Blynk-integrated ESP32 Based IoT System for Smart Home Light and Fan Control

Nor Suwadah Mohamed[1, a)], Nor Farahwahida Mohd Noor[1, b)], and Arasu Puventharan [2, c)]

[1]*Politeknik Kota Bharu,KM 24 Kok Lanas, 16450 Ketereh, Kelantan*
[2]*Politeknik Sultan Azlan Shah, Behrang Stesyen, 35950 Behrang, Perak.*

Author Emails
[a)] Corresponding author: suwadah@pkb.edu.my
[b)]farahwahida@pkb.edu.my
[c)]ararasu114@gmail.com

Received 15 October 2024, Accepted 5 December 2024, Published on 5 Januari 2025

Abstract The Internet of Things (IoT) has significantly transformed smart home automation by facilitating remote control and monitoring of household appliances. This paper presents the design and evaluation of a smart home system utilizing an ESP32 microcontroller integrated with the Blynk application for controlling lights and fans. The ESP32's robust processing power, built-in Wi-Fi, and Bluetooth capabilities, combined with Blynk's intuitive interface, offer a cost-effective and scalable solution for smart home automation. Thus, the system enables remote management of lighting and fan operations, including scheduling and energy consumption monitoring, thereby enhancing convenience, energy efficiency, and user control. The system's main components include the ESP32 microcontroller, relay modules for high-voltage control, an L298N motor driver for fan speed and direction, and a ME60N03 MOSFET PWM controller for adjustable light brightness. Hence, the system's design addresses challenges such as reliable Wi-Fi connectivity and device state synchronization, demonstrating practical applications of IoT in home automation. Results indicate that the system effectively integrates hardware and software to provide real-time feedback and control. Future work will incorporate ambient light sensors to optimize energy efficiency and user comfort by adjusting device operation based on environmental conditions. This research underscores the potential of combining advanced microcontroller technology with user-friendly IoT platforms to develop adaptable and efficient smart home systems.
Keywords: Smart Home, ESP32 Microcontroller, Lighting and Fan Control, Automation, Energy Efficiency

INTRODUCTION

The Internet of Things (IoT) has revolutionized the way we interact with our surroundings, particularly in the realm of smart home automation (1). By connecting everyday devices to the internet, IoT enables remote monitoring and control of household appliances, enhancing convenience, energy efficiency, and overall quality of life (2). Smart home systems can automate lighting, climate control, security, and entertainment, adapting to residents' preferences and routines (3). This technology not only offers convenience but also contributes to energy conservation and cost savings (4). As IoT continues to evolve, its applications in smart homes are becoming increasingly sophisticated, paving the way for more interconnected and intelligent living spaces that can anticipate and respond to occupants' needs (5).

The development and implementation of smart home systems have been greatly facilitated by the availability of affordable and versatile microcontrollers like the ESP32 (6). This study focuses on creating a smart home system for light and fan control using an ESP32 microcontroller integrated with the Blynk application (7). The ESP32, known for its powerful processing capabilities, built-in Wi-Fi and Bluetooth functionality, and low power consumption, serves as the central control unit of the system (8). Blynk, a popular IoT platform, provides a user-friendly interface for remote control and monitoring via smartphones (9). This combination allows for the creation of a cost-effective, scalable, and easily customizable smart home solution (10). The system enables users to control lights and fans remotely, set schedules, and monitor energy consumption, demonstrating the practical application of IoT in enhancing home automation and energy management (11).

The technical aspects of this smart home system encompass both hardware and software components (12). On the hardware side, the ESP32 microcontroller is connected to relays that control the lights and fans, while sensors may be incorporated to detect ambient conditions (13). The software implementation involves programming the ESP32 using the Arduino IDE, which allows for easy integration with various libraries and protocols (14). The Blynk library is utilized to establish communication between the ESP32 and the Blynk cloud server, enabling real-time control and data exchange (15). The system's design incorporates considerations for power efficiency, network security, and user experience (16). Implementation challenges, such as ensuring reliable Wi-Fi connectivity and managing device state synchronization, are addressed through careful programming and system architecture design (17).

