2024 8th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC) | 979-8-3503-7642-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/I-SMAC61858.2024.10714655

Remote based Home Automation with MQTT: ESP32 Nodes and Node-RED on Raspberry Pi

Dr. Arun Kumar Arigela
Department of CSIT
Marri Laxman Reddy Institute of Technology and Management
Hyderabad, Telangana, India
arun.arigala@gmail.com

Chandini Banapuram
Department of CSE (AI & ML)
G Narayanamma Institute of Technology & Science (For Women)
Hyderabad – Telangana, India
chandini.gnits@gmail.com

Dr. Nookala Venu
Center for Internet of Things
Madhav Institute of Technology & Science, Gwalior-474005, India
venunookala@mitsgwalior.in

Abstract— **Nowadays, Home automation has acquired a significant attention due to its potential to enhance convenience and efficiency in managing household systems. This paper presents a novel method to home automation utilizing the MQTT protocol, with a system is implemented using three ESP32 nodes and a Node- Red server is running on a Raspberry Pi, enabling users to control from Raspberry Pi as well as client terminals. The proposed system enables the users to control and monitor home appliances remotely from anywhere. By using the lightweight MQTT protocol, the system provides the real-time communication between the nodes and the server, ensure responsive control and data monitoring. The implementation of ESP32 nodes provides the flexibility and scalability, while Node-RED provides the user friendly interface for system management. Experimental results show the system effectiveness for maintaining reliable connectivity between nodes and server, also providing actionable insights into appliance usage. This work highlights the potential of MQTT-based home automation systems to provide enhanced control and monitoring capabilities in modern smart homes.**

Keywords— IoT, Home Automation, ESP32, Node- RED, Raspberry Pi, MQTT, Smart Home, Wireless Communication, Real-time Monitoring.

I. Introduction

The introduction of home automation technology has transformed the way households operate and manage various devices and systems. With the introduction of the Internet of Things (IoT), smart homes are becoming increasingly common, providing unmatched levels of convenience, efficiency, and energy savings. Home automation systems offer not only the convenience of remote control and monitoring but also enhance home safety and security. One of the essential factors enabling modern home automation is the Message Queuing Telemetry Transport (MQTT) protocol. MQTT is a lightweight, publish- subscribe network protocol that facilitates reliable and efficient communication between devices with minimal bandwidth usage. Its low overhead and ease of implementation make it an ideal choice for resource- constrained environments, which are common in home automation systems [1]. This paper presents a new approach to home automation using the MQTT protocol, implemented with three ESP32 nodes and a Node-RED server hosted on a Raspberry Pi. The ESP32, a cost-effective yet powerful microcontroller with built-in Wi-Fi and Bluetooth capabilities, serves as the core component of our system, enabling flexible and scalable deployment. Node-RED, an open-source flow-based development tool, provides a user-friendly interface for designing and managing the automation workflows [2]. Additionally, Page Kite is used on the Raspberry Pi to forward the 1880 port to the internet. The system that we propose aims to grant users the capability to remotely control and monitor their home appliances from any location with internet access. By utilizing MQTT, our system guarantees real-time communication and responsive control, which are crucial for effective home automation [3]. The combination of hardware and software components not only streamlines the setup and management of the system but also ensures robust and reliable performance. Through extensive experimentation, our system has demonstrated its ability to maintain consistent connectivity and provide useful information about appliance usage [4]. These results showcase the potential of MQTT-based home automation systems in enhancing the functionality and user experience in smart homes [5]. In the following sections, we will explore the details of our system's architecture, implementation, and experimental results, highlighting the advantages and practicality of the approach. RELATED WORKS

Research in home automation has investigated various technologies to enhance smart home systems. Integrated remote health monitoring with home automation and alarm systems, providing a multifaceted solution but lacking real-time communication capabilities. Vaigandla, K. K et al. [6] developed a phoneme-based voice-controlled system, enhancing accessibility but focusing heavily on voice recognition and Utilized multiple communication protocols, including IP and Bluetooth, to achieve interoperability, whereas our work simplifies communication using MQTT. Applied edge computing to improve data processing efficiency, but our system leverages MQTT for real-time communication without requiring additional infrastructure. Incorporated personal assistants for user interaction, contrasting with our approach that uses MQTT and Node-RED for broader automation capabilities. D. K et al. [7] Proposed a basic IoT sensing platform, while our system advances this by integrating MQTT for enhanced scalability Mr.RadhaKrishna Karne Et al. [8] Explored Zigbee for low- power communication, but MQTT provides greater compatibility and real-time benefits. Venu, D. N. et al. [9] demonstrated on LPWAN technologies, which, while efficient for range, have slower data transmission compared to our MQTT-based approach Used cloud- based architectures for scalability, whereas our local MQTT broker and Node-RED enhance performance and reduce reliance on external cloud services. Combined wireless and wired technologies for smart lighting, but our system's MQTT integration offers superior scalability [11]. Several researchers around the world have designed and focused on security with IoT and machine learning, while our research provides a comprehensive automation framework that addresses various functionalities [12]. Our work distinguishes itself by integrating MQTT with ESP32 nodes and Node-RED, addressing limitations of existing solutions and offering a robust, real-time automation system.

