Journal of Advanced Vocational Information and Communication Technology
Year: 2026, Volume: 1, Number: 1, Pages: 38-52

Journal of Advanced Vocational Information and Communication Technology

Homepage: https://journal.isas.or.id/index.php/javict, email: javict@isas.or.id

Smart Home Implementation with Home Assistant Platform in Modern Housing

Alan Tri Arbani Hidayat [1], M. Udin Harun Al Rasyid [1,], Isbat Uzzin Nadhori [1], Yeremia Ega Wahyudi [1], Yoga Prastyo Bayu Laksono [1]*

[1,*] Department of Informatics and Computer Engineering, Politeknik Elektronika Negeri Surabaya, Surabaya, Indonesia

*Corresponding Author: udinharun@pens.ac.id

Article Info

Article History:
Submitted 22 Dec 2025
First Revised 21 Jan 2026
Accepted 23 Jan 2026
Available Online 25 Jan 2026
Publication Date 30 Jan 2026

Keyword:
Smart Home;
Home Assistant,
IoT;
ESP32;
Firebase;
Node-RED.

Abstract

The implementation of smart home systems using the Home Assistant platform in modern residential environment addresses residents' needs for practical, effective, and energy-efficient home management. Although the smart home market is projected to grow by 16.3 until 2027, its adoption in Indonesia still faces challenges related to public understanding and device integration complexity. The developed system integrates temperature and humidity sensors, energy consumption measurement sensors, motion defection sensors, relays for device control, and an esp32 as the central control unit. Data from esp32 is transmitted in real-time to Firebase Real Time Database, then integrated with Home Assistant and Node-RED and add-ons for securing monitoring. Evaluation demonstrates effective performance in integrating devices from various vendors and protocols, as well as responsiveness of automation features. This development provides an efficient, cost-effective, and ease-adopted smart home implementation model, demonstrating how the open-source Home Assistant platform can enhance energy efficiency, comfort, and security for residents of modern housing in Indonesia.

1. Introduction

The rapid advancement of information and communication technology (ICT) has accelerated the adoption of Internet of Things (IoT)-based solutions across various aspects of daily life, including within household environments [1]. One of the most prominent applications of IoT is the smart home system, which enables centralized, automated control of electronic devices to enhance energy efficiency, comfort, and

DOI: http://dx.doi.org/10.xxxxx/javict.vXiX.paperID
p-ISSN: xxxx-xxxx | e-ISSN: xxxx-xxxx

security in residential settings [2]. According to a 2023 report by Statista, the global smart home market is projected to grow at a compound annual growth rate of 16.3% through 2027, driven by urbanization, rising energy concerns, and increased accesibility of smart technologies [3]. Despite its growing popularity, smart home implementation-particularly in developing countries like Indonesia-faces several challenges. A study by Rahman et al. Revealed that only 23% of urban Indonesian respondents fully understand the concept and benefits of smar thomes. Other major obstacle include the complexity of integrating devices form various manufacturers using different communication protocols, reliance on stable internet connectivity, the high cost of proprietary systems, and lack of intuitive user interfaces

To address these challenges, this research explores the use of Home Assistant, an open-source smart home platform that emphasizes local control, privacy, and interoperability [4]. Home Assistant supports integration with a wide range of IoT devices from different vendors and protocols, allowing users to build customized automation scenarios. Furthermore, its add-on ecosystem enables flexible and extensible integration with third- party services, offering an adaptable approach for modern smart home deployment.

However, the implementation of such a system introduces specific technical challenges that this research aims to address. Firstly, the integration of heterogeneous devices, such as lights, electronic appliances, and sensors, demands a deep understanding of the distinct communication methods and data structures employed by each component. Secondly, while platforms like Home Assistant simplify this process, the reliance on its add-on ecosystem for integrating third-party services necessitates careful evaluation to ensure the compatibility, stability, and long-term reliability of these components within the overall system. Therefore, this study focuses on navigating these integration complexities to build a robust and stable smart home prototype.

This study aims to investigate how Home Assistant can be effectively implemented to build accessible, cost-efficient, and user-friendly smart home systems in modern Indonesian residential environments. Specifically, the research objectives are: (1) to design an integrated smart home system using Home Assistant; (2) to analyze system performance in managing various IoT devices; and (3) to evaluate overall system stability and compatibility.

To provide a solid foundation for this research, several prior studies relevant to smart monitoring and smart home integration using the Home Assistant platform are reviewed. These works explore diverse implementations ranging from environmental monitoring and surveillance to accessibility and security, highlighting the platform's versatility and potential.

Suvar et al [5] addresses the challenge of monitoring server rooms, which are typically unmanned due to high noise levels and specific climate conditions. These environments necessitate a specialized monitoring system with robust hardware and software components, requiring remote communication capabilities within a private network, high speed responsiveness, and straightforward event identification.

To meet these requirements, the Home Assistant platform was selected for its significant advantages, including ease of customization and configuration without the need for specialized training. The Node-RED plugin was utilized as the primary tool to implement several critical automation routines. Key automations include: dispatching a smartphone alert when the temperature exceeds 26°C; logging a power failure of the primary air conditioning (AC) unit and notifying three designated smartphones; activating an auxiliary AC unit when the temperature rises above 21°C to provide supplementary cooling; and triggering an alarm status in the event of an AC malfunction.

