic load control. The system developed in this study integrates monitoring, control, and data logging within a single platform. The addition of RTC-based timestamping further distinguishes this system from several predecessors that lacked precise temporal data recording.

Suparmono et al. [28] demonstrated a relay-based control approach using PLC for elevator systems, and the auto cut-off mechanism in the present study applies a comparable relay-disconnect principle tailored to household electrical loads. The forecasting feature incorporated into the dashboard draws on the single exponential smoothing method validated by Deni et al. [26] for time-series prediction, adapted here to project 24-hour power consumption trends.

However, several limitations should be noted. First, testing was limited to light household loads such as lamps and fans; heavy inductive loads such as water pumps or air conditioning compressors were not evaluated. Second, the system does not incorporate advanced data encryption or user authentication mechanisms. Lin et al. [27] highlighted vulnerabilities in integrated development environments that underscore the broader importance of software security, and future iterations of this system should address such concerns. Watmah et al. [29] demonstrated the use of fingerprint-based authentication for security systems, suggesting a potential direction for physical access control in smart energy devices. For data-level security, Zakaria et al. [30] implemented the ElGamal encryption algorithm for securing data in wireless sensor networks, an approach that could be adapted to protect the Firebase communication channel in this system. Third, the monitoring scope covers voltage, current, power, and energy, without extending to power quality parameters such as harmonics or power factor.

4. Conclusion

Based on the design, implementation, and testing of the IoT-based Smart Energy system for household electricity monitoring and control, the following conclusions are drawn:

The system was successfully built by integrating the ESP32 microcontroller with the ACS758 current sensor, ZMPT101B voltage sensor, RTC DS3231 module, MicroSD card, and relay. All hardware components operate synergistically with the developed firmware, producing a functional and stable prototype. The system performs real-time acquisition of electrical parameters—voltage, current, power, and energy—and displays them through a Firebase-based web dashboard accessible remotely via smartphone with low latency. The data logger feature reliably records energy consumption history at 10-minute intervals to the MicroSD card, with each entry bearing an accurate timestamp from the RTC DS3231 module, enabling valid long-term consumption analysis. The auto cut-off mechanism functions effectively, with the relay automatically disconnecting the electrical flow when the detected load exceeds the configured power or energy threshold, thereby providing an additional protection layer against overload risks. Comparative testing against a calibrated multimeter confirms that the system achieves measurement accuracy sufficient for household applications, with an average error below 5.5% after calibration.

For future development, it is recommended to enhance network security through data encryption and user authentication, develop a dedicated mobile application with push notification support, optimize the hardware design using SMD components and custom PCBs for a more compact form factor, and explore artificial intelligence integration for intelligent energy consumption prediction and savings recommendations.

