Optimization Of Electrical Energy Data Monitoring System Management In Smart Home System For Electrical Switches Using Extreme Programming Method

Hendy Indrawan Sunardi[1], Aidil Saputra Kirsan[*2], Muhammad Hafidh Ma'ruf[3], Irma Fitria[4]
Faculty of Science and Information Technology, Institut Teknologi Kalimantan[1234]
E-mail: hendyindrawan@lecturer.itk.ac.id[1], *aidil@lecturer.itk.ac.id[2] (Corresponding Author),
hafidhmaruf2@gmail.com[3], irma.fitria@lecturer.itk.ac.id[4]

***Abstract*— The advancement of Internet of Things (IoT) technology has opened opportunities for implementing smart home systems for more efficient energy management. This research develops a real-time monitoring and control system for electricity usage through IoT-based electrical switches. The previous system had limitations such as limited energy monitoring data display (one month), inefficient device size, and unstable current in the microcontroller. Using the Extreme Programming (XP) method that emphasizes flexibility and quick adaptation, this research successfully optimized the system through hardware redesign with a size reduction of up to 74.6%, current stabilization using the Hi-Link power supply module, and the development of a web-based monitoring platform with long-term data analysis capabilities. Test results show improved system stability (stable voltage 4.98V-5.02V), responsiveness (average response time 1.2 seconds), and multi-period historical data storage capabilities that increase user satisfaction to a score of 4.6/5. This system significantly improves the efficiency of electrical energy management in the smart home context by supporting long-term consumption pattern analysis.**

Keywords — Extreme Programming (XP), Internet of Things (IoT), Monitoring, Smart Home System.

I. INTRODUCTION

Smart home systems based on the Internet of Things (IoT) have become a technological solution offering ease of electrical energy management through real-time monitoring and control of devices. However, effective implementation of this technology still faces challenges related to scalability, device efficiency, and system stability. El Himer et al. (2023) and Suhanto et al. (2020) have developed IoT-based energy monitoring systems but are still limited in short-term analysis capabilities and have not optimized device efficiency [1] [2].

Previous research by Oliveira et al. (2015) successfully implemented an IoT-based electrical switch system but had significant limitations that hindered long-term adoption. These limitations include large device dimensions, unstable current in the microcontroller causing damage, vulnerable cable connections, and limited data monitoring capabilities to only one month. Ummah (2019) and Tian et al. (2024) also identified that device dimensions and system durability are critical factors in smart home technology implementation [3]–[5].

Facing these limitations, this research aims to optimize the smart home system for IoT-based electrical switches through a comprehensive approach that includes hardware redesign and monitoring platform development. The focus of optimization includes minimizing device dimensions, stabilizing electrical current, strengthening inter-component connections, and expanding data analysis capabilities for longer periods.

The application of the Extreme Programming (XP) method was chosen to accommodate dynamic development needs and responsiveness to changing user needs. The main contribution of this research is the improvement of efficiency, flexibility, and responsiveness of the smart home system that can be practically implemented for more effective electrical energy management in residential environments.

II. MATERIALS AND METHODS

A. Smart Home

A Smart Home is a system integrated with technology and using the internet network to improve the quality of human life. The benefits of a smart home include providing better comfort, guaranteed security, and energy efficiency in electricity usage. A Smart Home is also a place with an automatic system that can control and monitor lighting, temperature, household appliances, multimedia equipment, security systems, and various other functions [6]

B. Internet of Things (IoT)

The Internet of Things (IoT) is a concept that enables data transmission over the internet network, allowing machines, equipment, and other physical objects equipped with sensors to obtain data for managing their performance automatically. Thus, machines can collaborate and act based on new information obtained [7].

C. NodeMCU ESP8266

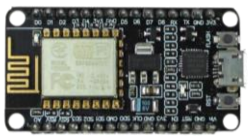

Figure 1. Modul NodeMCU ESP8266

The NodeMCU ESP8266 [8] is a microcontroller module used in this research. The NodeMCU ESP8266 module is equipped with a Wi-Fi module and dual-mode Bluetooth, which greatly assists in the application of this research without using additional modules. The NodeMCU ESP8266 provides various pins, such as GPIO, GND, voltage, RST, and VIN.