This paper will delve into the design and evaluation of a smart home system for light and fan control using an ESP32 microcontroller integrated with the Blynk application. Leveraging the ESP32’s processing power and built-in Wi-Fi and Bluetooth functionality, along with the Blynk platform’s user-friendly interface, this study demonstrates a cost-effective and customizable smart home solution. The system allows remote control of lights and fans, scheduling, and energy monitoring, while addressing challenges like reliable Wi-Fi connectivity, device state synchronization, and power efficiency, showcasing the practical application of IoT in home automation.

LITERATURE REVIEW

The field of smart home automation has seen significant advancements in recent years, driven by the proliferation of Internet of Things (IoT) technologies. In (18), the authors provide a comprehensive review of smart home systems, highlighting the integration of various sensors and actuators to create intelligent living environments. Their work emphasizes the potential of IoT in enhancing energy efficiency and user comfort. Building on this, (19) presents a holistic overview of the IoT ecosystem for smart homes, discussing key enabling technologies, major application domains, and challenges in widespread adoption. They particularly note the importance of low-cost, energy-efficient hardware solutions in driving market growth.

The ESP32 microcontroller has emerged as a popular choice for IoT applications due to its versatility and cost-effectiveness. In (20), the capabilities of ESP32 in smart home applications are demonstrated, showcasing its ability to handle multiple sensors and actuators simultaneously while maintaining low power consumption. Similarly, (21) presents a detailed implementation of an ESP32-based smart home system, integrating various household appliances and providing remote control capabilities. Their work highlights the ESP32's suitability for real-time data processing and wireless communication in home automation contexts.

The integration of user-friendly mobile applications with IoT systems has been crucial in enhancing the accessibility of smart home technologies. Blynk, an IoT platform for mobile app creation, has gained traction in this regard. In (22), the authors demonstrate the effectiveness of Blynk in creating intuitive user interfaces for remote device control in their early work on IoT-based monitoring and control systems. More recently, (23) presents a comprehensive implementation of a Blynk-based home automation system using ESP8266, a predecessor to ESP32. Their work provides valuable insights into the integration of microcontrollers with the Blynk platform, paving the way for more advanced implementations using ESP32. These studies collectively underscore the potential of combining ESP32's hardware capabilities with Blynk's user-friendly interface to create efficient and accessible smart home solutions.

