GSM-Enabled Integrated System for Real-Time Monitoring, Control, Diagnosing, and Troubleshooting of Street Lighting in Geographically Isolated and Disadvantaged Areas of Caraga Region, Philippines

Edgardo Ricardo B. Sajonia Jr.
Department of Electronics Engineering, College of Engineering and Geosciences, Caraga State University

Jive Ian A. Bogwat
Department of Electronics Engineering, College of Engineering and Geosciences, Caraga State University

Roderick C. Gutierrez
Department of Electronics Engineering, College of Engineering and Geosciences, Caraga State University

Roderick C. Gutierrez
Department of Electronics Engineering, College of Engineering and Geosciences, Caraga State University

他

https://doi.org/10.5109/7395597

出版情報：Proceedings of International Exchange and Innovation Conference on Engineering & Sciences (IEICES). 11, pp.748-753, 2025-10-30. International Exchange and Innovation Conference on Engineering & Sciences
バージョン：

GSM-Enabled Integrated System for Real-Time Monitoring, Control, Diagnosing, and Troubleshooting of Street Lighting in Geographically Isolated and Disadvantaged Areas of Caraga Region, Philippines

Edgardo Ricardo B. Sajonia Jr.[12], Jive Ian A. Bogwat[1], Roderick C. Gutierrez[1], Roderick C. Gutierrez[1], Jeffrey T. Dellosa[12], Alexander T. Demetillo[12], Hanzel B. Caringal[12]
[1]Department of Electronics Engineering, College of Engineering and Geosciences, Caraga State University, Ampayon, Butuan City, 8600, Philippines
[2]Center of Renewable Energy, Automation, and Fabrication Technologies Caraga State University, Ampayon, Butuan City, 8600, Philippines,
Corresponding author email: ebsajonia@carsu.edu.ph

Abstract: *This research presents the design, development, and deployment of a GSM-enabled integrated electronic system aimed at enhancing street lighting in the Caraga Region. The system uses solar-powered streetlights with IoT capabilities via the ESP32 TTGO T-Call microcontroller, addressing high maintenance costs and lack of real-time monitoring in Geographically Isolated and Disadvantaged Areas (GIDAs). It enables automated control and data tracking through a web and mobile dashboard. Key parameters monitored include battery state of charge (SOC), voltage, and current from the solar panel, load, and battery. Features like emergency 5V charging and deep sleep mode optimize performance and energy use. This scalable solution aligns with the Philippine government's goal of 100% electrification by 2028 and offers strong potential for smart infrastructure deployment in remote communities.*

Keywords: Solar-powered streetlights; ESP32 TTGO T-Call; GIDA; smart city; real-time monitoring

1. INTRODUCTION

The global demand for energy is projected to increase by 25% by 2040 due to urbanization and economic growth [1]. As a response to climate concerns and fossil fuel dependency, the adoption of renewable energy sources such as solar power is being adopted. Solar-powered streetlights, when integrated with Internet of Things (IoT) technologies, offer an energy-efficient, sustainable alternative that reduces carbon emissions and enables real-time monitoring to prevent energy waste [2]. Innovations in smart streetlighting have introduced features such as automatic lighting via LDR sensors, GSM-based monitoring in areas with limited connectivity, motion-based dimming, GPS-enabled fault detection, and emergency systems with CCTV and audio alerts [3]. These technologies aim to increase reliability, safety, and efficiency in public lighting systems.

In the Philippines, many communities remain unelectrified, particularly those classified as Geographically Isolated and Disadvantaged Areas (GIDAs) [4]. These regions, such as those in Caraga, the fourth poorest region in the country, some community suffer from poor infrastructure, unreliable lighting, and costly maintenance due to limited access and resources. The lack of real-time data results in inefficient energy use and delayed system repairs, posing safety and operational challenges. The National Electrification Administration (NEA) has committed to achieving total electrification by 2028 [5], highlighting the need for sustainable, remotely manageable lighting systems.