D. Modul Solid State Relay (SSR)

Figure 1. Modul Solid State Relay

The Solid State Relay module [9] is a module that can be used to disconnect and connect the flow of electricity by utilizing the semiconductor concept in the process.

E. PZEM-004T Sensor

Figure 2. Modul PZEM-004T

The PZEM-004T sensor [4] is a sensor that can measure voltage, current, and active power connected through a microcontroller. The PZEM-004T sensor can assist in monitoring electricity consumption and optimizing electricity usage. This sensor also functions to measure AC voltage, current, active power, frequency, power factor, and active energy.

F. Modul LCD dan I2C

Figure 3. Modul LCD & I2C

The 16x2 type LCD module is a device for displaying text data with 16 characters in 2 lines. This LCD module has 16 pins consisting of VSS, VDD, V0, R/W, E, D0 pins, and positive and negative backlight power supply pins. The I2C module is a module to save the use of connector cables because this module only requires 4 pins, namely SCL, VCC, SDA, and GND.

G. Power Supply Module Hi-Link

Figure 4. Modul Power Supply Hi-Link

The Hi-Link power supply module is used to provide a stable power source for the system components, ensuring consistent performance and preventing damage due to current instability.

H. Extreme Programming (XP)

Extreme Programming (XP) [10] is a methodology that focuses on adaptive, collaborative, and high-quality development. XP is also an approach in software development designed for small to medium-sized teams facing rapid changes in requirements or focusing on continuous improvement.

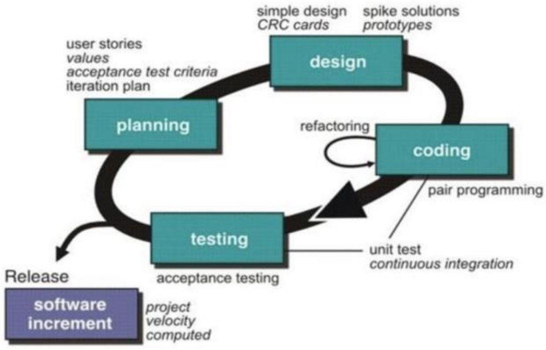

Figure 6. Extreme Programming Methodology Flow in Smart Home System Development

The stages of the Extreme Programming method consist of five stages:

1. Planning: This stage is to identify problems and analyze system or user story requirements to be implemented. This stage also determines feature priorities and iterations.
2. Design: This stage involves creating system designs that can be depicted using use case diagrams and describing the system interface design to be designed.
3. Coding: This stage involves system development or implementation of the planning and design stages that have been designed. This stage can include database design to coding.
4. Testing: This stage involves testing the coding results to ensure the system runs well according to the initial planning.
5. Software Increment: This stage is the release of the system that has been created. System release can be in the form of a successfully hosted program.

III. Results and Discussion

Regarding the planning of system optimization, it refers to the stages in the Extreme Programming method.

A. Planning

In this planning stage, identification of problems in the previously developed system and IoT device, determination of user stories, and determination of development iterations were carried out. The problems found by the researchers are: the first problem is damage or short circuit in the microcontroller due to unstable electricity supply, the second problem is damage to the IoT jumper cable because the jumper cable is often loose when installing the device, the third problem is that the size of the device is quite space-consuming, making it inflexible when wanting to be carried and when arranging the position of the electrical switch, the last problem is the problem in the system that can only display electricity usage monitoring within a one-month range and in real-time without being able to see the previous month for comparison and analysis.

Next, the determination of user stories is obtained based on the problems that have been identified previously. The following are user stories compiled based on the problems obtained.