METHODOLOGY

Block diagram

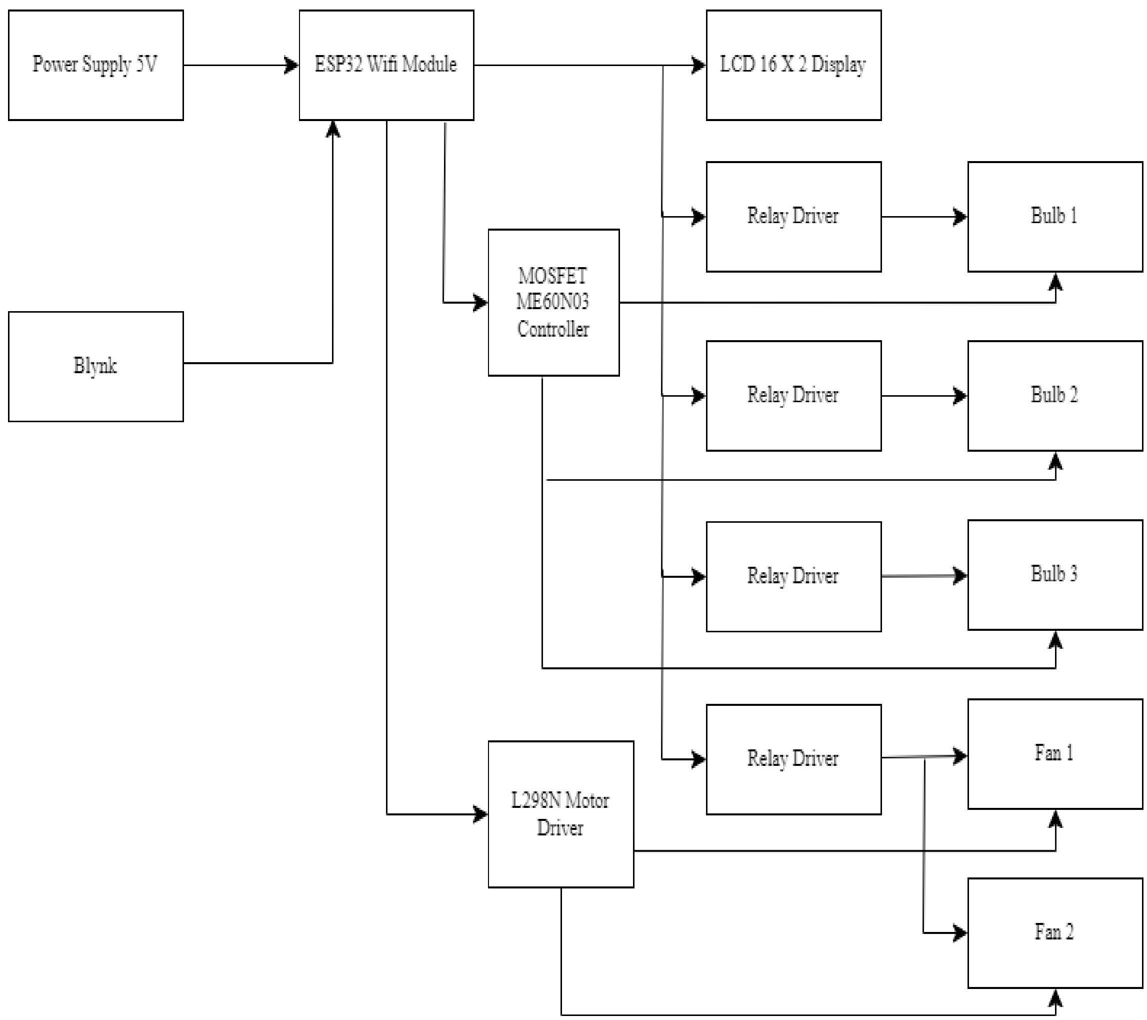

FIGURE 1. Block Diagram of the system

The block diagram in figure 1 depicts a Blynk-integrated ESP32 based IoT system for smart home light and fan control. A constant voltage supply (5V) powers the system, typically via a USB Type-C cable charger. The ESP32 microcontroller acts as the brain, receiving commands from the Blynk app through Wi-Fi and controlling connected devices. The L298N motor driver manages the motors for the fans, while the relay drivers, including the ME60N03 PWM controller, regulate power to lights and fans using relays. The output section displays information on the LCD 16 x 2 display and allows control of multiple light bulbs and a combination of DC fans (potentially adjusting speed). Arrows depict data flow, with the ESP32 receiving commands from the app and sending signals to control devices based on user input.