The study concludes that the implementation of this automated monitoring system using Home Assistant significantly enhances the efficiency and effectiveness of facility oversight. By providing comprehensive integration and granular control over smart devices, the system improves both the security and operational management of the server room environment.

Akhmetzhanov et al [6] proposed the design of a cost- effective home security system that integrates multiple CCTV cameras with the Home Assistant platform. The system architecture utilizes a Raspberry Pi microcontroller running MotionEyeOS, which functions as a central hub to manage and aggregate feeds from three individual cameras. This configuration enables the consolidated video streams to be seamlessly integrated into the Home Assistant dashboard by targeting the IP address or URL of the MotionEyeOS instance.

The primary contribution of this approach is its economic advantage over proprietary, third-party surveillance systems. By leveraging the Raspberry Pi and the open-source Home Assistant platform, which requires no additional licensing costs, this design offers an accessible and more easily implemented alternative for advanced home security monitoring.

Munteanu et al [7] proposed the design and implementation of a residential security system centered on a custom-designed door locking module that utilizes Radio Frequency Identification (RFID) for access control. A primary achievement of the study was the successful integration of this bespoke hardware with the Home Assistant platform. Within this integrated system, the principal function of Home Assistant was to serve as a monitoring and notification engine, providing real- time alerts for "door open" and "door close" events.

Rocha et al [8] proposed a solution to address the increasing complexity of technology, which often hinders the accessibility and benefits of smart home systems. They contended that a Home Assistant-based system could enhance this accessibility by providing an integrated, natural, and multimodal method of interaction with the home ecosystem. To validate this, the "Smart Green Homes" project was developed, creating an integrated home assistant built on an information and communication technology (ICT) infrastructure. This system was deployed as a functional demonstrator and evaluated by seventy users.

The results of the user evaluation were generally positive, with 61% of participants rating the assistant as "good" or "very good," and 51% indicating they would be likely to recommend it to others. These findings suggest that a home assistant offering an integrated view of the smart home through natural, multimodal, and adaptive interactions presents a viable solution for improving smart home accessibility and contributing to a better living environment for its occupants.

The review of related works indicates that Home Assistant is a robust and versatile platform for a wide range of smart home applications. Previous studies have successfully demonstrated its capacity for mission-critical automation, its integration with custom-designed hardware such as RFID access control modules, and its ability to serve as a low-cost hub for security systems using components like Raspberry Pi and MotionEyeOS. Furthermore, research has validated its potential for creating accessible and user-friendly interfaces that receive positive reception from end-users.

The review of related works indicates that Home Assistant is a robust and versatile platform for a wide range of smart home applications. Previous studies have successfully demonstrated its capacity for mission-critical automation, its integration with custom-designed hardware, and its ability to serve as a low-cost hub for security systems.

DOI: http://dx.doi.org/10.xxxxx/javict.vXiX.paperID
p-ISSN: xxxx-xxxx | e-ISSN: xxxx-xxxx

Furthermore, research has validated its potential for creating accessible and user-friendly interfaces that receive positive reception from end-users.

However, while these studies validate the platform's capabilities in specific domains, a gap persists in research that documents a practical, end-to-end implementation of a multi- domain smart home system. Much of the existing literature focuses on either a single application (e.g., security or energy management) or high-level user interaction, without detailing the process of integrating a wide spectrum of heterogeneous sensors and actuators into a single, unified control dashboard.

Therefore, this research aims to address this gap by presenting a replicable case study of a comprehensive smart home implementation. By detailing the integration of functionalities spanning security (motion detection and CCTV), environmental control (temperature and humidity), and granular energy management, this work serves as a practical blueprint. The objective is to demonstrate how disparate components can be effectively unified into a cohesive and manageable smart home ecosystem, thus bridging the gap between domain-specific solutions and a truly integrated management model.

2. Methods

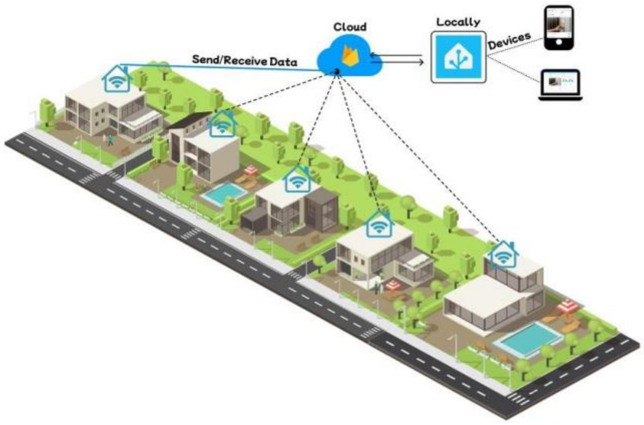


Figure 1. Integration of home assistant with firebase

Figure 1 illustrates a smart home network comprising five individual houses, each equipped with installed modules that are interconnected within a unified ecosystem. Each house is outfitted with various smart devices, enabling communication with a centralized Firebase server.

Residents utilize the Home Assistant platform to conveniently monitor and control home devices such as lighting. The system also provides access to essential environmental data, including electricity consumption, room temperature, humidity levels, and motion activity.