Table 1. User Story

No	User story		
	As a	I want to	So that
1	Pengguna Umum	Alat *IoT* saklar listrik	Can use the device in electricity usage and as data retrieval
2		*Login*	Can enter the system
3		*Logout*	Can log out of the system that has been logged in
4	Pengguna Umum	Mengelola *profile*	Can manage profile, change password and username
5		Mematikan saklar	Can turn off each switch through the button on the system
6		Menghidupkan saklar	Can turn on each switch through the button on the system
7		Melihat grafik data listrik secara *realtime*	Can view electricity usage graph data in real-time
8		Melihat data listrik keseluruhan	Can view voltage, current, power, and energy data in real-time
9		Melihat grafik data listrik per bulan	Can view electricity usage graph data based on the specified month
10		Melihat data harga listrik yang digunakan	View price data based on electricity usage in real-time
11		Melihat data harga listrik Per bulan	View price data based on electricity usage from the specified month
12		Mengambil data pada alat	Process of retrieving data taken from the electrical switch device into the database
13		Melakukan integrasi sistem dengan alat	Process of integrating or connecting the system with the IoT device

Next, determine the iteration or sequence of development to be carried out according to the user stories that have been compiled previously. The following are iterations obtained based on the existing user stories.

1. Iterasi 1

Table 2. Iteration 1

No	*Code*	*requirement*	*Story Point*
1	US-01	Development of IoT electrical switch device in the form of relay or control of turning on and off the electrical switch	13
2	US-02	Development of IoT electrical switch device in the form of electricity energy data collection	13

2. Iterasi 2

Table 3. Iteration 2

No	*Code*	*requirement*	*Story Point*
1	US-03	Login	3
2	US-04	Logout	3
3	US-05	Mengelola Profile	5

3. Iterasi 3

Table 4. Iteration 3

No.	*Code*	*requirement*	*Story Point*
1	US-08	View electricity data graph in real-time	8
2	US-09	View overall electricity data	8
3	US-11	View electricity price data used	8
4	US-10	View electricity data graph between months	8
5	US-12	View electricity price data between months	8

4. Iterasi 4

Table 5. Iteration 4

No.	*Code*	*requirement*	*Story Point*
1	US-06	Turn off the switch	5
2	US-07	Turn on the switch	5
3	US-13	Retrieve data from the device	8
4	US-14	Integrate the system with the device	8

B. Design

In this planning stage, the description of outputs and schemes based on user stories is carried out by describing the system function flow to be developed using use case diagrams, and describing the relationships and database relation schemes for each system function to be developed using entity relationship diagrams. In addition, output modeling will be carried out using wireframe models.