Flowchart

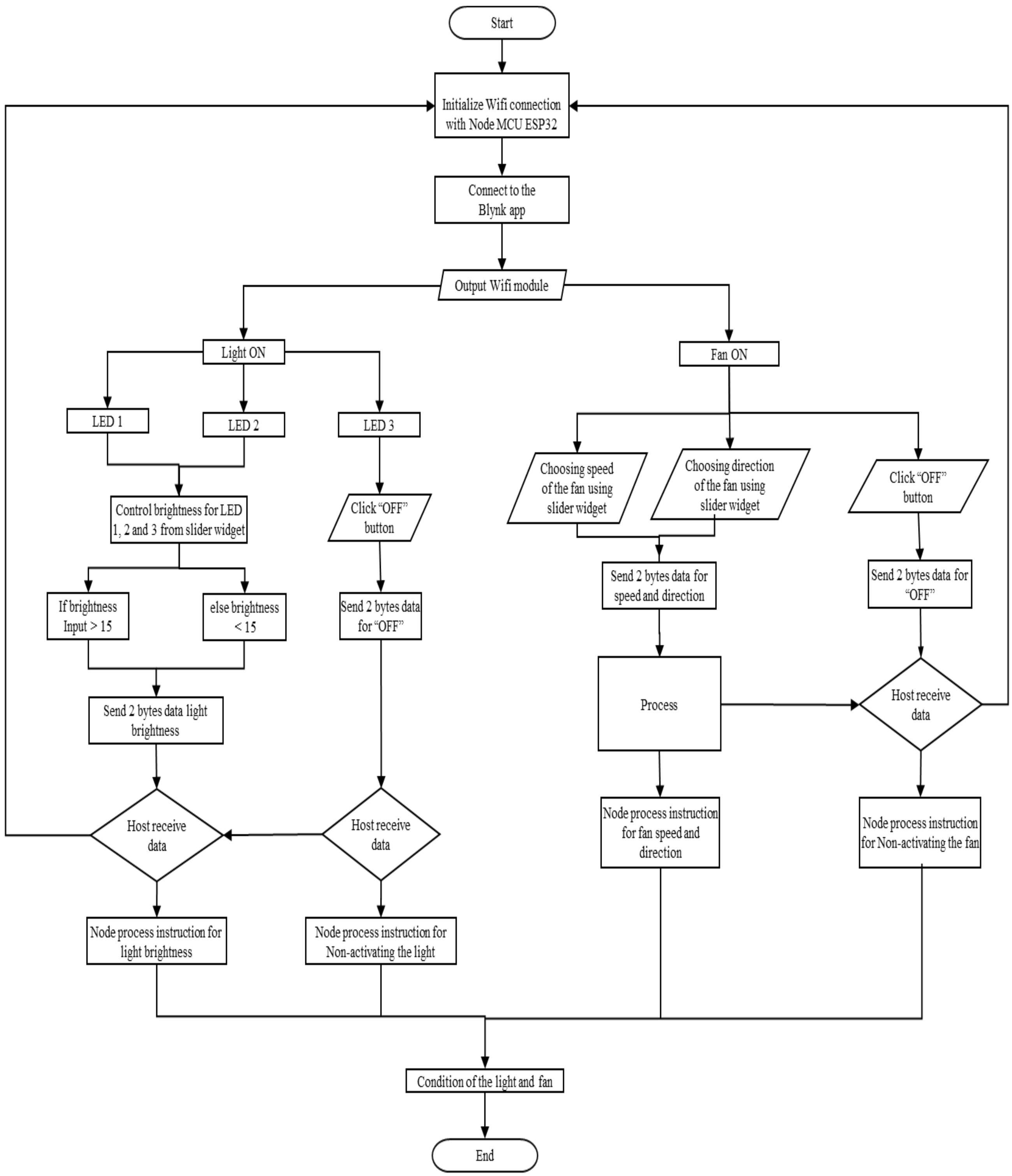
Start
Initialize Wifi connection with Node MCU ESP32
Connect to the Blynk app
Output Wifi module
Light ON
LED 1
LED 2
LED 3
Control brightness for LED 1, 2 and 3 from slider widget
If brightness Input > 15
else brightness < 15
Send 2 bytes data light brightness
Host receive data
Node process instruction for light brightness
Click "OFF" button
Send 2 bytes data for "OFF"
Host receive data
Node process instruction for Non-activating the light
Fan ON
Choosing speed of the fan using slider widget
Choosing direction of the fan using slider widget
Click "OFF" button
Send 2 bytes data for speed and direction
Process
Node process instruction for fan speed and direction
Send 2 bytes data for "OFF"
Host receive data
Node process instruction for Non-activating the fan
Condition of the light and fan
End

FIGURE 2. Flowchart of the system

Figure 2 show the flowchart outlines of the process for controlling a fan and light using a Blynk app and an ESP32. First, the ESP32 initializes Wi-Fi and connects to Blynk. The app can then display the connection status. Users can control light brightness with a slider: values above 15 turn the light on, while anything below turns it off. A separate button allows for complete shutoff. Fan speed is also controlled by a slider, while another lets users choose forward or reverse direction. Finally, a dedicated button allows for completely turning the fan off.