To enhance security, the system supports local integration features, CCTV cameras are implemented for real-time surveillance, and two-way voice communication allows interaction with individuals near the premises.

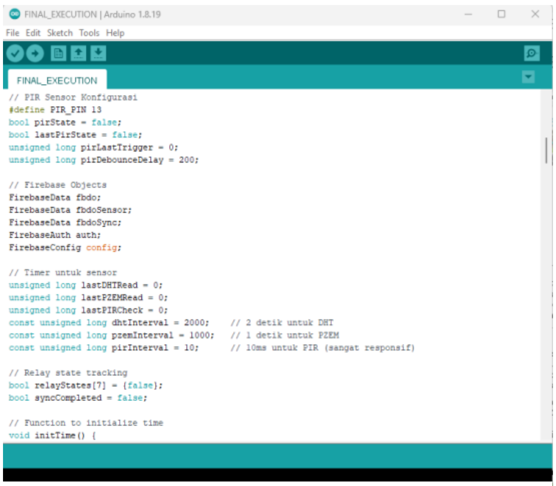
Through this integrated platform, residents are able to manage their smart home environment efficiently and securely via a single, user-friendly interfac

Figure 2. Sensor location design

Figure 2 illustrates a smart home network comprising five individual houses, each equipped with installed modules that are interconnected within a unified ecosystem. Each house is outfitted with various smart devices, enabling communication with a centralized Firebase server.

The lighting system, controlled via relays, comprises five LED lamps, distributed across key areas. A lamp is positioned in the living room, the primary space for occupant interaction, and another in the kitchen to support culinary activities. The remaining lamps are installed in the child's bedroom, the parent's bedroom, and the bathroom to provide dedicated illumination for these specific zones.

For environmental monitoring, the DHT11 temperature and humidity sensor is strategically placed in the living room, as this area serves as a representative benchmark for the overall ambient comfort level within the house. A PIR sensor is positioned near the main entrance to detect ingress and egress functioning as a trigger for both automated lighting and security notifications. While the PZEM-004T sensor measures the comprehensive electrical parameters of the home, its physical location is not depicted on the floor plan as it is integrated directly into the main electrical distribution panel. Finally, a TP-Link Tapo C200CCTV camera is installed at the front of the house to provide real-time visual surveillance of the entrance area, thereby enhancing overall security.




```
// PIR Sensor Konfigurasi
#define PIR_PIN 13
bool pirState = false;
bool lastPirState = false;
unsigned long pirLastTrigger = 0;
unsigned long pirDebounceDelay = 200;

// Firebase Objects
FirebaseData fbdo;
FirebaseData fbdoSensor;
FirebaseData fbdoSync;
FirebaseAuth auth;
FirebaseConfig config;

// Timer untuk sensor
unsigned long lastDHTRead = 0;
unsigned long lastPZEMRead = 0;
unsigned long lastPIRCheck = 0;
const unsigned long dhtInterval = 2000;    // 2 detik untuk DHT
const unsigned long pzemInterval = 1000;   // 1 detik untuk PZEM
const unsigned long pirInterval = 10;      // 10ms untuk PIR (sangat responsif)

// Relay state tracking
bool relayStates[7] = {false};
bool syncCompleted = false;

// Function to initialize time
void initTime() {
```

Figure 3. Ardunio IDE

DOI: http://dx.doi.org/10.xxxxx/javict.vXiX.paperID
p-ISSN: xxxx-xxxx | e-ISSN: xxxx-xxxx

The programming and configuration of the ESP32 microcontroller were conducted using the Arduino Integrated Development Environment (IDE), as depicted in Figure 3. This environment was utilized to write the C++ based code responsible for initializing the connected sensors and actuators. The developed firmware defines the logic for reading data from sensors, processing these inputs, and controlling the relays to manage devices such as lights. Finally, the Arduino IDE was used to compile the source code and upload the resulting firmware to the ESP32, making the system operational.

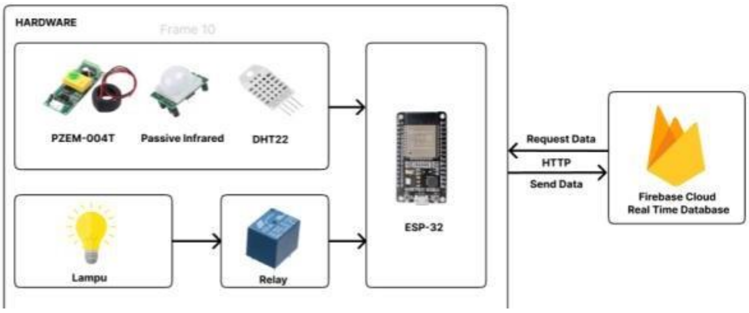


Figure 4. Hardware circuit system design

To connect the sensors and actuators to the microcontroller the Arduino IDE on Figure 4 programming environment is used. The data collected by the ESP32 microcontroller is transmitted in real-time to Firebase Realtime Database. Through Firebase, users can manage and modify logic values to control devices, such as turning lights on or off.