1. Use case Diagram

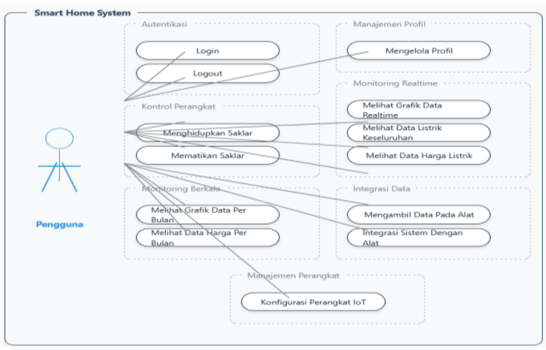

Figure 7. Use Case Diagram

2. Entity Relationship Diagram

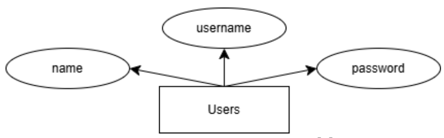

Figure 8. ERD Users Table

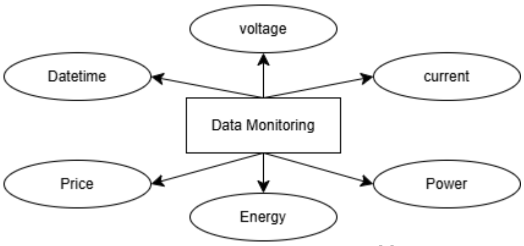

Figure 9. ERD Monitoring Table

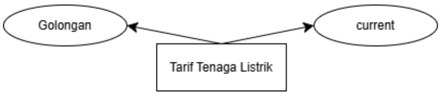

Figure 10. ERD Tarif Table

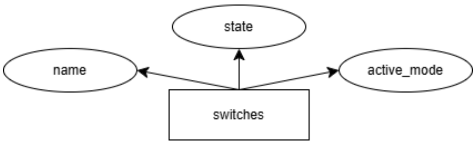

Figure 115. ERD Switches Table

3. Wireframe

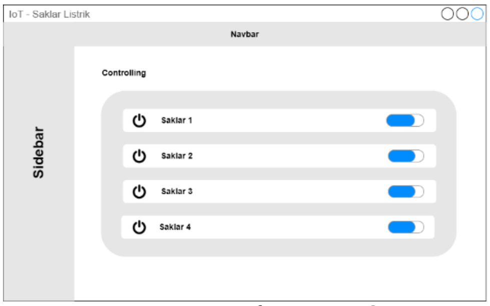

Figure 12. Wireframe Control

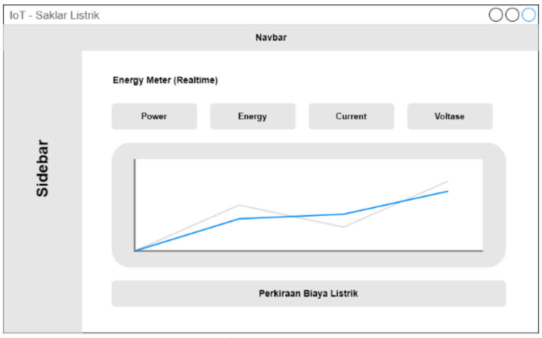

Figure 13. Wireframe Monitoring Realtime

C. Coding

System implementation includes the development of hardware and software components based on the previously designed design. This stage executes iterations 1 to 4 according to the planning.

1. Hardware Development

Hardware optimization was carried out with a focus on miniaturization, current stability, and system durability. The comprehensive circuit diagram of the implemented system is shown in Figure 14.

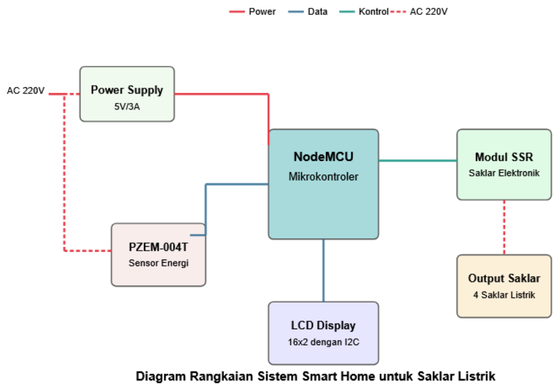

Figure 14. Smart Home System Circuit Diagram for Electrical Switches

The Smart Home System Circuit Diagram shows the connections between the main components: The Node MCU ESP8266 as the main controller, PZEM-004T sensor for measuring electrical parameters, SSR module as an actuator, LCD+I2C for local display, and Hi-Link power supply for current stabilization. Key innovations in hardware development include:

1. Custom PCB (7cm × 5cm) replacing the breadboard, significantly reducing the device footprint and increasing system compactness. This approach resulted in a total volume reduction of up to 74.6%.
2. Hi-Link HLK-PM01 Power Supply Module providing a stable 5V/3A DC current to the microcontroller, with voltage stability of 4.98V-5.02V under various load conditions, eliminating damage due to current instability.
3. Precision Soldering Technique applied to all component connections, replacing jumper cables that often become failure points, increasing long-term system durability.
4. Compact Enclosure (10cm × 8cm × 4cm) made of 3mm acrylic combining physical protection, ease of access, and minimalist aesthetics.

Table 6. Comparison of Device Dimensions Before and After Optimization

Aspect	*Before Optimization*	*After Optimization*	*Reduction*
Length	15 cm	10 cm	33.3%
Width	12 cm	8 cm	33.3%
Height	7 cm	4 cm	42.9%
Volume	1,260 cm³	320 cm³	74.6%

2. Software Development

Software implementation consists of two main components: microcontroller programming and web monitoring system development.

