Prototype of IoT Application for Monitoring Street Lighting by Utilizing Solar Power

Fattah Satrio Atmojo[1], Nurchim[2], Pramono[3]
[1,2,3]Teknik Informatika, Fakultas Ilmu Komputer, Universitas Duta Bangsa Surakarta
Email : 210103131@mhs.udb.ac.id[1], nurchim@udb.ac.id[2], pramono@udb.ac.id[3]
Coresponden Author: 210103131@mhs.udb.ac.id

Diterima Redaksi: 13 Juni 2025 Revisi Akhir: 07 Juli 2025 Diterbitkan Online: 26 Juli 2025

Abstract – *Street lighting plays a vital role in ensuring the safety and comfort of road users, especially at night. Manual monitoring of streetlights is considered inefficient as it requires significant time and effort. This study proposes an automatic monitoring system based on the Internet of Things (IoT) to enhance the effectiveness of streetlight supervision. The system is built using an ESP32 microcontroller integrated with a current sensor, voltage sensor, and light sensor (LDR). The device connects to the internet and transmits data in real-time to a cloud-based platform, where it is visualized on a monitoring dashboard. Test results show that the system can accurately detect the on/off status of the lights, with LDR readings ranging from 0 to 4045, current sensor showing 0.2 amperes, and voltage sensor recording 7.4 volts. Additionally, the system can automatically control the relay to switch the lights on or off based on sensor input. Overall, the proposed system improves the efficiency of monitoring, enables faster detection of lighting conditions, and supports more timely and energy-efficient maintenance of street lighting infrastructure.*

Keywords *— Street light monitoring, Esp32, Internet of Things (IoT)*

1. INTRODUCTION

The development of digital technology has changed various transformations in various aspects of life, including in urban infrastructure systems such as public street lighting (PJU). Optimal street lighting is essential in supporting the safety and comfort of road users, especially at night. However, conventional lighting systems still have many limitations, especially in terms of energy efficiency and real-time device condition monitoring capabilities [1]. This condition often causes the lights to turn on continuously without adequate supervision, resulting in energy waste and potential safety risks when the lights are damaged.

In the past five years, the application of Internet of Things (IoT) technology in public street lighting systems has grown rapidly, especially with the integration of renewable energy sources such as solar panels. Anguraj et al. [2] designed an intelligent streetlight system that utilizes solar energy and is equipped with light intensity sensors and cloud-based connectivity, which enables automatic control and remote monitoring. The system is proven to improve energy efficiency and support the concept of a sustainable smart city. Furthermore, a literature review by Kusakana and Vermaak [3] emphasizes the importance of selecting power-efficient communication protocols such as LoRa and ZigBee in an IoT architecture for street lighting, to ensure reliable remote connections without high power consumption. These two references show that the combination of sensors, microcontrollers, communication networks, and renewable energy is key in the development of efficient and sustainable modern street lighting systems.

Public Street Lighting (PSL) is a vital infrastructure that plays an important role in improving the safety and comfort of road users, especially at night [4]. However, conventional PJU systems still face various obstacles such as delays in detecting lamp damage, inefficient use of energy, and limitations in real-time monitoring and control [5]. The main problem faced today is the unavailability of an efficient and integrated monitoring system automatically to know the condition of street lights directly. Manual monitoring requires a lot of labor, time, and cost [4].

On the other hand, inaccuracy in detecting faults in lights can cause delays in repairs and reduce the quality of public services [6]. In high-activity areas such as Keputih, Surabaya, this condition is exacerbated by time-consuming and labor-intensive manual monitoring methods, so that often dead lights are not immediately addressed, resulting in increased risk of accidents and crime [5]. In addition, power supply disruptions also cause PJU lights to go out, which negatively affects the quality of public services and environmental safety [4]. To overcome these problems, an Internet of Things (IoT)-based system has been developed to efficiently and remotely monitor lamp conditions, current, voltage, and environmental parameters [5]..

Based on this background, the purpose of this research is to design and build an IoT-based street lighting monitoring system that is able to monitor the status of the lights in real time, detect interference, and support energy efficiency. This system is expected to be a solution in improving the quality of street lighting management and supporting the implementation of the smart city concept that is energy efficient and responsive to community needs.

2. RESEARCH METHODS

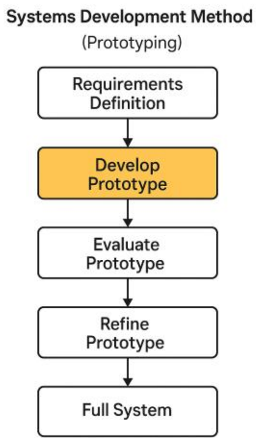
2.1. Types and Sources of Data

The research conducted utilized two main data sources, namely primary data and secondary data. Primary data was obtained directly through the testing process of the monitoring system designed by utilizing light sensors (LDR), current sensors, and voltage sensors, all of which are controlled by the ESP32 microcontroller. Data collection is based on observations of the device's performance in various environmental conditions. In addition, real-time data sent to a cloud-based storage platform was also studied to assess the effectiveness of the system. This observation process aims to measure the system's ability to automatically and responsively detect and report the operational conditions of the lighting fixtures.