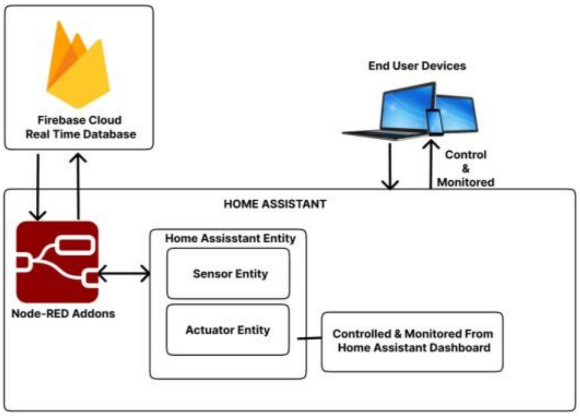


Figure 5. Home assistant with Node-RED

The system architecture leverages Node-RED as middleware to establish a robust data pipeline between the Firebase Realtime Database and the Home Assistant front-end. A specific flow was developed in Node-RED to continuously listen for data updates within a designated path in Firebase. When new sensor data is pushed to the database, the Node-RED flow is triggered, retrieving the data payload. This data is then formatted and channeled to update specific entities within Home Assistant, enabling real-time visualization on the user dashboard. This integration method simplifies the process of monitoring various sensor readings in a centralized and intuitive manner, as illustrated in the system design in Figure 5.

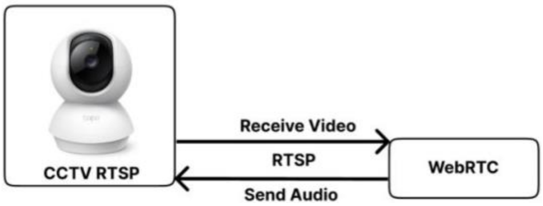


Figure 6. CCTV integration using the real-time streaming protocol (RTSP)

Furthermore, the system in Figure 6 includes CCTV integration using the Real-Time Streaming Protocol (RTSP) to support real-time video monitoring for enhanced home security. RTSP-based CCTV cameras are connected to Home Assistant, enabling residents to view live video feed through the platform. These cameras support two-way communication, allowing users not only to see but also to speak with individuals near their home via built-in microphones and speakers

This functionality is facilitated by the WebRTC add-on in Home Assistant, which serves as a bridge to manage video streams from various camera sources, including RTSP-compatible devices. The WebRTC plugin enables low-latency video streaming and supports both audio transmission and reception. Through this integration, residents can send voice messages and engage in real-time communication directly through the CCTV system, significantly improving remote interaction and overall security responsiveness.

2.1. Firebase Data

Table 1. Firebase Data

	DATA	TYPES
DHT	Humidity	**°C (*celcius*)**
	Temperature	**%**
electricity	Current	**V (*Voltage)***
	energy	**W (*Watt)***
	Power	**A (*Ampere*)**
	Voltage	**Wh (Watt Hours)**
Motion	True/false	**Boolean**
relay	Relay1	**True/False**
	Relay2	
	Relay3	
	Relay4	
	Relay5	

Table 1 delineates the data structure implemented within the Firebase Realtime Database for this research. The hierarchical, JSON-based structure is designed to logically organize all sensor and actuator data, facilitating efficient real-time retrieval and management by the Node-RED middleware. The data model consists of four primary root nodes: DHT, electricity, Motion, and relay.

The DHT root node contains two child nodes for environmental data: Temperature, which stores a numerical value representing the ambient temperature in degrees Celsius (°C), and Humidity, which stores a numerical value for the relative humidity as a percentage (%). All power-related metrics are grouped under the electricity parent node. This node includes four children: Voltage (in Volts, V), Current (in Amperes,

DOI: http://dx.doi.org/10.xxxxx/javict.vXiX.paperID
p-ISSN: xxxx-xxxx | e-ISSN: xxxx-xxxx

A), Power (in Watts, W), and energy (in Watt-hours, Wh). This structured approach allows for the simultaneous monitoring of all critical electrical parameters.

For state-based data, the Motion node holds a Boolean value (true or false) to indicate the presence or absence of detected movement. Finally, the control states of the actuators are managed under the relay node. This parent node contains child nodes for each individual relay (e.g., Relay1, Relay2), with each storing a Boolean value to represent its on (true) or off (false) state. This design is inherently scalable to accommodate additional actuators as needed and serves as the single source of truth for the system's integration flows.

3. Results and Discussion

3.1. Firebase Integration with Home Assistant

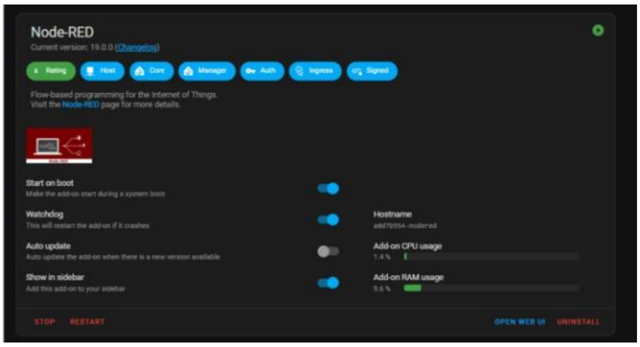


Figure 7. Integration of home assistant with firebase

Figure 7 For Sensor data stored in Firebase is integrated into Home Assistant as entities using a custom component method. This integration enables users to monitor sensor data and control relay statuses directly through the Home Assistant interface. The following steps were performed to achieve this integration:

First, the Node-RED add-on was installed in Home Assistant. This add-on provides a visual, flow-based development environment that allows users to create automation workflows and logic without extensive coding. It enables seamless communication between Home Assistant and external services, including Firebase, through configurable nodes and event-driven programming. Using Node-RED, the author was able to define flows that retrieve sensor values from Firebase and assign them to specific entities within Home Assistant.