This study addresses these challenges by designing, developing, and deploying a smart solar-powered streetlight monitoring system tailored to the needs of GIDAs in the Caraga Region. The system integrates GSM communication, an ESP32 microcontroller, and a web and mobile-based graphical user interface (GUI) for real-time monitoring and control. Key features include energy-saving automation, emergency mobile charging, and deep-sleep power management, ensuring cost-efficiency, reliability, and scalability in off-grid areas.

The proposed system reduces the need for on-site inspections through remote diagnostics, supports infrastructure development aligned with national electrification goals, and demonstrates the feasibility of smart renewable-powered lighting in underserved areas. While the system enables real-time monitoring of electrical parameters and battery status, the study does not cover long-term charging efficiency under varying weather conditions or hardware durability recommendations for future work. Ultimately, this research contributes to advancing sustainable smart infrastructure and improving public safety and energy accessibility in remote Philippine communities.

2. RELATED WORKS

Street lighting in rural areas of developing countries, such as the Caraga Region in the Philippines, faces major challenges due to limited access to power grids. Many communities, particularly those in geographically isolated and disadvantaged areas (GIDAs), remain without electricity, resulting in safety risks and economic limitations after dark. Traditional street lighting systems are costly and unsustainable, requiring manual inspections that increase maintenance expenses. As a solution, this study highlights the use of solar-powered smart streetlights integrated with IoT technology, enabling real-time monitoring and automated control. These systems offer a reliable, energy-efficient, and low-maintenance alternative, ideal for deployment in off-grid and underserved areas.

Various research works have proposed different concepts for smart street light control systems, which are focused on surveillance of environmental parameters, automation, and remote access to the system. Prasanna et al. and Kumar et al. discuss intelligent streetlights that have different working times in a day depending upon the road traffic density [6]. The street light control system controls the streetlights automatically to turn ON/OFF the streetlights when cars/vehicles or their density are present on the road, which would reduce the wastage of more energy and over-consumption of power. As for the improvement of streetlight efficiency, Prince Devaraj states that it can be done with an automated system, whereby the streetlight will turn on only if it detects the presence of a person [7]. Sajonia and Dagsa enhanced system efficiency with Kalman filter estimation in IoT-based solar streetlight monitoring, demonstrating intelligent remote control and data logging to reduce maintenance costs and improve energy conservation [1]. Some of the smart solar streetlight systems in the prior arts have revealed various innovations. Zou et al. patented a design that incorporated the use of NB-IoT for remote monitoring with the integration of solar panels and sensors [9]. Yim Sung Yul exhibits a streetlight that is used as an electric vehicle charging station but happens to be not a cheap one, since it requires big batteries. On the contrary, the proposed study presents a solution that can be considered cost-efficient, focusing on energy efficiency with lower operating costs, and does not consider EV charging at all [10].

3. METHODOLOGY

This study follows a structured approach to design, develop, and deploy a smart solar streetlight monitoring system tailored for selected areas in the Caraga Region. The methodology involves integrating hardware components such as solar panels, batteries, sensors, microcontrollers, and GSM modules, with software platforms for web and mobile dashboards. The system enables real-time monitoring of streetlight status and electrical parameters to improve maintenance efficiency and energy management in remote areas.

INPUT	PROCESS	OUTPUT
• Current and voltage levels of battery, solar panel, and LED bulb. • Light Rays (captured by Photovoltaic cell.	• Data sent to the central server via GSM. • SSL status detection using time series data driven. • Deep sleep and autobooting. • 70% above for emergency charging triggering.	• Display current and voltage level of solar panel, battery, ad LED bulb • SSL status • Battery SOC • Web and mobile GUI display

Fig. 1. Framework of the study.

The model outlines the data flow and operation of the solar streetlight monitoring system. It captures inputs like light intensity, voltage, and current from sensors, processes them for energy optimization and control, and transmits data via GSM to a central server. Output includes real-time readings, streetlight status, battery state-of-charge, and a user interface for remote monitoring and automation.

3.1.2 System's Block Diagram

Fig. 2 shows the diagram of the systems where it portrays the logical flow of the components. The system integrates solar power generation, energy storage, and remote monitoring capabilities. Solar panels charge a 12V battery via a solar charge controller. A buck converter steps down the voltage to power the microcontroller and sensors. Current and voltage are monitored by ACS712 and INA219 sensors, respectively.