Meanwhile, secondary data was collected from various relevant literature sources, such as research reports, previous scientific journals, and technical documentation regarding the application of IoT in energy systems and infrastructure. This information is used to strengthen the theoretical basis and provide comparisons in analyzing the performance of the developed system. The use of secondary data also supports the system design process to comply with applicable technical standards and ensure the validity of the research results.

2.2 Research Method

The prototyping method is an approach in systems research and development that focuses on creating an initial model (prototype) of the system to be built. The main purpose is to provide a real picture of how the system will work, so that it can be tested and evaluated early before the final system is fully developed. In this research, the development was carried out using a prototyping model. The prototype model is one of the system life cycle methods based on the concept of a working model [7], because it allows developers to conduct incremental testing, receive user feedback, and correct deficiencies in a literative manner following the stages:



Figurer 1. Prototyping Methode

3. RESULTS AND DISCUSSION

The initial stage is to identify the needs of the system, namely designing a device capable of monitoring the condition of street lighting automatically and in real time [6], where the main parameters monitored include voltage, current, light intensity, and the status of the lights on or off, and choosing the ESP32 microcontroller because it has a Wi-Fi connection to send data to a cloud-based server such as Supabase. After the needs are determined, a quick design is made in the form of a schematic of the relationship between components such as sensors, microcontrollers, and monitoring dashboards, complete with workflows and an overview of the website interface to display data in real time. The prototype was then built by assembling the voltage, current, and LDR sensors to the ESP32, programming them to read and send data to the cloud, and starting the development of the website to display device information visually and interactively. The completed prototype is tested in the evaluation stage to ensure the accuracy of sensor readings, the stability of data transmission, and the feasibility of the dashboard, then the deficiency records are used to improve the prototype both in terms of hardware and software, such as sensor calibration, signal amplification, and website display improvements to make it more responsive and stable, as well as adding new features such as notifications if necessary. The last stage is implementation and documentation, which is the implementation of the prototype on a small scale and the preparation of complete documentation including the dashboard. The last stage is implementation and documentation, which is the application of prototypes on a small scale and the preparation of complete documentation including tool diagrams, program code, test results, and dashboard views, so that this documentation becomes an important part of preparing final reports and scientific articles.Results and Discussion

3.1 System Requirement Identification

3.1.1 Blok Diagram

In this process, the measurement of light intensity, voltage, and current is carried out, the system will read the input from the ldr sensor, current and voltage which will be continued by the system to turn on the relay.

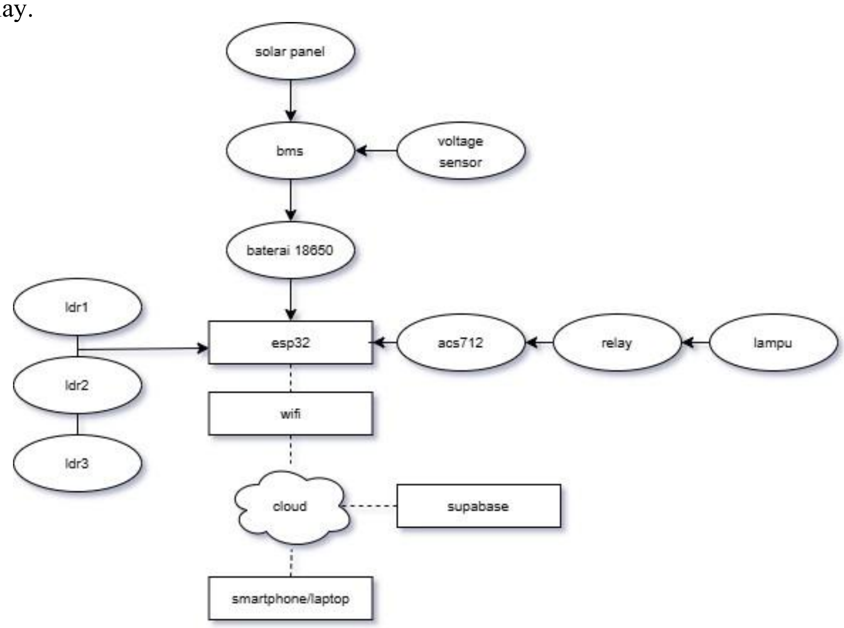


Figure 2. Architecture Tools

3.1.1.1 The block diagram of the system framework above can be explained as follows:

3.1.1.2 Solar panel serves as the main energy source for the system. This solar panel converts sunlight energy into DC electrical energy which is used to charge the battery through the BMS [8].

3.1.1.3 BMS (battery management system) Is a battery management system that functions to regulate the charging and discharging process of the battery to keep it safe and extend battery life. BMS also protects the battery from overcharge and overdischarge [9]..