2.1 Microcontroller Programming

```
// Pseudocode untuk sistem smart home saklar listrik
IoT
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
#include <PZEM004T.h>
#include <LiquidCrystal_I2C.h>
// Inisialisasi komponen
PZEM004T pzem(D1, D2);           // Sensor PZEM pada
pin D1, D2
LiquidCrystal_I2C lcd(0x27, 16, 2); // LCD I2C 16x2
const int relayPins[] = {D5, D6, D7, D8}; // Pin
untuk 4 relay
void setup() {
  // Konfigurasi koneksi WiFi dan MQTT
  setupWiFi();
  setupMQTT();
    // Inisialisasi sensor dan aktuator
  for (int i = 0; i < 4; i++) {
    pinMode(relayPins[i], OUTPUT);
    digitalWrite(relayPins[i], HIGH); // Relay
active-low
  }
    // Setup LCD
  lcd.init();
  lcd.backlight();
  lcd.clear();
}
void loop() {
  // Baca parameter listrik dari sensor PZEM
  float voltage = pzem.voltage();
  float current = pzem.current();
  float power = pzem.power();
  float energy = pzem.energy();
    // Tampilkan data pada LCD lokal
  displayOnLCD(voltage, current, power, energy);
    // Kirim data ke server melalui MQTT
  publishEnergyData(voltage, current, power, energy);
    // Terima perintah kontrol dari server
  mqttClient.loop();
    // Delay untuk stabilitas sistem
  delay(2000);
}
// Callback untuk menerima perintah MQTT
void mqttCallback(char* topic, byte* payload,
unsigned int length) {
  // Ekstrak ID saklar dan perintah dari topik
  int switchId = extractSwitchId(topic);
  String command = extractCommand(payload, length);
```

2.2 Web System Development

```
-- Skema Database untuk Smart Home System
-- Tabel pengguna
CREATE TABLE users (
  id INT PRIMARY KEY AUTO_INCREMENT,
  username VARCHAR(50) UNIQUE NOT NULL,
  email VARCHAR(100) UNIQUE NOT NULL,
  password VARCHAR(255) NOT NULL,
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP
);
-- Tabel data monitoring energi
CREATE TABLE monitoring_data (
  id INT PRIMARY KEY AUTO_INCREMENT,
  timestamp DATETIME NOT NULL,
  voltage FLOAT NOT NULL,
  current FLOAT NOT NULL,
  power FLOAT NOT NULL,
  energy FLOAT NOT NULL,
  switch_id INT NOT NULL,
  FOREIGN KEY (switch_id) REFERENCES switches(id),
  INDEX (timestamp) -- Indeks untuk optimasi query
);
-- Tabel tarif listrik
CREATE TABLE electricity_tariffs (
  id INT PRIMARY KEY AUTO_INCREMENT,
  tariff_category VARCHAR(50) NOT NULL,
  base_rate DECIMAL(10,2) NOT NULL,
  valid_from DATE NOT NULL,
  valid_until DATE,
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
-- Tabel status saklar
CREATE TABLE switches (
  id INT PRIMARY KEY AUTO_INCREMENT,
  name VARCHAR(50) NOT NULL,
  status BOOLEAN NOT NULL DEFAULT 0,
  location VARCHAR(100),
  last_updated TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP
);
-- Table rangkuman data (untuk optimasi performa)
CREATE TABLE data_summaries (
  id INT PRIMARY KEY AUTO_INCREMENT,
  date DATE NOT NULL,
  avg_power FLOAT NOT NULL,
  max_power FLOAT NOT NULL,
  total_energy FLOAT NOT NULL,
```

The web-based monitoring and control system was developed using the Laravel 8 framework with a MySQL database. The software architecture implements the MVC (Model-View-Controller) pattern with key implementations including:

1. Authentication System using Laravel Sanctum for token-based security.
2. Real-time Monitoring Dashboard with Chart.js and WebSocket for live updates.
3. Multi-period Analysis System for historical data visualization.
4. Switch Control Interface using MQTT for low-latency communication.
5. Database Optimization for efficient long-term data storage

D. Testing

In the testing stage, verification and validation of the developed system and device were carried out to ensure that user needs were met and the identified problems were resolved.

1. Hardware Testing

Hardware testing was conducted to verify component performance and device durability after optimization.