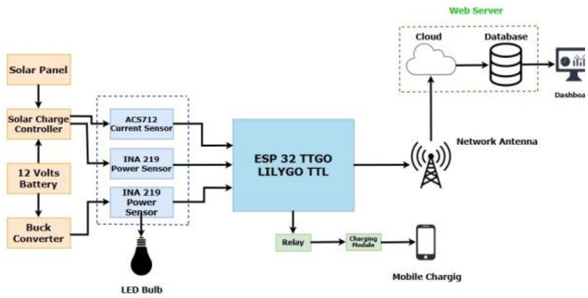

Fig. 2. Block diagram of the system.

An ESP32 microcontroller reads sensor data, controls the LED bulb, and transmits data wirelessly to a cloud server. This data is stored in a database and displayed on a remote dashboard, enabling real-time monitoring and control of the system's voltage and current data and device charging.

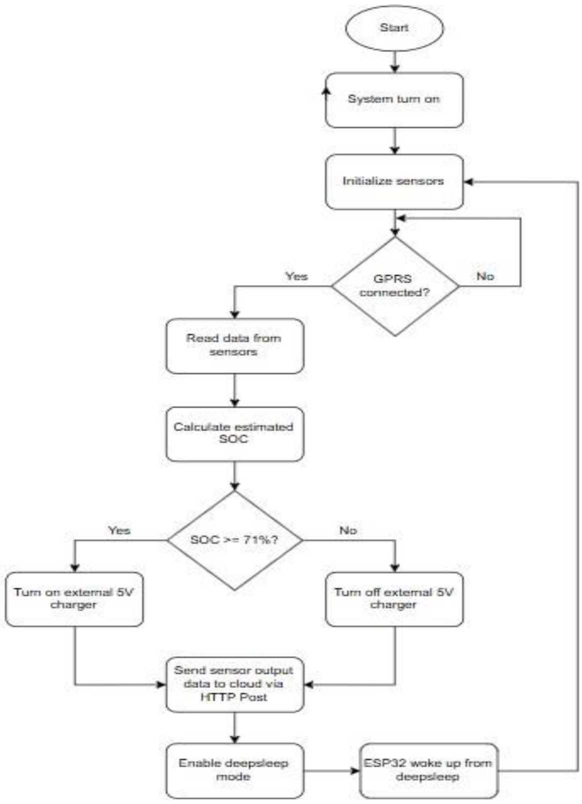

Fig. 3. Flowchart of the system

This solar-powered streetlight system, controlled by an ESP32 with onboard sensors and GSM, prioritizes efficient energy management. After sensor initialization, it establishes a GSM connection for cloud data transmission (voltage, current, battery SOC). If the battery SOC is ≥71%, an external 5V charger activates; otherwise, it deactivates to conserve power. Sensor data is then sent via HTTP POST to a cloud server for remote monitoring. Finally, the ESP32 enters deep sleep to minimize energy consumption, repeating this cycle autonomously.

3.1.4 System's Wiring Diagram

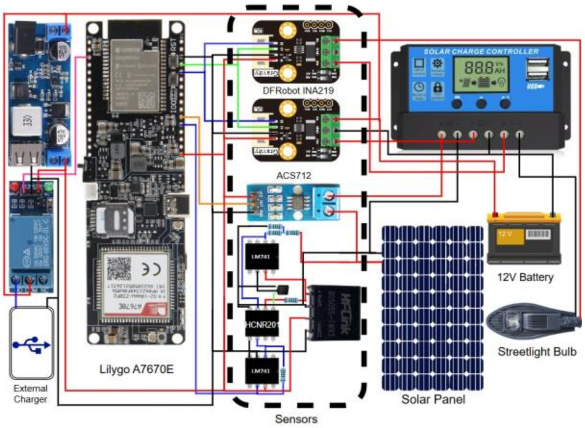

Fig. 4. Wiring diagram of the system.