3.1.1.4 18650 Battery A type of lithium-ion battery that stores electrical energy from the solar panel and supplies power to the ESP32 and other components. This battery allows the system to continue functioning at night or when the weather is cloudy[10]..

3.1.1.5 Voltage Sensor Used to measure battery voltage. This sensor helps the system know the status of the battery power capacity, whether it is sufficient to turn on the lights or needs recharging [11]..

3.1.1.6 LDR (Light Dependent Resistor) Sensor consists of 3 LDR sensors that function to detect the light intensity of the environment. The higher the light (daytime), the resistance of the LDR will decrease, and vice versa at night. Data from LDR is used by ESP32 to regulate when the lights turn on and off automatically [12].

3.1.1.7 ESP32 Is the main microcontroller in this system. ESP32 processes all data from sensors (LDR, voltage sensor, current), controls relays to turn lights on or off, and sends data to the cloud via WiFi connection. ESP32 is also power efficient and has integrated WiFi and Bluetooth features[4].

3.1.1.8 Current Sensor712 This sensor functions to measure the amount of electric current flowing into the lamp. This information is important for monitoring power consumption and detecting if a fault occurs, such as overcurrent [1].

3.1.1.9 Relay is an electronic switch controlled by ESP32 to connect or disconnect electricity to the lights. When night falls, the relay will be active to turn on the lights, and during the day, the relay will be off [13].

3.1.1.10 Lampu Komponen output dari sistem. Lampu akan menyala otomatis saat malam hari dan mati saat siang sesuai perintah dari ESP32 yang berdasarkan pembacaan sensor LDR [6].

3.1.1.11 Lights The output component of the system. The lights will turn on automatically at night and turn off during the day according to commands from ESP32 based on LDR sensor readings [6]..

3.1.1.12 Smartphone/laptop Used by users to access system information in real time. Through the dashboard connected to the Supabase, users can monitor lamp conditions, energy consumption, and sensor status from anywhere [14].

3.1.2 Flowchart

Flowchart is a type of diagram that illustrates the algorithm or sequential instruction steps of a system [15]. The main purpose of a flowchart is to show how a program moves from one process to another.

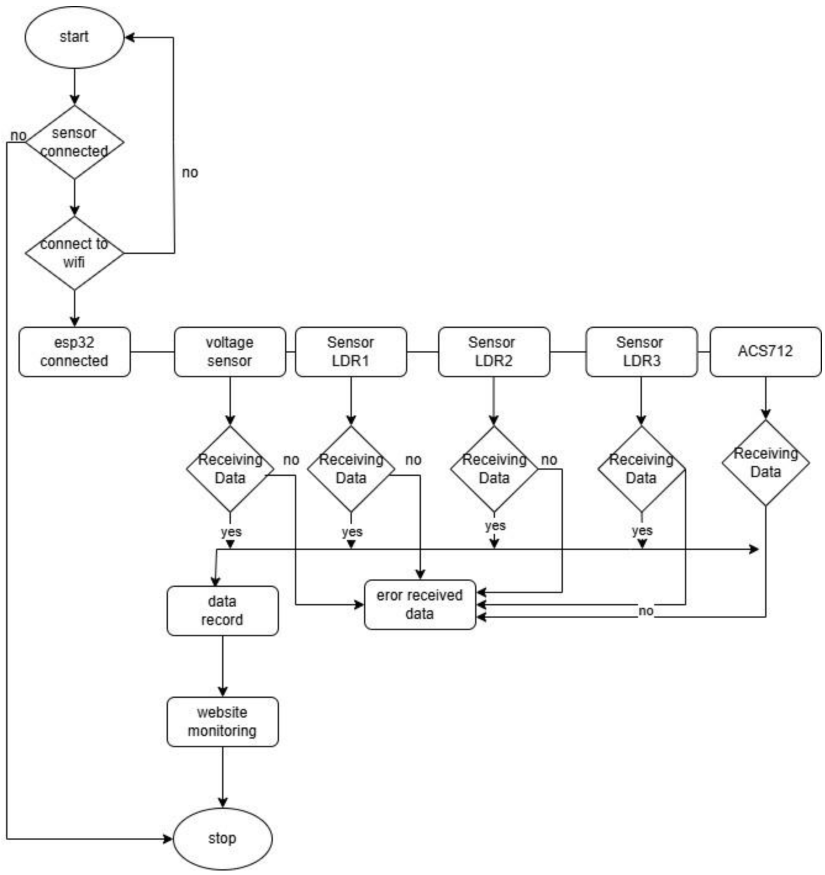


Figure 3. Sensor Reading Flowchart

Each sensor will be checked for successful data transmission. If all sensors send data correctly (Receiving Data = yes), the data will be recorded and sent to the web-based monitoring dashboard via the cloud. If there is an error or the sensor does not send data (Receiving Data = no), it will enter the error received data process for error handling. This process continues in a loop as long as the system is active.