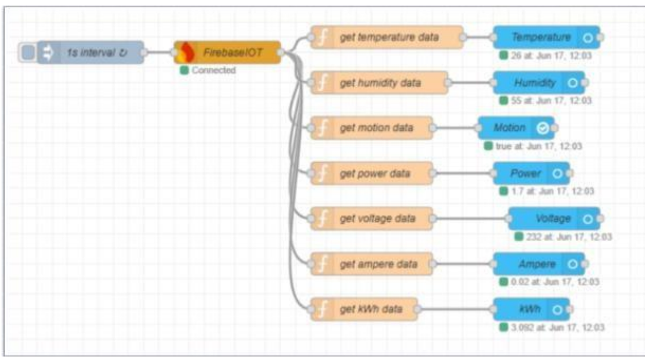


Figure 8. Get sensor data into home assistant entities

To facilitate real-time data monitoring in Home Assistant, a flow was developed using the Node-RED add-on, as shown in Figure 8. The process begins with a trigger that activates every second, prompting the system to retrieve the latest sensor data from the Firebase Realtime Database.

The data payload retrieved by this flow comprises a diverse set of readings, including ambient temperature, binary motion detection status, and a comprehensive suite of electrical parameters. Within the Node-RED environment, this raw data undergoes a crucial parsing and routing stage. The flow is designed to separate each measurement from the main data object and channel it to a dedicated Home Assistant entity. For instance, extracted temperature values are formatted and pushed to a specific temperature sensor entity, ensuring they are correctly rendered on the user interface in degrees Celsius (°C).Similarly, motion data updates a binary sensor to reflect an 'active' or 'inactive' state. The system also provides a granular view of electrical usage by individually updating entities for power in watts (W), voltage in volts (V), current in amperes (A), and cumulative energy consumption in kilowatt-hours (kWh).

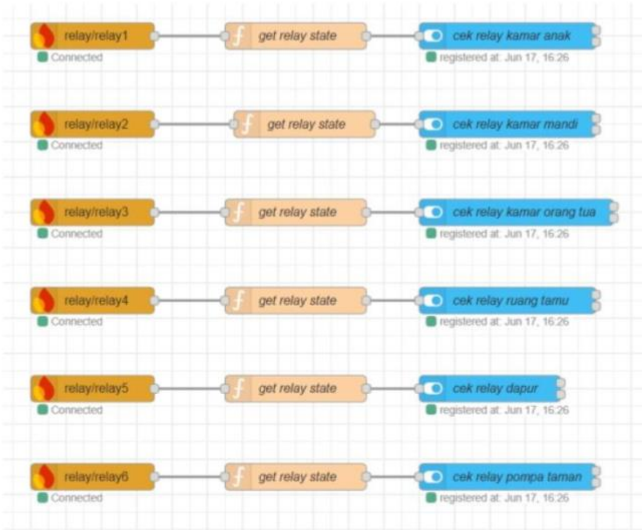


Figure 9. Real-time relay control using Node-RED

Figure 9 In addition to sensor monitoring, the system also supports real-time relay control using a Node-RED flow integrated with Home Assistant. Firebase input nodes are configured to track changes for relay1 through relay5. When a change is detected, a function node processes the data to ensure it is a valid boolean value. If valid, the corresponding relay state is updated in Home Assistant as a switch entity. A debug node displays the relay status in Node-RED for monitoring purposes.

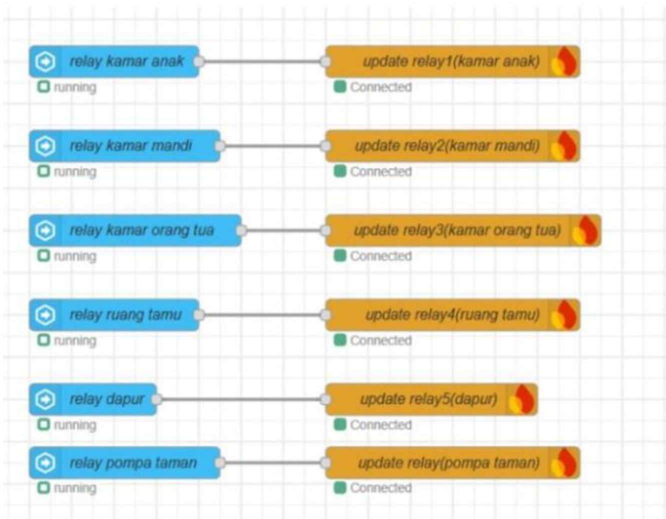


Figure 10. Node-RED flow updates the status of the relays

DOI: http://dx.doi.org/10.xxxxx/javict.vXiX.paperID
p-ISSN: xxxx-xxxx | e-ISSN: xxxx-xxxx

This setup allows reliable and responsive control of connected devices through Home Assistant, based on real-time data from Firebase. As shown in Figure 10 the Node-RED flow updates the status of the five relays based on changes in Home Assistant, which are then reflected in Firebase. The process starts with a node that monitors the status of each relay. When a change is detected, the new data are processed and formatted correctly. It is then sent to another node, which updates the relay status in Firebase. This approach ensures that whenever the status of the relay changes in Home Assistant, the information in Firebase is automatically updated, keeping it accurate and allowing users to easily monitor and control the relays.