II. Proposed System Architecture

The system utilizes the MQTT protocol for efficient real-time data acquisition and transmission, allowing users to remotely monitor and control their home environment using a user-friendly interface. The Raspberry Pi 4, acting as the central gateway server, is at the heart of the system and is connected to the home network [13]. It functions as the control hub, integrating and managing data from the ESP32 nodes and ensuring seamless communication with the MQTT broker [14]. The system's architecture comprises three ESP32 nodes, each serving a specific function within the network [15]. Node 1 controls multiple appliances and lighting fixtures in the common hall through six relay modules and receives control commands and status updates via the MQTT protocol, ensuring precise and responsive device management [16]. Node 2 includes an array of sensors, such as temperature, humidity, light intensity, and harmful gases, as well as two relays that are controlled at the edge according to the light intensity of outside the lights get off if sunlight is there, and if not, the lights get automatically turned on and show status on the Node-RED dashboard [17]. It performs outdoor environmental monitoring and automatically adjusts outdoor lighting based on real-time light intensity data. Node 3 is similar to Node 2 but is dedicated to monitoring indoor environmental conditions, providing comprehensive data coverage within the home [18]. Fig. 1 shows the proposed system architecture, detailing the integration of the ESP32 nodes with the Raspberry Pi 4. The diagram depicts the MQTT-based communication framework between the nodes and the central Node-RED server, which facilitates real-time data visualization and control. The Raspberry Pi 4 also serves as the MQTT broker, managing the data flow and ensuring reliable communication between all system components [19]. Pagekite is employed to enable remote access by forwarding port 1880, allowing users to interact with the Node-RED dashboard via a static URL.

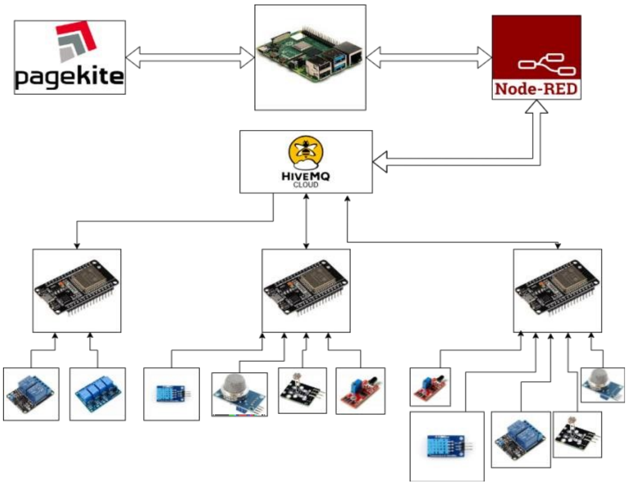


Fig 1. Overview of MQTT System Architecture

i.RASPBERRY PI : The Raspberry Pi 4, which has 4GB of RAM and a 32GB SD card, acts as the central hub for this home automation system. It consists of a quad-core Cortex-A72 processor and dual-band wireless LAN for robust performance and connectivity. The 4GB of RAM enables real-time data processing, while the 32GB SD card offers ample storage for the Node-RED server and MQTT broker. This setup allows for seamless communication between the ESP32 nodes and remote access through Pagekite [20].

ii.ESP32: The ESP32 is a versatile microcontroller that integrates Wi-Fi and Bluetooth connectivity. It features a Dual-core processor, extensive I/O options, and low power consumption. In this system, ESP32 modules facilitate appliance control, environmental monitoring, and data communication with the Raspberry Pi using MQTT, enabling efficient and remote home automation.

iii. Relay: 8Relays are electrically operated switches that allow low-power signals to control higher-power circuits. They consist of an electromagnet that actuates a switch mechanism, enabling or disabling the connected load. In this system, relay modules are utilized to control various appliances and devices based on signals from the ESP32 microcontrollers, ensuring reliable switching for home automation tasks.