Software Development

With the PCB assembled, the next step involves configuring the Blynk software application and programming the ESP32. Blynk, an iOS and Android-compatible app for managing IoT devices and appliances, requires the creation of a Graphical User Interface (GUI) to control the fan and light. The setup process begins with downloading the Blynk app from the Google Play Store or Apple App Store, followed by user registration. Upon launching the application, users select "New Project" and are prompted to input project parameters. For this implementation, "ESP32" is chosen as the device and "Wi-Fi" as the connection type. This configuration establishes the foundation for integrating the hardware with the user-friendly Blynk interface, enabling remote control of the smart home system.

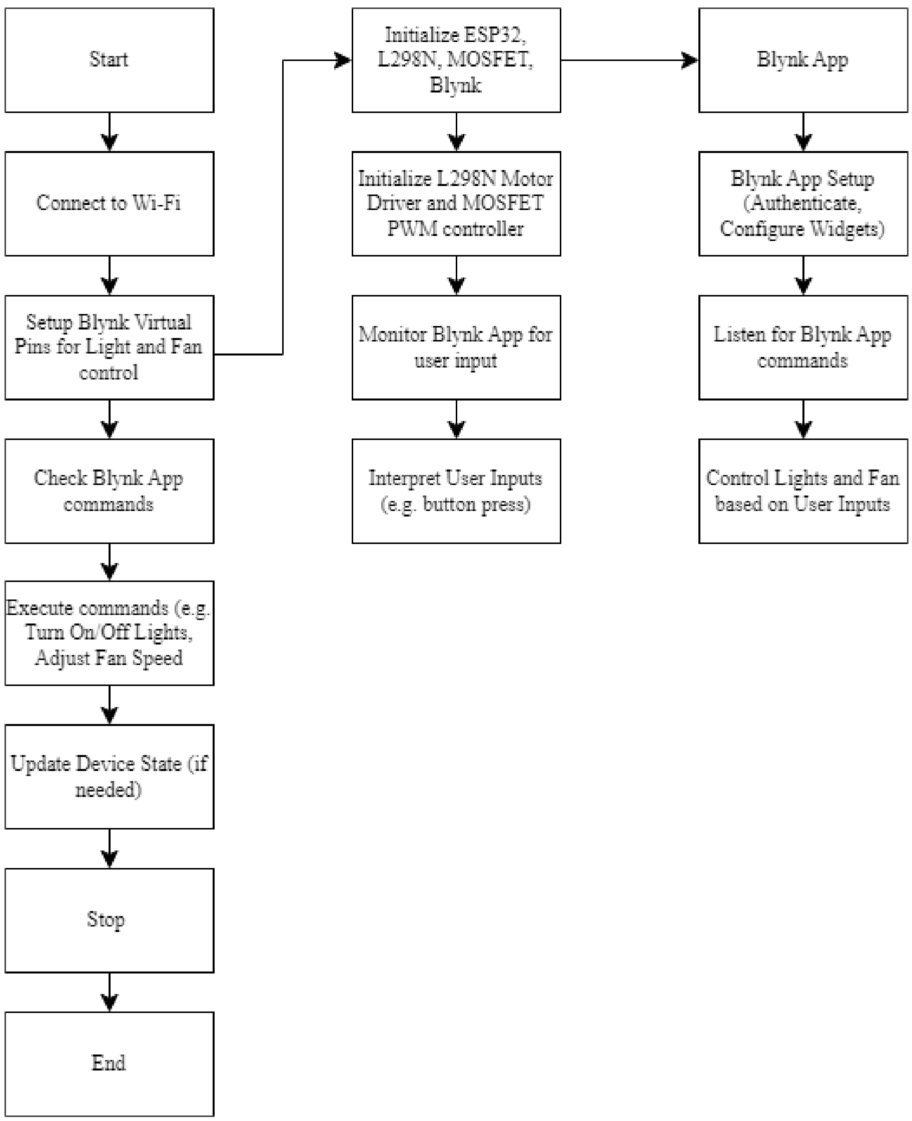

FIGURE 3. Block chart of the system

The system begins by initializing the ESP32, L298N motor driver, MOSFET PWM controller, and necessary pins. Subsequently, it establishes a Wi-Fi connection for communication with the Blynk app. After connecting to the Blynk app using an authentication token, virtual pins are set up to control lights and the fan. Users interact with the Blynk app to send commands, which the ESP32 reads through virtual pins. Based on these commands, the system controls the state of the lights and the fan. If available, light intensity is read from sensors or Blynk app sliders to adjust light output using PWM signals and the MOSFET controller. Similarly, fan speed is read from Blynk app sliders and adjusted using PWM signals and the L298N motor driver. If scheduling or timer features are implemented, the system checks for scheduled on/off times for lights and the fan. Finally, the system continuously updates the Blynk app with the current status of lights and the fan.

Hardware Development

The hardware development involves selecting electronic components and integrating all parts. Table 1 provides an overview of the key components used in the system, their primary functions, and the connections required for their operation. It serves as a reference guide to understand the system's architecture and how different components interact with each other

TABLE 1. System Components and Functions

Component	Function	Specifications / Connections
ESP32	Microcontroller for IoT communication	Connect to Wi-Fi, control lights and fan based on IoT commands.
Relay (x4)	Control high-voltage devices like lights and fans	Connect to ESP32 GPIO pins for on/off control