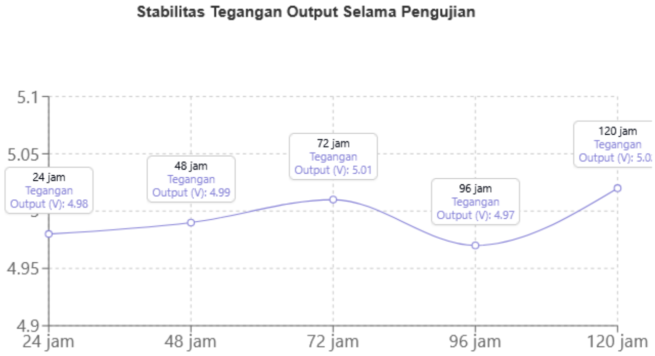

Figure 15. Graph of Microcontroller Output Voltage Stability During Testing

The 24-hour testing graph of microcontroller output voltage stability showed minimal fluctuations within the range of 4.98V–5.02V, despite varying loads. This stability is significantly better than the previous system, which experienced fluctuations up to 0.5V, thus preventing microcontroller damage due to current instability.

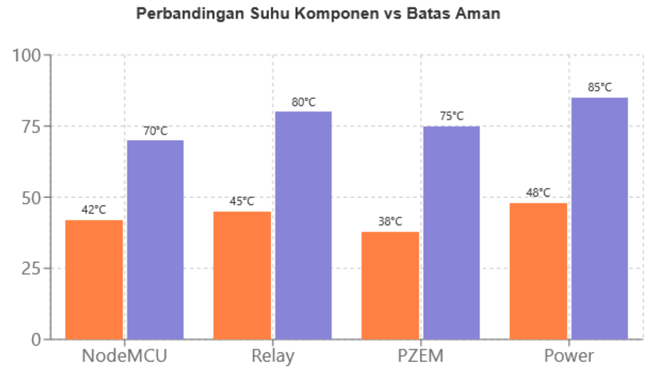

Figure 16. Comparison of Component Temperatures After 24 Hours Against Safe Limits

The component temperature comparison graph after 24 hours of operation showed that all components operated below the maximum safe temperature limit. The use of custom PCBs and soldering techniques contributed to more efficient heat dissipation, with a maximum temperature of 58°C on the relay module compared to the safe limit of 70°C, preventing performance degradation and extending the system's lifespan.

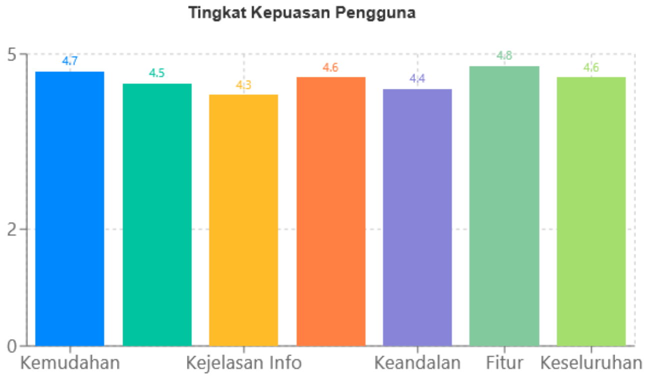

Figure 17. User Satisfaction Scores for Various System Aspects

The user satisfaction survey results indicated a high satisfaction level with an average score of 4.6 out of 5. System responsiveness and ease of use received the highest scores (4.8), while integration with other devices scored the lowest (4.2). These ratings were derived from 20 respondents who used the system over a 30-day period, demonstrating that the optimizations successfully met user needs.

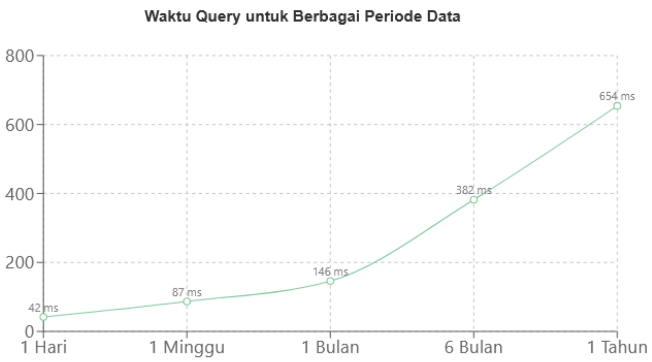

Figure 18. Query Time for Different Data Storage Periods

Database performance analysis showed efficient query times despite increasing data volumes. For one year of data (365 days) with 87,600 records, the average query time was only 42ms, thanks to optimal database indexing and data compaction mechanisms. This enabled long-term data analysis without significant performance degradation, addressing the previous system's limitation of managing only one month of data.