3.1.5 Sensor Calibration and Data Analysis

This section shows the sensor calibrations of both INA 219 and the ACS712. To ensure the accuracy of data measurements, sensor calibration was conducted by comparing the readings of the INA219 DFRobot I2C module with those obtained from the GW Instek GDM-397 multimeter. Calibration parameters were adjusted based on this comparison to enhance the precision of both current and voltage readings from the INA219 DFRobot I2C module. The GW Instek GDM-397 multimeter was selected for this calibration process due to its high accuracy, with a stated error of ±0.5% + 1 digit, compared to the INA219's typical error margin of ±1%. This difference in accuracy makes the GDM-397 a good reference instrument for calibrating the INA219 DFRobot I2C, ensuring the reliability of the measurements used in the study.

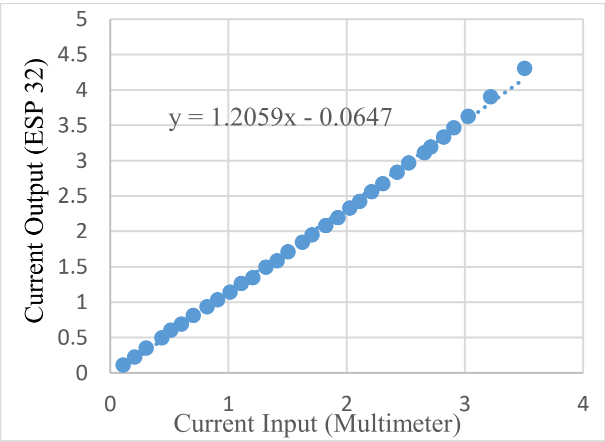

Fig. 5. ACS 712 Current Calibration.

Fig. 5 shows the linear fitting model using linear regression. This calibration method enhances data precision, aligned with the accuracy of the GW Instek GDM-397. The testing was conducted in a GSM-enabled area to ensure reliable data transmission. The linear equation presented serves as the basis for data correction during cloud-based processing. The input data are the current value set with adjustable power source and adjusted using the multimeter. The output is the voltage output sent by the ESP 32 to the server.

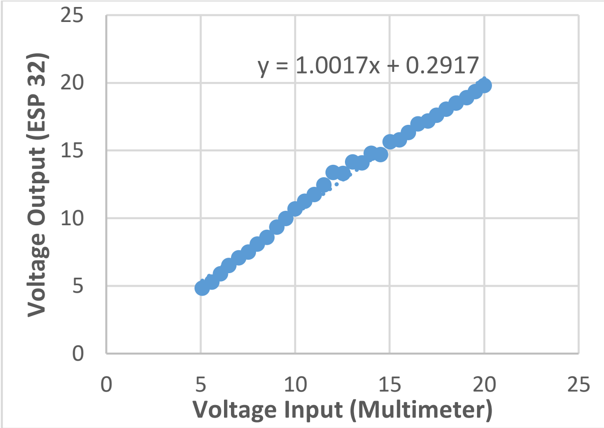

Fig. 6. ACS 712 Voltage Calibration

Fig. 6 shows the linear fitting model using linear regression. This calibration method enhances data precision, aligned with the accuracy of the GW Instek GDM-397. The testing was conducted in a GSM-enabled area to ensure reliable data transmission. The linear equation presented serves as the basis for data correction during cloud-based processing. The input data are the current value set with adjustable power source and adjusted using the multimeter. The output is the data sent by the ESP 32 to the server.

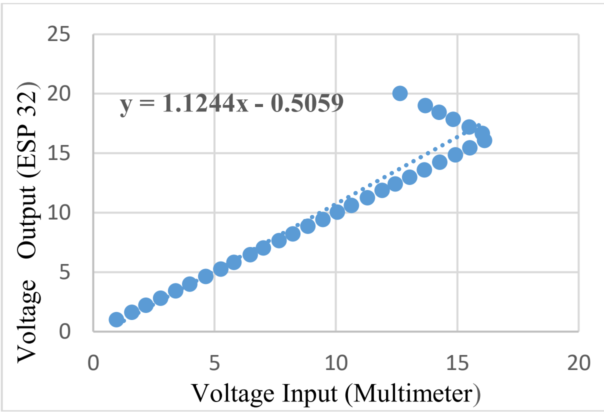

Fig. 7. INA 219 DFRobot I2C Voltage Calibration.