3.2. Connecting to the Surveillance System

Home Assistant can be integrated with TP-LINK Tapo C200 cameras using the RTSP protocol and WebRTC add-ons for more responsive and high-quality video playback. The following steps outline the integration process:

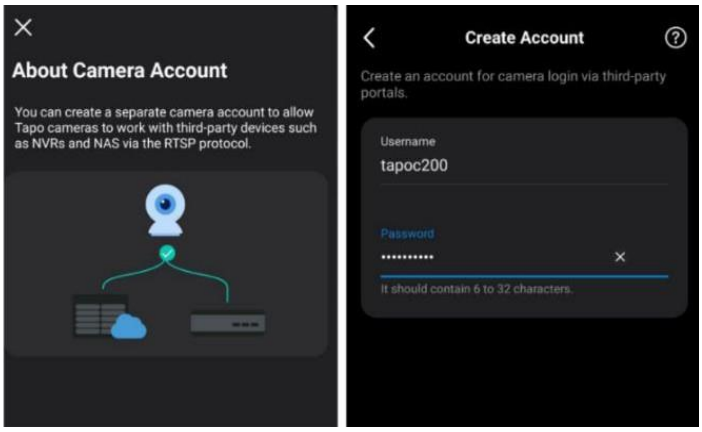


Figure 11. CCTV RTSP configuration

As shown in Figure 11 the first step is to activate RTSP on the Tapo C200 camera. Open the Tapo aplication on your smartphone, select the CCTV camera, access the camera settings, locate the "Advanced Settings" option, and configure the camera account by establishing a username and password for the camera

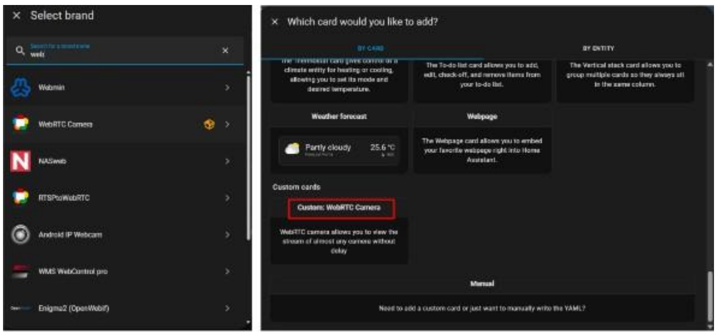


Figure 12. Home assistant WebRTC widget

Figure 12 illustrates the process of opening the Home Assistant dashboard and navigating to Settings - Device and Integrations. Search for and install "WebRTC." After installation, ensure WebRTC can run without issues. Then, access the Home Assistant Main Dashboard, search for and add the "WebRTC" widget to implement the integration.

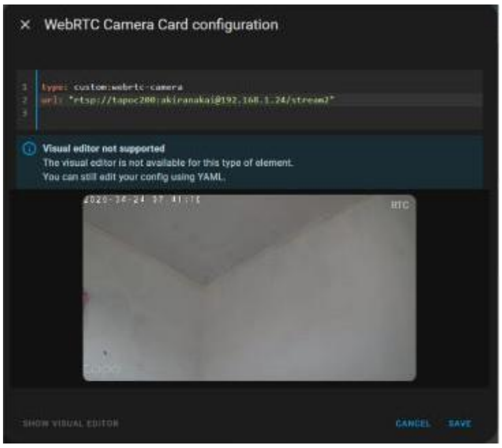


Figure 13. CCTV configuration to home assistant

Figure 13 shows the configuration process after the WebRTC add-on has been successfully installed and added. Configure the camera widget by entering the requested URL information in the format rtsp://username@IP/stream1. Once the CCTV camera appears, click save to add the CCTV widget to the Home Assistant dashboard. You can now view video from the Tapo C200 camera directly on the Home Assistant dashboard.

3.3. Home Monitoring with Home Assistant

Figure 14. Home assistant final dashboard

As the final result of this smart home system implementation, according to Figure 14 the dashboard serves as a centralized interface designed to efficiently manage and monitor various aspects of the home. A primary feature is the control panel for managing luminaires in multiple areas, including the living room, bathroom, master bedroom, children's bedroom, and kitchen. This allows users to directly control lighting, which provides convenience while promoting energy efficiency. From a security perspective, the system is equipped with a PIR sensor near the door. Security is further enhanced through the integration of a live video feed from a TP-Link Tapo C200 camera positioned at the front of the house, enabling real-time visual monitoring.