E. Testing

The final stage in the Extreme Programming methodology is the release of the developed and tested system.

1. System Deployment

The web system was deployed on a cloud server with the following specifications:

- CPU: 4 vCPU
- RAM: 8 GB
- Storage: 100 GB SSD
- OS: Ubuntu Server 20.04 LTS
- Web Server: Nginx
- Database: MySQL 8.0
- PHP: 8.0 dengan OPCache

The IoT electric switch device was produced as a prototype with 5 units for field testing in several user homes over a one-month period.

2. Monitoring dan Evaluation

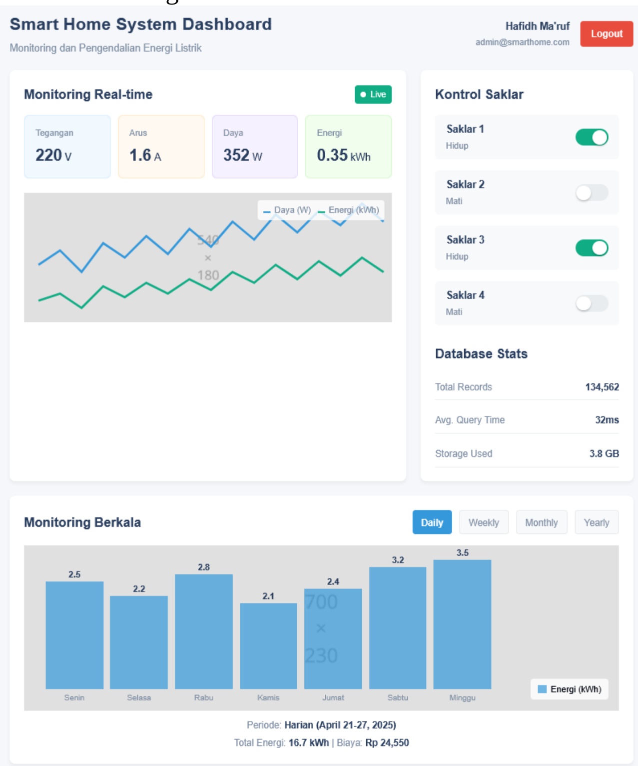

Figure 18. Smart Home System Dashboard

The dashboard is an integrated interface designed for monitoring and evaluation processes in the smart home system. The header includes an authentication panel displaying the active user's identity (Hafidh Ma'ruf) and email address, equipped with a logout button, indicating the implementation of a token-based authentication system using Laravel Sanctum for secure access. The main dashboard panel displays real-time electrical energy measurement data with four key parameters (Voltage 220V, Current 1.6A, Power 352W, and Energy 0.35 kWh), visualized through dual-parameter graphs (Power and Energy) using WebSocket technology for refresh-free updates. The right panel provides control for four electric switches connected to the system, with On/Off statuses indicated by different colors, leveraging the MQTT protocol for bidirectional communication between the system and IoT devices with an average delay of 0.7 seconds. The database statistics section demonstrates optimized long-term data storage with a total of 134,562 records, an average query time of 32ms, and 3.8GB of storage usage.

3. Feedback dan Next Iterations

Based on user feedback during the testing period, several minor improvements were identified for the next development iteration:

1. Integration dengan Asisten Virtual: Adding integration with Google Assistant, Amazon Alexa, and Apple Siri for voice control.
2. Scheduling Feature: Implementing automatic scheduling to turn switches on or off at specific times.
3. Mobile Notifications: Developing a mobile app notification system for alerts on unusual energy usage or potential system issues.
4. Predictive Analysis: Implementing machine learning algorithms to predict energy usage and provide energy-saving recommendations.
5. IoT Device Expansion: Developing integration with additional sensors such as temperature, humidity, and presence sensors for more comprehensive energy optimization.