Fig. 7 presents the calibration graph of the INA219 DFRobot I2C sensor. The voltage readings show consistent behavior, with a gradual decline observed beyond 16 volts. However, this shift does not affect system performance, as the operational voltage range remains within 0–12 volts. The input data are the voltage value set with adjustable power source and adjusted using the multimeter. The output is the voltage value levels sent by the ESP 32 to the server.

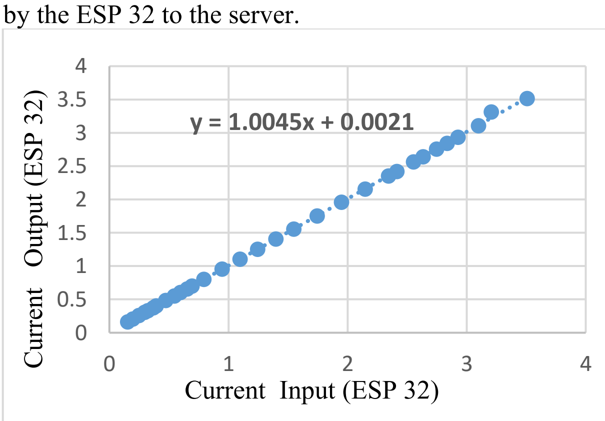

Figure 8. INA DFRobot 219 Current Calibration.

Fig. 8 presents the linear fitting model using linear regression. This calibration method enhances data precision, aligned with the accuracy of the GW Instek GDM-397. The testing was conducted in a GSM-enabled area to ensure reliable data transmission. The linear equation presented serves as the basis for data correction during cloud-based processing. The input data are the current value set with adjustable power source and adjusted using the multimeter. The output is the current output sent by the ESP 32 to the server.

3.2 Graphical User Interface Development

This section described the methods used in developing the web and mobile graphical user interfaces (GUI) for the smart solar streetlight system. It included the design flowcharts and the various software components involved in the development process.

In terms of programming languages, frameworks, and tools, the researcher utilized a range of technologies to support the system's software development. For the firmware, the Arduino IDE was chosen due to its intuitive and user-friendly interface, making it suitable for programming the ESP32 board used in this project. It facilitated effective firmware development and debugging, ensuring accurate sensor data acquisition and reliable GSM communication.

The backend API was implemented using PHP. The ESP32 microcontroller transmitted sensor data via HTTP (POST) requests to a dedicated PHP endpoint at predefined intervals. Upon receiving the data, the PHP script authenticated each request, parsed the payload, and stored the validated records in a MySQL relational database.

For the frontend, the GUI was developed using the Laravel framework. It accessed real-time data from the MySQL database and was built using HTML for structure, CSS for styling, and JavaScript for interactive control and dynamic monitoring. Laravel's Object-Relational Mapping (ORM) capabilities enabled efficient querying and rendering of the latest data into user-friendly charts, status indicators, and control elements. This architecture ensured seamless data management and a smooth user experience for real-time monitoring and control of the streetlight system.

3.3 Error Detection Using Time Series Data Driven

This section explains the method used by the researcher to obtain data related to the device's error detection. This feature will help troubleshoot potential system faults that may occur, enabling fault identification. Consequently, troubleshooting can be performed directly through the dashboard.

Table 1. Specifications adopted for simulated inverter

Error	Issue or Problem
Error:201	A possible issue is that the bulb does not work at night, as both the current and voltage drop to zero during that time.
Error: 212	A possible issue is that the bulb lights up during the daytime, with the current measuring between 1 to 0.9 amps and the voltage between 10–12 volts, respectively.
Error: 111	A possible issue is that the solar panel connections are broken or not functioning properly, resulting in the current and voltage values being approximately zero.
Error:311	Over three consecutive days, the state of charge (SOC) showed a decreasing trend during the daytime. A possible issue is that the bulb is illuminating during the day due to bad weather. If the weather is good, then the problem may be that the solar panel has a weak charging capacity.
Error: 312	Over three consecutive days, the state of charge (SOC) remained idle during the daytime. A possible issue is that the bulb is illuminating during the day despite good weather conditions.
Error: 313	The battery becomes fully charged in just 2 hours or less than 6 hours, whereas normal charging typically takes about 11 hours.
Error: 314	The sudden jumps or down of SOC values on every hour.
Error:305	The sudden jumps or down of SOC values on every hour