L298N Motor Driver	Control the fan motor speed and direction	Connect to ESP32 GPIO pins for motor control
MOSFET PWM Light Controller	Control the intensity of each light	Connect to ESP32 GPIO pins for motor control
Bulbs (x3)	Provide illumination	Connect to individual relay outputs for on/off control.
Fans (x2)	Provide air circulation	Connect to L298N motor driver for speed and direction control
Power Supply (Battery, DC adapter)	Provide power to ESP32, relay, motor driver	Connect to appropriate voltage inputs of ESP32, relay, and L298N
IoT Platform (Blynk)	Enable remote control of the system	Integrate with ESP32 for IoT functionality

The ESP32 microcontroller is used for this project due to its built-in Wi-Fi and low cost. The relay module primarily switches electrical devices on or off and isolates the control circuit from the controlled device. The L298N motor driver controls the fan’s speed and direction, allowing for precise adjustments and bidirectional operation for enhanced air circulation. The MOSFET ME60N03, combined with a PWM controller, is central to the light control system. Acting as a switch, the ME60N03 regulates current flow to the light source, while the PWM controller adjusts brightness by modulating the signal's duty cycle, enabling smooth, customizable, and energy-efficient lighting. Lights can be controlled using relays, MOSFET ME60N03, PWM controllers, solid-state relays, or smart switches. The microcontroller sends commands to turn lights on/off and adjust brightness based on sensor inputs or user commands. Fans can be controlled using motor drivers or relays. The microcontroller adjusts fan speed, direction, or turns it on/off based on sensor inputs or user commands. The power supply provides the necessary voltage and current for the circuit, including a voltage regulator for input regulation and capacitors for stabilization. It can connect to either an AC or DC source, depending on system requirements. The IoT platform Blynk enables remote control of the system and integrates with the ESP32 for enhanced IoT functionality.

.

RESULTS AND DISCUSSION

Prototype Development

FIGURE 4. The front view of the prototype system

The component hardware used includes ESP32, two relay modules, an LCD display, ME60N03 PWM controller (MOSFET), L298N motor driver, LED bulbs, two DC fans, a breadboard, female-to-male, female-to-female, and male-to-male wire connectors. The system uses a constant voltage supply of 5V provided by a USB Type-C cable charger.