For environmental monitoring, the dashboard displays real- time data on the living room's temperature and humidity. This information is vital for maintaining thermal comfort and preventing excessively dry or humid atmospheric conditions. Furthermore, a comprehensive electrical monitoring panel is integrated, presenting critical data such as voltage (V) to assess supply stability, current (A) to detect load spikes, active power (W) reflecting instantaneous consumption, and cumulative energy consumption (kWh) to aid in managing electricity bills.

To provide additional context, the dashboard is augmented with real-time weather information sourced from an online service, including outdoor temperature, humidity,

DOI: http://dx.doi.org/10.xxxxx/javict.vXiX.paperID
p-ISSN: xxxx-xxxx | e-ISSN: xxxx-xxxx

and daily forecasts to assist with activity planning. As a complement, a history panel meticulously logs every status change of the light switches, complete with timestamps. This activity log serves as a valuable reference for reviewing energy usage patterns or for developing more precise automation routines. Overall, this dashboard configuration provides an integrated and user- friendly digital platform for efficiently managing and monitoring daily home activities.

3.4. Home Assistant System Testing

Following system implementation, functional testing was conducted to validate the performance of the developed system. The validation process involved systematically testing the functionalities available on the Home Assistant dashboard. Each control icon on the dashboard was interacted with to confirm that the corresponding physical device, such as a lamp, activated correctly. Additionally, the sensor data displayed on the interface was verified to ensure accurate real-time readings.

Table 2. System Testing Scenario

No	Test Scenario	Expected Result	Conclusion
1	The living room light is activated from the Home Assistant dashboard	Relay activates; lamp turns on.	**Success**
2	The child's bedroom light is activated from the Home Assistant dashboard.	Relay activates; lamp turns on.	**Success**
3	The bathroom light is activated from the Home Assistant dashboard.	Relay activates; lamp turns on.	**Success**
4	The parent's bedroom light is activated from the Home Assistant dashboard.	Relay activates; lamp turns on.	**Success**
5	The kitchen light is activated from the Home Assistant dashboard.	Relay activates; lamp turns on.	**Success**
6	Motion is detected by the PIR sensor at the main entrance.	Dashboard showing active motion	**Success**
7	The temperature sensor reading exceeds the pre-set threshold (23 °C).	Dashboard showing expected temperature	**Success**
8	The humidity sensor measures the ambient humidity in the living room.	Real-time humidity value displayed.	**Success**
9	Electrical parameters are monitored while a load (e.g., CCTV) is active.	Accurate electrical data (V, A, W, kWh) displayed.	**Success**
10	The weather monitoring feature is accessed on the dashboard.	Weather data is displayed correctly.	**Success**
11	The activity log for light switches is inspected on the dashboard.	Log shows accurate State changes and timestamps.	**Success**
12.	The live feed from the Tapo C200 camera is accessed via Home Assistant.	Low-latency video stream displayed	**Success**

The results presented in Table 2 demonstrate that all tested functionalities, from actuator control and real-time sensor monitoring to third-party service integrations, performed successfully. This comprehensive validation confirms the stability and reliability of the implemented smart home system in meeting its operational requirements.

This study reinforces the findings from previous literature by successfully demonstrating Home Assistant's capacity for comprehensive, multi-domain integration, spanning environmental monitoring, energy management, and security, within a unified dashboard, an aspect only partially explored in prior, more domain-specific works. Furthermore, it validates and extends earlier research on cost-effective integration and user accessibility by providing a practical, end-to-end blueprint that combines custom ESP32-based hardware with third-party services like Firebase and RTSP cameras into a cohesive and manageable ecosystem.

4. Conclusion

This paper has detailed the design and implementation of a comprehensive smart home system developed on the open- source Home Assistant platform. The primary objective was to create a unified and accessible interface for managing and monitoring a diverse range of household devices to improve residential comfort, security, and energy efficiency. The evaluation confirms that the system successfully meets its primary objectives, demonstrating robust communication between the IoT device layer (ESP32), the Firebase Realtime Database, Node-RED middleware, and the Home Assistant user interface.

The system demonstrated reliable real-time data acquisition and visualization. The DHT11 sensor provided consistent temperature and humidity readings, while the PIR sensor accurately detected motion at the main entrance. Furthermore, the PZEM-004T sensor delivered precise measurements of the home's voltage, current, power, and cumulative energy consumption. Testing of the actuator module confirmed that all relay-controlled lamps could be reliably operated from the Home Assistant interface without error. The integration architecture proved to be highly effective, with data successfully synchronized bidirectionally between the devices and the

In conclusion, the developed system has fulfilled the core smart home functionalities of real-time monitoring and remote control. While the system operates as designed, several avenues for future enhancement have been identified. Further development could focus on refining the user interface (UI) to be more informative and user-friendly. Additionally, a critical area for future work is the implementation of a robust security infrastructure to enable secure remote access to the Home Assistant instance from public networks. This would involve configuring secure protocols, such as HTTPS via a reverse proxy, and enforcing strong authentication measures to protect the system from external threats.

5. Acknowledgment

The author would like to express sincere gratitude the members of the Mobile Sensing and Edge Computing Technology (MSECT) for their stimulating discussions and to the technical staff of the MSECT Laboratory at Politeknik Elektronika Negeri Surabaya (PENS) for their assistance with the experimental setup.