Table 2. Error code number representation

Error Legend	Digit 1: Component (1 = solar, 2 = bulb, 3 = battery), Digit 2: 1 = Day / 0 = night, Digit 3: Number of Error
Error:201	Bulb current = 0 A, bulb voltage = 0
Error: 212	Bulb current = 0.1 A, bulb voltage = 12V
Error: 111	Solar current = 0 A, Solar voltage = 0V

Tables 1 and 2 outline the system's error code structure for quick diagnostics. Table 2 lists error codes with descriptions for dashboard display during malfunctions. Table 3 explains each code's three-digit format: the first digit indicates the component (1–solar panel, 2–bulb, 3–battery), the second shows the time (1–day, 0–night), and the third is a unique event identifier. This format streamlines issue detection and maintenance and processed by the server side.

4. RESULTS AND DISCUSSIONS

This section presents the study's results on the device's design, software development, and deployment. It includes data analysis, system implementation, and categorized results on hardware, software, and experimental deployment. Fig. 9 and Fig. 10 show the IoT box used for solar streetlight monitoring. It houses a charge controller, antenna for cloud communication, and an ESP32 TTGO T-Display microcontroller for data processing. Internal components include a buck converter, transistors, resistors, and various sensors. The durable acrylic enclosure measures 150×150×60 mm.

Fig. 9. Actual hardware setup

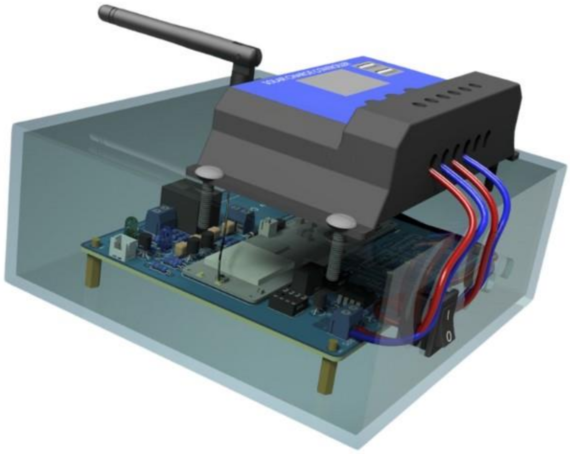

Fig. 10. Rendered 3D model illustrating the hardware layout.

Fig. 11 shows the complete integration of the hardware components, including the ESP32 microcontroller, ACS712 current sensor, INA219 sensor, buck converter, relay module, LEDs, resistors, and transistors.

Fig. 11. Final PCB prototype setup with component integration.

4.2 Graphical User Interface Results

This section presents the developed GUI for the solar streetlight monitoring system, highlighting the dashboard's layout and its functional features.

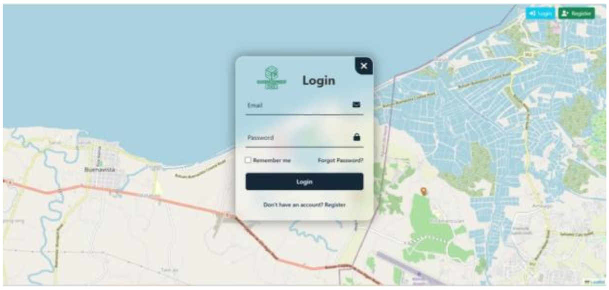

Figure 12. Login page of the Web GUI.

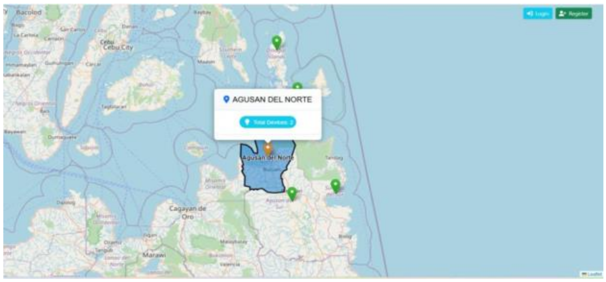

Figure 13. Province selection of the Web GUI.