3.2 Quick Design

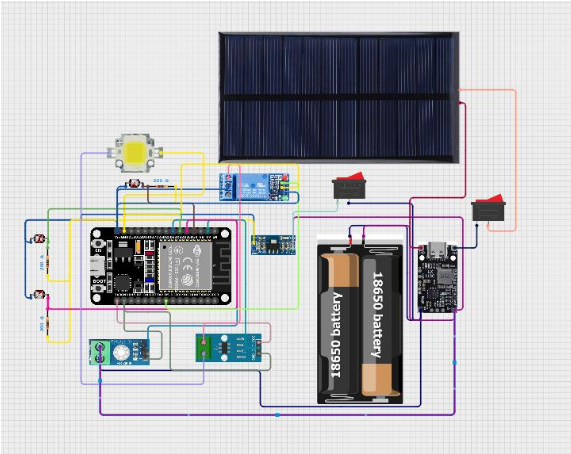


Figure 4. IoT-based Automatic Street Lighting System Monitoring Tool Circuit

Figure 4 shows the circuit schematic of an Internet of Things (IoT) system designed to monitor and manage street lighting automatically. The main energy comes from the solar panel which is channeled to an 18650-type rechargeable battery through a charge regulator module (TP4056). The system is controlled by an ESP32 microcontroller, which also has Wi-Fi connectivity for online data monitoring purposes.

The table 1,2,3 is circuit is equipped with various sensors, such as light sensors (LDRs) that detect day and night, current and voltage sensors to record power consumption, and environmental sensors such as temperature, humidity, and sound. Data from all sensors is transmitted by the ESP32 to the cloud platform for analysis and display via a dashboard.

The LED lights turn on automatically when the light intensity is low at night, and do not turn on during the day. This creates an energy-efficient and environmentally-friendly street lighting system that uses renewable resources and can be monitored in real-time.

.

Table 1. Relay circuit

Pin Relay	Pin ESP32	Pin Acs712
VCC	5V	-
GND	Ground 5V	-
IN	D5	-
COM	Vin	-
NO	-	-
NC	-	1

Table 2. LDR sensor circuit to ESP32

Pin	LDR1	LDR2	LDR3
VCC	5v	5v	5v
GND	Ground 5V	Ground 5V	Ground 5V
IN	D32	D34	D35

Table 3. ACS712 circuit

Pin	Relay	Led	ESP32
1	NC	-	-
2	-	+ (LED)	-
VCC	-	-	3V
GND	-	-	Ground 3V
OUT	-	-	VP

Table 4. Voltage sensor circuit

Pin	Bms	ESP32
GND	B-	
VCC	B+	
-		GND
S		D33

3.3 Prototyping

The design of this street lighting monitoring system involves the integration of several main components such as the ESP32 microcontroller, light sensor (LDR), voltage sensor, current sensor (ACS712), and WiFi module. Electrical energy is obtained from solar panels and stored in 18650 batteries managed by a battery management system (BMS). The ESP32 functions as a control center that reads data from all sensors, processes the information, and controls actuators such as relays to turn the lights on or off automatically. In addition, the ESP32 also sends data to the Supabase cloud platform so that it can be monitored in real time through user devices such as smartphones or laptops. The system is designed to be able to work independently and efficiently in monitoring environmental conditions as well as energy usage of street lighting.

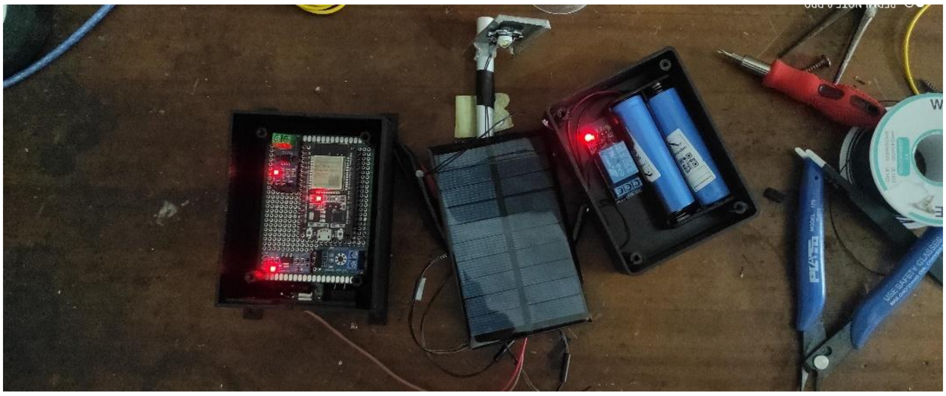


Figure 4. Prototype Monitoring Tool Runs

Figure 4 shows that the monitoring system is run through the provision of an 8 volt power supply from 2 18650 batteries arranged in series, the ESP32 is connected to the internet via a wifi network, and reads the installed sensors. The voltage divider reading at 8.4 volts means that the backup battery charging is running normally, the LDR sensor reading is still in bright conditions so that the relay connected to the 3v led does not turn on.