6. Author's Note

The authors hereby declare that there is no conflict of interest related to the publication of this article. Furthermore, the authors confirm that the manuscript is original and free from any form of plagiarism.

7. References

[1] R. Ruuhwan, R. Rizal, and I. Karyana, "Sistem Kendali dan Monitoring Pada Rumah Pintar Berbasis Internet of Things (IoT)," Innovation in Research of Informatics (INNOVATICS), vol. 1, no. 2, pp. 43–50, Oct. 2019, doi: 10.37058/innovatics.v1i2.877.

[2] A. E. Okterina, "Pengaruh Gaya Hidup Modern dan Persepsi Penghuni Terhadap Karakter Fisik Perumahan Cluster di Kota Semarang," Tesis, Magister Teknik Arsitektur, Univ. Diponegoro, Semarang, Indonesia, 2008. [Online]. Available: https://core.ac.uk/download/pdf/11716241.pdf

[3] Statista, "Smart Home – Worldwide -*Statista Market Forecast*," Statista, Mar. 2024. [Online]. Available: https://www.statista.com/outlook/cmo/smart-home/worldwide

[4] Home Assistant, "Home Assistant," 2019. [Online]. Available: https://www.home-assistant.io/

[5] M. C. Suvar, L. Munteanu, and C. Cioară, "Optimal Monitoring of Server Rooms with Home Assistant Platform," in MATEC Web Conf., vol. 373, 2022, Art. no.00044, doi: 10.1051/matecconf/202237300044.

[6] B. K. Akhmetzhanov, O. A. Gazizuly, Z. Nurlan, and N. Zhakiyev, "Integration of a Video Surveillance System Into a Smart Home Using the Home Assistant Platform," in Proc. 2022 Int. Conf. Smart Inf. Syst. Technol. (SIST), Nur-Sultan, Kazakhstan, 2022, pp. 1–5, doi: 10.1109/SIST54437.2022.9945718.

[7] L. Munteanu, M. C. Suvar, and G. D. Florea, "Residential Security Through the Home Assistant Platform," in MATEC Web Conf., vol. 354, 2022, Art. no. 00008, doi: 10.1051/matecconf/202235400008.

[8] A. P. Rocha, M. Ketsmur, N. Almeida, and A. Teixeira, "An Accessible Smart Home Based on Integrated Multimodal Interaction," Sensors, vol. 21, no. 16, p.5464, Aug. 2021, doi: 10.3390/s21165464.

Authors Biographies

Alan Tri Arbani Hidayat is graduate of the Informatics and Computer Engineering Department at Politeknik Elektronika Negeri Surabaya (PENS), Indonesia. His research interests including Internet of Things (IoT), smart home systems, embedded systems, and edge computing. His academic activities focus on the development and integration of IoT-based systems for monitoring and automation applications.
Email: alanhhidayat92@gmail.com
ORCID: -

M. Udin Harun Al Rasyid received the B.Sc. degree in the Informatics Engineering Department from Sepuluh Nopember Institute of Technology (ITS), Indonesia, in 2004 and the Ph.D. degree in Computer and Communication Network Program, College of Electrical Engineering and Computer Science (CECS) from National Taiwan University of Science and Technology (NTUST), Taiwan, in 2012. He is currently an Professor at Informatics and Computer Engineering Department, Politeknik Elektronika Negeri Surabaya (PENS), Indonesia. He heads the research group of Mobile Sensing and Edge Computing Technology (MSECT) PENS. His Research Interest is mainly in Wireless Sensor Network (WSN), Wireless Body Area Network (WBAN), Internet of Things (IoT), and Web Technology.
Email: udinharun@pens.ac.id
ORCID: 0000-0002-4473-2364

Isbat Uzzin Nadhori received Bachelor's degree in Informatics Engineering ITS and a Master's degree in Electrical Engineering ITS specializing in Multimedia Intelligent Networks. He is currently a lecturer at the Department of Informatics and Computer Engineering, Politeknik Elektronika Negeri Surabaya (PENS), Indonesia. He is a member of the Mobile Sensing and Edge Computing Technology (MSECT) research group at PENS. His research interests include the Internet of Things (IoT), Network Security, and big data processing.
Email: isbat@pens.ac.id
ORCID: 0009-0005-9525-7398

Yeremia Ega Wahyudi is an undergraduate student in the Informatics and Computer Engineering Department at Politeknik Elektronika Negeri Surabaya (PENS), Indonesia. He is currently pursuing his Diploma IV degree, with research interests including Internet of Things (IoT), smart home systems, embedded systems, and edge computing. His academic activities focus on the development and integration of IoT-based systems for monitoring and automation applications.
Email: yremiawahyudi@gmail.com
ORCID: -

Yoga Prastyo Bayu Laksono is an undergraduate student in the Informatics and Computer Engineering Department at Politeknik Elektronika Negeri Surabaya (PENS), Indonesia. He is currently pursuing his Diploma IV degree, with research interests including Internet of Things (IoT), smart home systems, embedded systems, and edge computing. His academic activities focus on the development and integration of IoT-based systems for monitoring and automation applications.
Email: yogaprastyobayu@gmail.com
ORCID: -