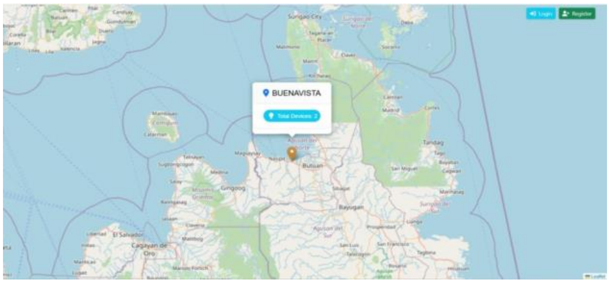

Figure 14. Municipality selection for the streetlight monitoring in Web GUI.

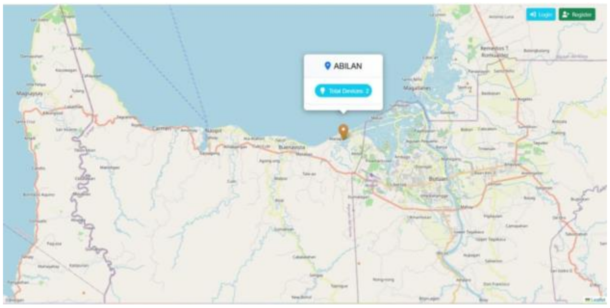

Figure 15. Barangay selection page.

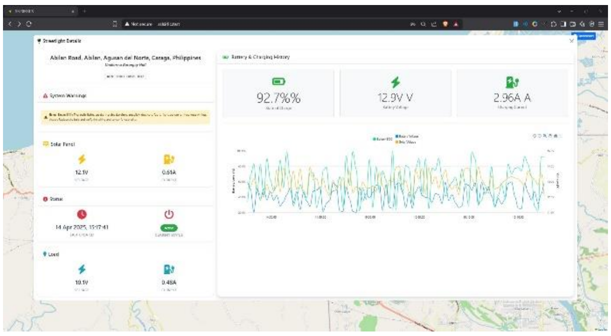

Figure 16. Streetlight details.

Fig. 12–16 display the web-based graphical user interface of the system. It begins with a login page, allowing users to create an account and sign in. Upon login, users navigate through a map-based selection menu, from province to municipality, then to barangay. The barangay menu lists associated streetlights, enabling users to view detailed information for each. The web and mobile GUI gave the same feature but only differ on the layout size.

4.3 Experimental Data Results

This section discusses the experimental data recorded in the database during the sample implementation of the device. The implementation was conducted in a location with GSM coverage, as the system is designed for deployment only in areas with available GSM connectivity.

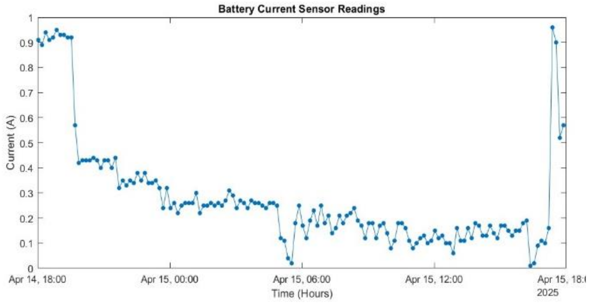

Fig. 17. Battery current data in 24 hours.

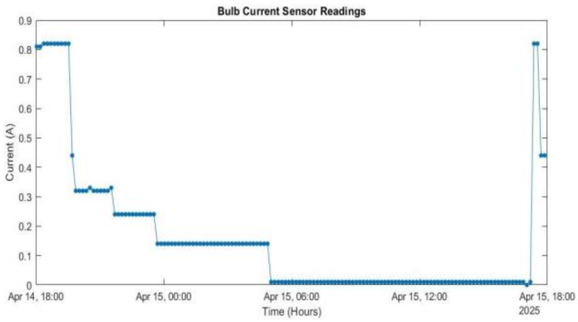

Fig. 18. Recorded current through the bulb in 24 hours.