The Blynk application is an IoT platform used to interact with the ESP32 and control output devices like light bulbs and fans. It can be downloaded from the Play Store for Android and the App Store for iOS users. Figure 5(a) shows the Blynk application's interface. A shortcut to the app is placed on the smartphone's home screen. Figure 5(b) shows the app's buttons, which can be used to turn devices on and off, as well as adjust light brightness and fan speed. The Blynk application has a user-friendly interface that is easy to configure, even for first-time users. The on and off buttons are available as soon as the app is opened. Additionally, users can use the app to turn on lights before arriving home.

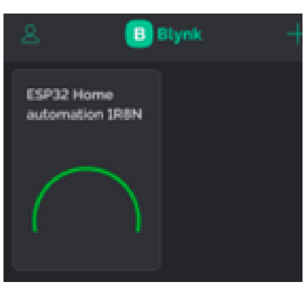

FIGURE 5(a). The interface of the Blynk application for the system

FIGURE 5(b) The interface of Blynk application for the system showing the ON and OFF button

The Blynk application features an intuitive and user-friendly dashboard, allowing users to effortlessly monitor and control their smart home light and fan system. With real-time updates, the interface provides instant feedback on the status of the lights and fans, ensuring users are always informed about their home's environment. The app offers customizable control widgets such as buttons, sliders, and switches, enabling users to tailor the interface to their preferences for seamless operation. It allows users to remotely access and manage their smart home devices from anywhere, providing unparalleled convenience and control. The Blynk interface ensures secure communication between the app and the ESP32-based system, leveraging robust encryption protocols to safeguard user data and device operation.

CONCLUSION

This IoT-based home automation system controls lights and fans using an ESP32 microcontroller, Blynk app, L298N motor driver, ME60N03 MOSFET PWM controller, and relays. It allows users to remotely monitor and control their home environment through a smartphone application. The system functionalities include on/off control for three lights and a fan, fan speed adjustment with directional control using the L298N motor driver, and dynamic adjustment of light brightness via the ME60N03 MOSFET PWM controllers. The system also provides real-time feedback on fan speed and direction through LCD updates. This user-friendly and versatile system offers control of lights, DC fans, and potentially AC devices through a Blynk app, leveraging the capabilities of the ESP32 microcontroller.

For future enhancements, ambient light sensors will be integrated to automatically adjust fan speeds and light intensities based on room occupancy and ambient lighting conditions, thereby enhancing energy efficiency and user comfort. These sensors will detect the presence of individuals in a room and the level of natural light, enabling the system to optimize the operation of fans and lights. By dynamically responding to real-time environmental changes, the system will reduce unnecessary energy consumption and provide a more comfortable living environment. This automated adjustment not only conserves energy but also extends the lifespan of the devices, contributing to a more sustainable and cost-effective smart home solution.

ACKNOWLEDGEMENTS

The authors would like to extend their gratitude to Politeknik Kota Bharu and Politeknik Sultan Azlan Shah for providing the facilities and resources necessary to conduct this research. Special thanks to the Blynk IoT platform team for their comprehensive documentation, which was instrumental in the development of this system. We also wish to acknowledge the technical support provided by colleagues and students who assisted during the hardware assembly and testing phases of the project. Lastly, heartfelt thanks to our families and friends for their encouragement and understanding throughout this research endeavor.

REFERENCES

1. Sun, Y., and Li, S., "A systematic review of the research framework and evolution of smart homes based on the Internet of Things," *Telecommunication Systems*, vol. 77, no. 3, pp. 597–623, 2021.
2. Dasgupta, A., Gill, A. Q., and Hussain, F., "Privacy of IoT-enabled smart home systems," in *Internet of Things (IoT) for Automated and Smart Applications*, pp. 9, 2019.
3. Karami, A. B., Fleury, A., Boonaert, J., and Lecoeuche, S., "User in the loop: Adaptive smart homes exploiting user feedback—state of the art and future directions," *Information*, vol. 7, no. 2, pp. 35, 2016.
4. Singh, P. P., Khosla, P. K., and Mittal, M., "Energy conservation in IoT-based smart home and its automation," in *Energy Conservation for IoT Devices: Concepts, Paradigms and Solutions*, pp. 155–177, 2019.
5. Alam, M. R., Reaz, M. B. I., and Ali, M. A. M., "A review of smart homes—Past, present, and future," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 1190–1203, 2012.