References

- [1] N. C. Sembiring et al., "Evaluasi penggunaan energi listrik mahasiswa Unimed terhadap efisiensi dan kesadaran energi di lingkungan sehari-hari," *Innovative: Journal of Social Science Research*, vol. 4, no. 5, pp. 8570–8581, 2024, doi: 10.31004/innovative.v4i5.15772.
- [2] S. E. Himer et al., "Energy consumption monitoring system based on IoT for residential rooftops," *Computation*, vol. 11, no. 4, p. 78, 2023, doi: 10.3390/computation11040078.
- [3] H. O. Garcés et al., "Development of an IoT-enabled smart electricity meter for real-time energy monitoring and efficiency," *Electronics*, vol. 14, no. 6, p. 1173, 2025, doi: 10.3390/electronics14061173.
- [4] E. W. Pratama and A. Kiswantono, "Electrical analysis using ESP-32 module in realtime," *JEECS (Journal of Electrical Engineering and Computer Sciences)*, vol. 7, no. 2, pp. 1273–1284, 2023, doi: 10.54732/jeeecs.v7i2.21.
- [5] M. B. Yusuf, L. Rosyidi, and H. Saptono, "Implementasi sistem IoT untuk monitoring konsumsi energi listrik di rumah pintar," *DBESTI: Journal of Digital Business and Technology Innovation*, vol. 2, no. 1, pp. 28–34, 2025, doi: 10.54914/dbesti.v2i1.1354.
- [6] H. J. El-Khozondar et al., "A smart energy monitoring system using ESP32 microcontroller," *e-Prime - Advances in Electrical Engineering, Electronics and Energy*, vol. 9, p. 100666, 2024, doi: 10.1016/j.prime.2024.100666.
- [7] S. Hadi, A. S. Anas, and L. G. R. Putra, "Rancang bangun sistem monitoring penggunaan daya listrik berbasis Internet of Things," *Circuit: Jurnal Ilmiah Pendidikan Teknik Elektro*, vol. 6, no. 1, pp. 54–66, 2022, doi: 10.22373/crc.v6i1.10862.
- [8] F. Condon et al., "Design and implementation of a cloud-IoT-based home energy management system," *Sensors*, vol. 23, no. 1, p. 176, 2023, doi: 10.3390/s23010176.
- [9] F. Dinmohammadi, A. M. Farook, and M. Shafiee, "Improving energy efficiency in buildings with an IoT-based smart monitoring system," *Energies*, vol. 18, no. 5, p. 1269, 2025, doi: 10.3390/en18051269.
- [10] H. R. Khan et al., "A low-cost energy monitoring system with universal compatibility and real-time visualization for enhanced accessibility and power savings," *Sustainability*, vol. 16, no. 10, p. 4137, 2024, doi: 10.3390/su16104137.
- [11] M. K. Hasan et al., "Internet of Things-based smart electricity monitoring and control system using usage data," *Wireless Communications and Mobile Computing*, vol. 2021, no. 1, p. 6544649, 2021, doi: 10.1155/2021/6544649.
- [12] K. Adyana Putra, I. N. Satya Kumara, and I. G. A. P. Raka Agung, "Rancang bangun sistem monitoring tegangan, arus, dan kecepatan mobil listrik Agnijaya Weimana," *Jurnal SPEKTRUM*, vol. 10, no. 4, p. 205, 2023, doi: 10.24843/SPEKTRUM.2023.v10.i04.p25.
- [13] P. W. Hendrawan and N. P. Agustini, "Simulasi kendali dan monitoring daya listrik peralatan rumah tangga berbasis ESP32," *ALINIER: Journal of Artificial Intelligence & Applications*, vol. 3, no. 1, pp. 54–68, 2022, doi: 10.36040/aliner.v3i1.4855.
- [14] R. Ande et al., "Internet of Things: Evolution and technologies from a security perspective," *Sustainable Cities and Society*, vol. 54, p. 101728, 2020, doi: 10.1016/j.scs.2019.101728.
- [15] F. Baskoro, M. Rohman, and A. P. Nurdiansyah, "Serial Peripheral Interface (SPI) communication application as output pin expansion in Arduino Uno," *INAJEEE (Indonesian Journal of Electrical and Electronics Engineering)*, vol. 3, no. 2, pp. 63–69, 2020, doi: 10.26740/inajeee.v3n2.p63-69.
- [16] Z. Haq, S. H. Wijoyo, and K. Rahman, "Pengembangan e-modul pembelajaran 'Informatika' menggunakan metode Research and Development (R&D)," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 4, pp. 1883–1891, 2023.
- [17] M. A. S. Ekowati et al., "Pengembangan sistem kelas cerdas berbasis Internet of Things (IoT) untuk proses pembelajaran tingkat SMP di Kota Surakarta," *JITU: Journal Informatic Technology and Communication*, vol. 7, no. 1, pp. 20–33, 2023.
- [18] Abdurrahman, "Metode Research and Development dalam pendidikan Islam," *RABBAYANI: Jurnal Pendidikan dan Peradaban Islami*, vol. 4, no. 1, pp. 26–41, 2024.
- [19] M. Y. Iqbar and K. P. K. Riyanti, "Rancang bangun lampu portable otomatis menggunakan RTC berbasis Arduino," *Antivirus: Jurnal Ilmiah Teknik Informatika*, vol. 14, no. 1, pp. 61–72, 2020, doi: 10.35457/antivirus.v14i1.1115.
- [20] Y. Mongdong, G. Mangindaan, and S. Silimang, "Analysis of gross power using digital and analog kWh meters," 20
- [21] R. S. Lubis, N. A. K. Harahap, and M. Putri, "Perancangan sistem proteksi beban lebih menggunakan TOR pada motor induksi 1 fasa untuk mesin pengepres limbah kaleng," 2024.
- [22] H. Nur Syamsi Hidayat et al., "Implementasi kendali dan monitoring sistem proteksi arus listrik pada extension kabel berbasis Node-Red sebagai upaya pencegahan kebakaran," *Venus: Jurnal Publikasi Rumpun Ilmu Teknik*, vol. 2, no. 6, pp. 74–86, 2024, doi: 10.61132/venus.v2i6.627.
- [23] T. O. Priyono and Sumardi, "Monitoring dan pengendalian beban listrik berbasis Internet of Things (IoT) menggunakan ESP32," vol. 11, no. 1, 2023.
- [24] R. Hilmansyah, R. M. Utomo, and A. W. Saputra, "Rancang bangun wireless battery monitoring system berbasis ESP32," 2020.
- [25] S. Antara et al., "Controlling of capacity household electricity using ACS758 sensor," *Patria Artha Technological Journal*, vol. 8, no. 1, pp. 15–25, 2024, doi: 10.33857/patj.v8i1.808.
- [26] D. R. Deni, M. A. Barata, and Sahri, "Forecasting metode single exponential smoothing dalam meramalkan penjualan barang," *Jurnal Informatika Polinema*, vol. 9, no. 4, pp. 435–444, 2023, doi: 10.33795/jip.v9i4.1405.
- [27] E. Lin et al., "UntrustIDE: Exploiting weaknesses in VS Code extensions," in *Proceedings 2024 Network and Distributed System Security Symposium*, San Diego, CA, USA: Internet Society, 2024, doi: 10.14722/ndss.2024.24073.
- [28] Suparmono et al., "Control system design for 3 storey elevator using PLC OPTA Finder," *International Journal of Research in Vocational Studies (IJRVOCAS)*, vol. 4, no. 4, pp. 73–78, 2024, doi: 10.53893/ijrvocas.v4i4.384.
- [29] S. Watmah, H. A. A. Kautsar, and S. Suryanto, "Perancangan sistem keamanan brankas menggunakan metode finger print berbasis SMS gateway," *INSANtek*, vol. 2, no. 2, pp. 35–39, 2021, doi: 10.31294/instk.v2i2.907.
- [30] M. N. Zakaria et al., "Implementasi algoritma ElGamal untuk pengamanan data pada wireless sensor network," *JURNAL ELTEK*, vol. 23, no. 1, pp. 24–31, 2025, doi: 10.33795/eltekv23i1.6910.