Fig. 17 and 18 show the 24-hour current data trend. At night, the current gradually decreases as the system powers the load. During the day, the current fluctuates at a consistent level, reflecting normal operation as the microcontroller consumes power to send updates to the database.

5. CONLUSION

This study designed, developed, and implemented a smart solar-powered streetlight monitoring system tailored for GIDAs in the Caraga Region. Integrating GSM, an ESP32 microcontroller, and a web- and mobile-based GUI, the system enables real-time monitoring and control without relying on conventional internet infrastructure.

The solution addresses high maintenance costs, lack of real-time data, and inefficient energy use. Features such as deep sleep power management, sensor calibration, and emergency charging improve reliability, cost-efficiency, and scalability. The GUI minimizes on-site inspections, streamlining maintenance operations.

Aligned with the national goal of full electrification by 2028, the project supports smart infrastructure development and contributes to a more sustainable and connected future for underserved communities.

6. REFERENCES

[1] The International Energy Agency (IEA), "IEA Outlook Projects 25-Percent Increase in Energy Demand by 2040."

[2] R. Prasad, "Energy Efficient Smart Street Lighting System in Nagpur Smart City using IoT-A Case Study," *2020 5th International Conference on Fog and Mobile Edge Computing, FMEC 2020,* pp. 100-103, 2020, doi:10.1109/FMEC49853.2020.9144848.

[3] H. P. Khandagale, R. Zambare, P. Pawar, P. Jadhav, P. Patil, and S. Mule, "STREET LIGHT CONTROLLER WITH GSM TECHNOLOGY," vol. 4, no. 10, pp. 268–271, 2020.

[4] Department of Social Welfare and Development MIMAROPA (DSWD), "Geographically Isolated and Disadvantaged Areas".

[5] Business World, "NEA targets total electrification by 2028." [Online]. Available: https://www.bworldonline.com/economy/2023/01/31/502218/nea-targets-total-electrification- by-2028/

[6] P. Arjun, S. Stephenraj, N. Naveen Kumar, and K. Naveen Kumar, "A study on IoT based smart street light systems," *2019 IEEE International Conference on System, Computation, Automation and Networking, ICSCAN 2019*, PP. 1-7, 219, doi:10.1109/ICSCAN.2019.8878770.

[7] M.C. Viola, S. Mary, G. P. Devaraj, D. J. Pushparaj, and J.M. Esther, "Intelligent Energy Efficient Street Light Controlling System based on IoT for Smart City," *2018 International Conference on Smart Systems and Inventive Technology (ICSSIT)*, no. Icssit, pp. 551-554, 2018, doi:10.1109/ ICSSIT.2018.8748324

[8] E.R. Sajonia, Jr, and L.M. Dagsa, "IoT-Based Smart Street Light Monitoring System with Kalman Filter Estimation," 2021.

[9] Zhou et al., "Street Lamp Intelligent Measurement and Control Device.pdf," 2023.

[10] Y. S. YUL, "IoT Smart Street light system equipped with IoT function and capable of charging electric vehicles," 2023 [Online]. Available: https://worldwide.espacenet.com/patent/search/

[11] C. J. T. Niog, R. N. Mendoza, L. M. Dagsa, and J. C. Martinez, "IoT-based climate control system for cultivation of oyster mushroom (Pleurotus ostreatus) fruiting bag," *Int. Exch. Innov. Conf. Eng. Sci.*, vol. 10, pp. 252–259, 2024, doi: 10.5109/7323271.

[12] E. B. Sevandal, M. F. T. Menque, L. M. Dagsa, A. D. B. Balamad, and E. R. B. Sajonia, "Real-time IoT-based Motorcycle Accident Detection and Alert System," *Int. Exch. Innov. Conf. Eng. Sci.*, vol. 10, pp. 494–499, 2024, doi: 10.5109/7323306.

[13] D. J. B. Felias, D. P. Mondea, J. T. Dellosa, and R. N. Mendoza, "Design and development of an Internet of Things (IoT)-based Air Quality Monitoring for Libjo Mining Corporation," Int. Exch. Innov. Conf. Eng. Sci., vol. 10, pp. 365–370, 2024, doi: 10.5109/7323287.