3.4 Prototype Evaluation and Testing

The previously designed monitoring device was successfully realized in the form of a functional tool. After the design and assembly phase is complete, the next step is to carry out the testing process to ensure that the system and program run as planned. Testing starts from the initial stage, namely ensuring the connection of the 3 ldr sensors can be connected properly to the ESP32 microcontroller as the system control center.

Table 5. LDR Sensor Testing

No	Light Conditions	LDR1 Value	LDR2 Value	LDR3Value	Status	Description
1	Bright (daytime)	4000	3985	3992	Normal	Sensor detects optimal light
2	Bright	3520	3555	3540	Normal	High light intensity
3	Bright (cloudy)	2890	2950	2920	Normal	Light is stable enough
4	Dim	1980	2050	2025	Normal	Light intensity starts to decrease
5	Dusk	1210	1195	1250	Normal	Natural light is fading

6	Dark	450	460	430	Normal	Sensor detects low light
7	Total Darkness	50	45	48	Normal	Almost no light
8	Artificial Light	2780	2755	2800	Normal	Response to lamplight
9	Partially blocked	1600	420	1580	LDR2 Error	LDR2 is blocked or damaged
10	One-sided Brightness	4000	3890	2450	LDR3 Tidak Stabil	LDR3 gets full light
11	Flashing Light	3100	2900	1000	LDR3 Tidak Stabil	Light fluctuation on LDR3
12	Sensor Not Read	3400	0	3390	LDR2 Error	LDR2 does not respond (loose wire?)

Voltage sensor testing is carried out to ensure that the device is able to read and transmit voltage data accurately from the power source used. This test is important to ensure that the monitoring system is able to provide real-time information about voltage conditions, so that it can support the overall performance of the tool optimally. Test results can be seen in table 6.

Tabel 6. Voltage Sensor Testing

No	Testing Conditions	Result(V)
1	When the battery finishes discharging	7.80037
2	When the battery finishes discharging with the solar panel	7.60037
3	When used in tests with the sensor on but the light on for ½ hour	7.20037
4	When used in tests with the lights on for ½ hour	6.40037

Testing the current sensor aims to ensure measurement accuracy and consistency. The first step is to prepare the tools and materials, such as the current sensor, current source, multimeter, and electrical load. After that, the sensor is properly installed in the circuit, ensuring a secure connection. Test results can be seen in table 7

Table 7. Current Sensor, Relay and Lamp Testing

No	Relay Status	Lamp Status	Result(A)
1	ON	ON	0.222421
2	ON	ON	0.200015
3	ON	ON	0.201305
4	ON	ON	0.210975
5	ON	ON	0.207108
6	OFF	OFF	0
7	OFF	OFF	0
8	OFF	OFF	0
9	OFF	OFF	0
10	OFF	OFF	0

When all the equipment is assembled for testing, the process begins by ensuring that all components are correctly and securely installed. The current sensor is attached to the circuit, with a secure connecting cable, and the connections are checked that there are no loose connections. The current source or power supply is then connected, followed by a suitable electrical load, such as a resistor or current-consuming device. The tresult of testing all sensor and tool components can be seen in table 8.

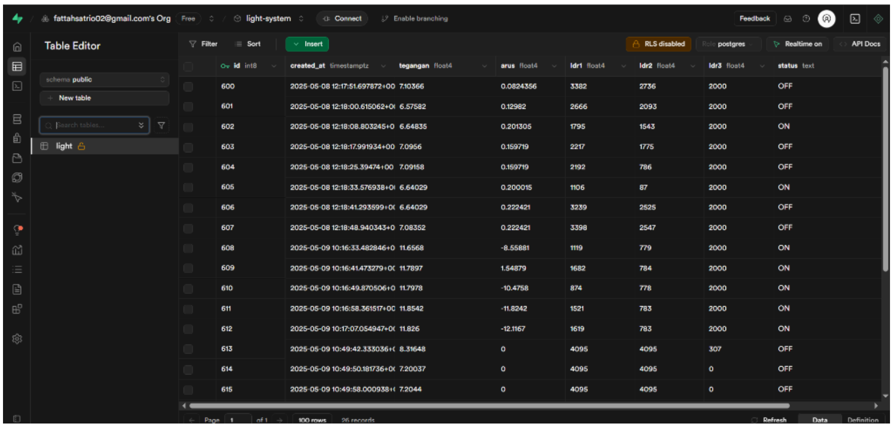
Table 8. Testing All Sensor and Tool Components