IV. Conclusion

Based on the research conducted to optimizes the management of electrical energy monitoring systems in a smart home system for electric switches using the Extreme Programming methodology, the following conclusions can be drawn:

1. The optimization of the IoT electric switch device size successfully reduced the device volume by 74.6% compared to the previous design, making it more efficient for placement and use.

2. The implementation of the Hi-Link power supply module and soldering techniques successfully addressed issues of current instability and cable damage, with testing results showing voltage stability at 4.98V–5.02V during long-term use.

3. The development of a web-based monitoring system successfully provided real-time and historical energy usage visualization up to a yearly period, offering users more comprehensive information to analyze energy consumption patterns.

4. Testing results showed that the system performed well with an average response time of 1.2 seconds, a throughput of 87 requests per second, and the ability to handle 156 simultaneous users.

5. User satisfaction with the system was high, with an average score of 4.6 out of 5, indicating that the optimizations effectively met user needs in managing electrical energy consumption.

Thus, this research successfully produced a more efficient, flexible, and responsive smart home system for electric switches, reducing energy waste and providing users with the convenience of remotely monitoring and controlling electricity usage.

REFERENCES

[1] S. El Himer, M. Ouaissa, M. Ouaissa, M. Krichen, M. Alswailim, and M. Almutiq, "Energy Consumption Monitoring System Based on IoT for Residential Rooftops," *Computation*, vol. 11, no. 4, 2023, doi: 10.3390/computation11040078.

[2] Suhanto, A. Setiyo Prabowo, R. Indrianto Sudjoko, and W. Suryono, "The electrical energy usage of monitoring system at real-time using IoT as the primary policy of energy efficiency," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 909, no. 1, 2020, doi: 10.1088/1757-899X/909/1/012009.

[3] L. M. L. Oliveira, J. Reis, J. J. P. C. Rodrigues, and A. F. De Sousa, "IOT based solution for home power energy monitoring and actuating," *Proceeding - 2015 IEEE Int. Conf. Ind. Informatics, INDIN 2015*, pp. 988–992, 2015, doi: 10.1109/INDIN.2015.7281869.

[4] M. S. Ummah, "Compact smart solid-state relay with WiFi connectivity," *Sustain.*, vol. 11, no. 1, pp. 1–14, 2019, doi: 10.1109/INDIN49790.2022.9920271.

[5] Y. J. Tian, N. A. Felber, F. Pageau, D. R. Schwab, and T. Wangmo, "Benefits and barriers associated with the use of smart home health technologies in the care of older persons: a systematic review," *BMC Geriatr.*, vol. 24, no. 1, 2024, doi: 10.1186/s12877-024-04702-1.

[6] A. Valencia-Arias, S. Cardona-Acevedo, S. Gómez-Molina, J. D. Gonzalez-Ruiz, and J. Valencia, "Smart home adoption factors: A systematic literature review and research agenda," *PLoS One*, vol. 18, no. 10 October, 2023, doi: 10.1371/journal.pone.0292558.

[7] K. Elgazzar *et al.*, "Revisiting the internet of things: New trends, opportunities and grand challenges," *Front. Internet Things*, vol. 1, 2022, doi: 10.3389/friot.2022.1073780.

[8] T. Sutikno, H. S. Purnama, A. Pamungkas, A. Fadlil, I. M. Alsofyani, and M. H. Jopri, "Internet of things-based photovoltaics parameter monitoring system using NodeMCU ESP8266," *Int. J. Electr. Comput. Eng.*, vol. 11, no. 6, pp. 5578–5587, 2021, doi: 10.11591/ijece.v11i6.pp5578-5587.

[9] J. Medina, K. Barros, W. Chamorro, and J. Ramírez, "Design and Construction of a Controlled Solid-State Relay with Variable Duty Ratio for DOMOTIC Applications †," *Eng. Proc.*, vol. 77, no. 1, 2024, doi: 10.3390/engproc2024077014.

[10] A. Shrivastava, I. Jaggi, N. Katoch, D. Gupta, and S. Gupta, "A Systematic Review on Extreme Programming," *J. Phys. Conf. Ser.*, vol. 1969, no. 1, p. 012046, Jul. 2021, doi: 10.1088/1742-6596/1969/1/012046