6. Hercog, D., Lerher, T., Truntič, M., and Težak, O., "Design and implementation of ESP32-based IoT devices," *Sensors*, vol. 23, no. 15, pp. 6739, 2023.
7. Kholik, M. A., Setiawan, F. B., and Fauzi, A., "Design and implementation of a smart home system with the Internet of Things (IoT) using ESP32," *Inspiration: Jurnal Teknologi Informasi dan Komunikasi*, vol. 13, no. 1, pp. 22–30, 2023.
8. Espressif Systems, "ESP32 Technical Reference Manual," 2023.
9. Blynk Inc., "Blynk IoT Platform Documentation," 2024.
10. Babiuch, M., and Postulka, J., "Smart home monitoring system using ESP32 microcontrollers," *Internet of Things*, pp. 82–101, 2020.
11. Zehnder, M., Wache, H., Witschel, H. F., Zanatta, D., and Rodriguez, M., "Energy saving in smart homes based on consumer behavior: A case study," in *Proc. 2015 IEEE First International Smart Cities Conf. (ISC2)*, pp. 1–6, Oct. 2015.
12. Bîrlog, I. A., Borcan, D. M., and Covrig, G. M., "Internet of things hardware and software," *Informatica Economica*, vol. 24, no. 2, pp. 54–62, 2020.
13. Imam, A. S., Garba, A., Isah, A. S., Chukwu, I. F., Baballe, M. A., and Shehu, S., "Design of an ESP32-based IoT smart home automation management system," *GJR Publication: Global Journal of Research in Engineering Computer Sciences*, vol. 4, no. 4, 2024.
14. Arduino LLC, "Arduino IDE for ESP32 Development," 2023.
15. Daş, R., and Ababaker, T., "Design and application of a smart home system based on Internet of Things," *European Journal of Technique (EJT)*, vol. 11, no. 1, pp. 34–42, 2021.
16. Bai, X., "Research on smart home system design and user experience improvement strategies," *Advances in Computer, Signals and Systems*, vol. 8, no. 5, pp. 34–39, 2024.
17. Pradeep, S., Kousalya, T., Suresh, K. A., and Edwin, J., "IoT and its connectivity challenges in smart home," *International Research Journal of Engineering and Technology (IRJET)*, vol. 3, pp. 1040–1043, 2016.
18. Alaa, M., et al., "A review of smart home applications based on Internet of Things," *Journal of Network and Computer Applications*, vol. 97, pp. 48–65, 2017.
19. Stojkoska, B. L. R., and Trivodaliev, K. V., "A review of Internet of Things for smart home: Challenges and solutions," *Journal of Cleaner Production*, vol. 140, pp. 1454–1464, 2017.
20. Maier, A., Sharp, A., and Vagapov, Y., "Comparative analysis and practical implementation of the ESP32 microcontroller module for the Internet of Things," in *Proc. 2019 Internet Technologies and Applications (ITA)*, pp. 143–148, 2019.
21. Kumar, S., Tiwari, P., and Zymbler, M., "Internet of Things is a revolutionary approach for future technology enhancement: a review," *Journal of Big Data*, vol. 7, no. 1, pp. 1–21, 2020.
22. Pavithra, D., and Balakrishnan, R., "IoT-based monitoring and control system for home automation," in *Proc. 2015 Global Conference on Communication Technologies (GCCT)*, pp. 169–173, 2015.
23. Kodali, R. K., and Mahesh, K. S., "A low-cost implementation of MQTT using ESP8266," in *Proc. 2017 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, pp. 404–408, 2017.