No	Voltage(V)	Current (A)	LDR1	LDR2	LDR3	Relay	Led
1	7.68086	0	3777	4095	74	OFF	OFF
2	7.60037	0	4010	3600	2000	OFF	ON
3	7.74561	0.217143	1119	779	2000	ON	ON
4	7.89877	0.200015	1106	87	2000	ON	ON
5	7.84465	0.222421	1053	714	109	ON	OFF
6	7.22051	0.159719	1600	1500	1850	ON	ON
7	8.31648	0	4095	4095	307	OFF	OFF
8	7.31648	0.238515	1266	0	2000	ON	ON
9	7.31648	0.146732	0	0	2000	ON	ON
10	7.58423	0	4001	4095	201	OFF	OFF

Based on the data obtained int table 8, the IoT-based streetlight monitoring system shows a good ability to detect environmental conditions and regulate the on-off of lights automatically. When the light sensor (LDR) value is low indicating dark conditions, the system activates the relay and the current starts flowing, turning on the lights as shown in lines 3, 4, 6, 8, and 9. Conversely, when the LDR value is high indicating bright conditions, the relay and LED are OFF, and the current = 0 A, as in lines 1, 2, 7, and 10. This shows that the system automation logic works according to the programming. However, there is an anomaly on line 5, where even though the relay is ON and current is detected, the LED is not lit, indicating a possible fault in the sensor or output module.

Several technical risks were identified in this system. First, sensor reading errors or noise can lead to inappropriate system decisions, such as in the case of the LED not lighting up even though the input indicates otherwise. Secondly, faulty or unconnected sensors, as shown in row 9 with LDR value = 0, can cause the system to turn on the lights continuously without considering the actual conditions, resulting in energy waste. Thirdly, relays as mechanical components have a limited lifespan and are at risk of failing in the long run. Finally, the reliance on internet connection carries the risk of failure of the monitoring system in the event of a network disruption, as well as potential data loss if not equipped with a local storage system.

3.5 Prototype Refinement



id int8	created_at timestamptz	tegangan float4	arus float4	ldr1 float4	ldr2 float4	ldr3 float4	status text
600	2025-05-08 12:17:51.697872+00	7.10366	0.0824356	3382	2736	2000	OFF
601	2025-05-08 12:18:00.615062+0(	6.57582	0.12982	2666	2093	2000	OFF
602	2025-05-08 12:18:08.803245+0	6.64835	0.201305	1795	1543	2000	ON
603	2025-05-08 12:18:17.991934+00	7.0956	0.159719	2217	1775	2000	OFF
604	2025-05-08 12:18:25.39474+00	7.09158	0.159719	2192	786	2000	OFF
605	2025-05-08 12:18:33.576938+0(	6.64029	0.200015	1106	87	2000	ON
606	2025-05-08 12:18:41.293599+0(	6.64029	0.222421	3239	2525	2000	OFF
607	2025-05-08 12:18:48.940343+0	7.08352	0.222421	3398	2547	2000	OFF
608	2025-05-09 10:16:33.482846+0	11.6568	-8.55881	1119	779	2000	ON
609	2025-05-09 10:16:41.473279+0(	11.7897	1.54879	1682	784	2000	ON
610	2025-05-09 10:16:49.870506+0	11.7978	-10.4758	874	778	2000	ON
611	2025-05-09 10:16:58.361517+00	11.8542	-11.8242	1521	783	2000	ON
612	2025-05-09 10:17:07.054947+0(	11.826	-12.1167	1619	783	2000	ON
613	2025-05-09 10:49:42.333036+(	8.31648	0	4095	4095	307	OFF
614	2025-05-09 10:49:50.181736+0(	7.20037	0	4095	4095	0	OFF
615	2025-05-09 10:49:58.000938+(	7.2044	0	4095	4095	0	OFF

Figure 5. Supabase data

In Figure 5. shows that the test runs according to the expectations of the researcher, namely turning on the relay, then turning on the lights and sending notifications to the database according to the value of the sensor readings installed.

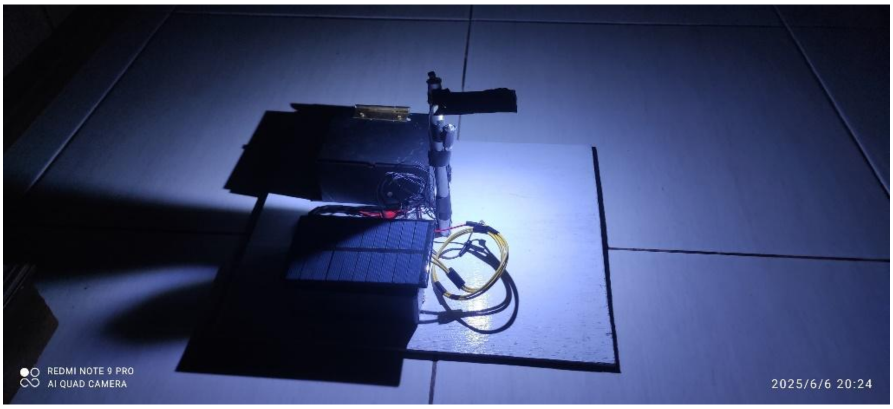


Figure 6. Monitoring Tool Running Relay

Figure 6 shows that the reading results of the LDR1 and LDR2 sensors are less than the specified threshold which is less than 2000 so that the relay that functions as a switch is active, and the 3v LED lights up.

3.6 Implementation and Documentation

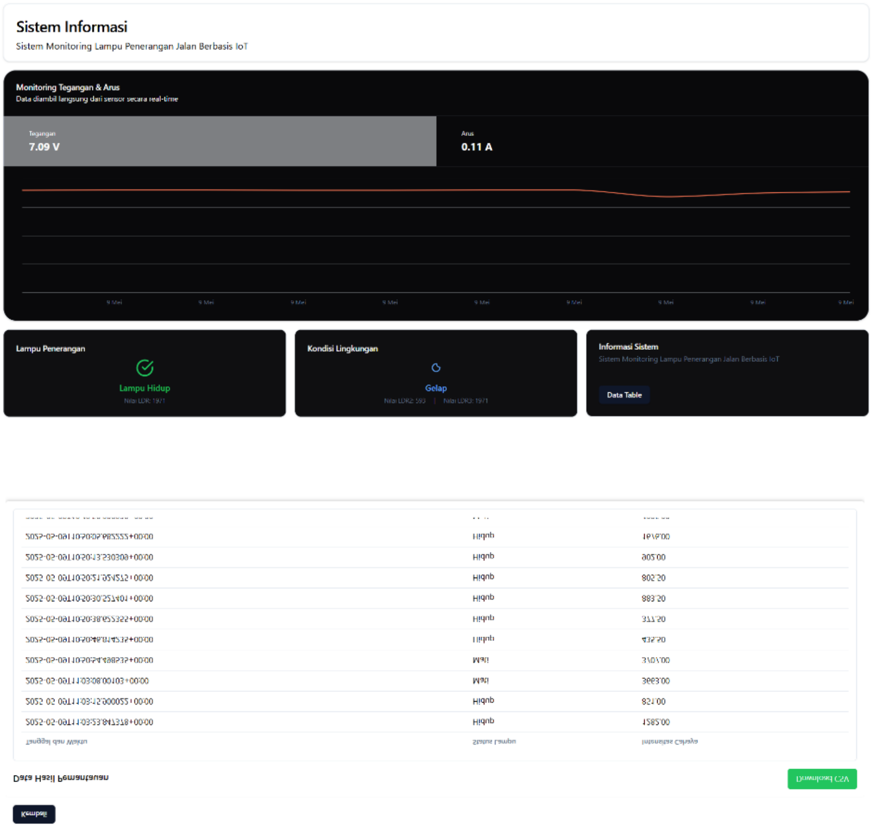


Figure 7. Monitoring Website

Figure 7 presents live monitoring of streetlight conditions through IoT technology. Users can see the operational status of the lights (on, off, or problematic) indicated by visual indicators. Important information such as amperage, voltage, and lighting levels around the lamp are displayed in real-time. In addition, a table presents a complete list of lights with details of their current state, including status, current, voltage, and light intensity, all of which are updated automatically.

The developed monitoring system proved to be more economical than conventional PLC or SCADA-based systems, with a manufacturing cost of around Rp 600,000 per unit. The use of ESP32 and simple sensors enables real-time monitoring with low operational costs. Compared to commercial solutions, the system offers energy efficiency through solar power and up to 40% reduction in maintenance costs, making it an effective option for wide-scale implementation in budget-constrained areas.

4. CONCLUSIONS

Based on the results of the research that has been conducted, it can be concluded that:

1. LDR sensor is able to detect the light intensity of the environment well;
2. LDR sensors can be effectively utilized to measure ambient lighting conditions;
3. Sensor reading data is successfully sent to the database and displayed in real-time on the web-based monitoring dashboard.

Thus, the main objective of this research has been achieved, which is to enable users or technicians to monitor the system in real time through the website. In addition, the automation system also runs as expected, where the lights will turn on automatically when the light intensity is below a predetermined threshold, and the condition of the electric current can be monitored when the lights are active.

5. ADVICE

This research still has room for further development to improve the functionality and effectiveness of the system. First, it is recommended that the system be equipped with an automatic notification feature, such as via email or text message, which will provide alerts if anomalies are detected in the system, such as unstable voltage or lights not turning on. This feature will greatly assist officers in responding to disturbances quickly and appropriately. Secondly, for the system to be implemented in areas with limited Wi-Fi network access, the use of communication technologies such as LoRa or NB-IoT is highly recommended. These technologies have a wider range and low power consumption, making them more suitable for monitoring needs in outdoor or remote areas.

REFERENCES

[1] A. Adam, M. Muharnis, A. Ariadi, and J. Lianda, "Penerapan IoT untuk Sistem Pemantauan Lampu Penerangan Jalan Umum," *Elinvo (Electronics, Informatics, Vocat. Educ.*, vol. 5, no. 1, pp. 32–41, 2020, doi: 10.21831/elinvo.v5i1.31249.

[2] T. E. Achar, C. Rekha, and J. Shreyas, "Smart automated highway lighting system using IoT: a survey," *Energy Informatics*, vol. 7, no. 1, 2024, doi: 10.1186/s42162-024-00375-7.

[3] Mila Ilieva-Obretenova and Radi Pipev, "IoT Management for Street Lighting—Information Model—Part II," *Manag. Stud.*, vol. 11, no. 5, pp. 253–261, 2023, doi: 10.17265/2328-2185/2023.05.002.

[4] D. T. Listrik, F. Vokasi, U. N. Surabaya, D. T. Listrik, F. Vokasi, and U. N. Surabaya, "Rancang Bangun Sistem Kontrol dan Monitoring Lampu Penerangan Jalan Umum (PJU) Menggunkan Yolo dan Node-red Achmad Malikur Robbani Musthofa Widi Aribowo , Reza Rahmadian , Ayusta Lukita Wardani," pp. 95–101, 2020.

[5] A. A. Hidayah, "Monitoring Arus Dan Tegangan Pada Pengisian Lampu Penerangan," vol. 18, no. I, pp. 48–54, 2025, doi: 10.5281/zenodo.15052833.

[6] Q. Hidayati, N. Jamal, N. Yanti, and Y. Tri Prasetyo, "Sistem monitoring dan kontrol penerangan jalan umum tenaga surya berbasis Internet of Things," *JITEL (Jurnal Ilm. Telekomun. Elektron. dan List. Tenaga)*, vol. 3, no. 1, pp. 19–26, 2023, doi: 10.35313/jitel.v3.i1.2023.19-26.

[7] F. O. Dayera, Musa Bundaris Palungan, "G-Tech : Jurnal Teknologi Terapan," *G-Tech J. Teknol. Terap.*, vol. 8, no. 1, pp. 186–195, 2024, [Online]. Available: https://ejournal.uniramalang.ac.id/index.php/g-tech/article/view/1823/1229

[8] M. A. Prasetyo and H. K. Wardana, "Rancang Bangun Monitoring Solar Tracking System Menggunakan Arduino dan Nodemcu Esp 8266 Berbasis IoT," *Resist. (Elektronika Kendali Telekomun. Tenaga List. Komputer)*, vol. 4, no. 2, p. 163, 2021, doi: 10.24853/resistor.4.2.163-168.

[9] I. Ihsan and A. wahyu Aditya, "Rancang Bangun Battery Monitoring System (BMS) berbasis LabVIEW," *JTT (Jurnal Teknol. Terpadu)*, vol. 9, no. 1, pp. 44–49, 2021, doi: 10.32487/jtt.v9i1.972.

[10] M. T. Setiawan, I. Winarno, and B. Y. Dewantara, "Implementasi Internet Of Things Dalam Rancang Bangun Sistem Monitoring Pada Solar Cell Berbasis Web," *JEECOM J. Electr. Eng. Comput.*, vol. 3, no. 1, pp. 34–38, 2021, doi: 10.33650/jeecom.v3i1.1981.

[11] A. Mubarak 'aafi, J. Jamaaluddin, and I. Anshory, "Implementasi Sensor Pzem-017 Untuk Monitoring Arus, Tegangan dan Daya Pada Instalasi Panel Surya dengan Sistem Data Logger Menggunakan Google Spreadsheet dan Smartphone," *SNESTIK Semin. Nas. Tek. Elektro, Sist. Informasi, dan Tek. Inform.*, p. 191, 2022, [Online]. Available: https://ejurnal.itats.ac.id/snestikdanhttps://snestik.itats.ac.id

[12] S. Sintaro, A. Surahman, and C. A. Pranata, "Sistem Pengontrol Cahaya Pada Lampu Tubular Daylight

Berbasis Iot," *J. Teknol. dan Sist. Tertanam*, vol. 2, no. 1, p. 28, 2021, doi: 10.33365/jtst.v2i1.1034.

[13] A. Manik Dirgayusari and I. W. Sudiarsa, "Implementasi Sistem Monitoring dan Kontrol Suhu Kelembaban Ruang Budidaya Jamur Berbasis IoT," *J. Sist. Inf. dan Komput. Terap. Indones.*, vol. 4, no. 2, pp. 78–89, 2021, doi: 10.33173/jsikti.127.

[14] E. V. Haryanto and R. Puspasari, "Rancang Bangun Monitoring Penerangan Ruangan Menggunakan Kamera Berbasis Komputer Dengan Metode Fuzzy Logic," *It (Informatic Tech. J.*, vol. 4, no. 2, p. 192, 2017, doi: 10.22303/it.4.2.2016.192-201.

[15] M. Tinambunan and S. Sintaro, "Aplikasi Restfull Pada Sistem Informasi Geografis Pariwisata Kota Bandar Lampung," *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 2, no. 3, pp. 312–323, 2021, doi: 10.33365/jatika.v2i3.1230.