iv. DHT11: The DHT11 is a cost-effective digital sensor that measures temperature and humidity accurately. It offers a straightforward interface, making it suitable for various environmental monitoring applications. In this system, the DHT11 sensor monitors indoor and outdoor temperature and humidity levels, transmitting the data to the ESP32 for processing and visualization in the Node- RED dashboard.

v. MQ135: The MQ135 gas sensor is a versatile device capable of detecting various gases, including carbon dioxide (CO_2), ammonia (NH_3), nitrogen dioxide (NO_2), carbon monoxide (CO), and benzene. Due to its sensitivity to harmful gases, it is widely employed in air quality monitoring systems. In this particular system, the MQ135 sensor detects and measures gas concentrations, with the gathered data being sent to the ESP32 for processing and visualization.

vi. KY018: The KY-018 is a light intensity sensor that employs a photo resistor (LDR) to detect changes in light levels. It outputs an analog signal proportional to the light intensity detected. In this system, the KY-018 sensor measures light conditions, allowing the ESP32 to control lighting and execute automated responses based on the detected light levels.

vii. Flame Sensor: A flame sensor is designed to identify the presence of flames or fire. It uses infrared light to detect the specific wavelengths emitted by flames, alerting users when flames are detected. In this system, the flame sensor monitors for fire or flame conditions, providing valuable data to the ESP32 for real-time reactions and integration into the home automation setup [10].

viii. HiveMQ: HiveMQ is a scalable MQTT broker that allows for real-time communication between devices. It efficiently facilitates messaging by publishing and subscribing to messages across a network [21]. In this system, HiveMQ functions as the central messaging broker, ensuring reliable and real-time data transmission between the ESP32 nodes and the Raspberry Pi for the home automation system.

ix. Node-RED: Node-RED is a flow-based development tool for visual programming, designed to connect devices and services with ease. It provides a user-friendly interface for wiring together hardware devices, APIs, and online services. In this system, Node-RED runs on the Raspberry Pi, allowing users to manage and visualize data from ESP32 nodes, automate tasks, and create custom dashboards for real-time monitoring and control of the home automation system [22].

x. PAGEKITE : Pagekite is a service that enables remote access to local devices and servers over the internet by creating a public URL. This service provides a secure way to expose services running on local networks to the external world. In the home automation system, Pagekite is utilized to create a static URL for accessing the Node-RED dashboard hosted on the Raspberry Pi, allowing users to monitor and control the system remotely from any location.

III. RESULT AND DISCUSSION

The development of the home automation system involved a systematic approach, integrating hardware, schematic design, software configuration, and system integration to achieve a cohesive and functional solution.

Hardware Setup and Configuration:
The system comprises three ESP32 nodes, each designed for specific functions. Node 1 is responsible for controlling lighting and appliances in the common hall. It uses six relay modules, each managing a separate device, allowing for individual management of various appliances. The practical setup is illustrated in Figure 2, which captures the actual hardware arrangement with the relay modules and connected appliances.

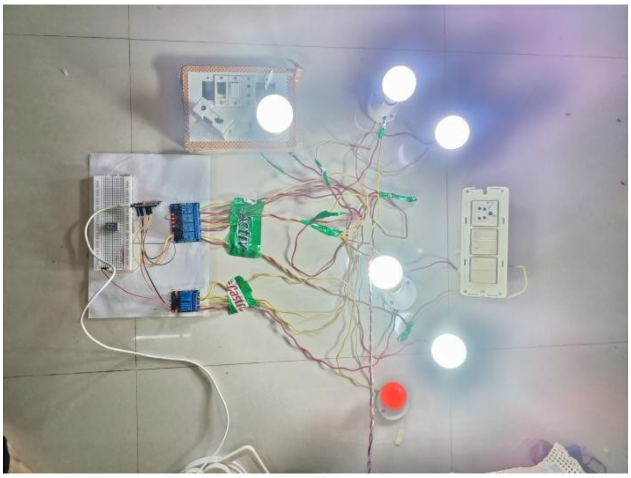


Fig. 2. Implemented Node 1.

Node 2 is configured for outdoor environmental monitoring. It integrates multiple sensors, including the DHT11 for temperature and humidity, the MQ135 for detecting harmful gases, the KY018 for light intensity, and a flame sensor for fire detection. Additionally, Node 2 includes two relay modules to control outdoor lighting based on light intensity. The implementation photo in Figure 3 shows the real-world setup of Node 2, highlighting the sensors and relay modules in an outdoor environment. Fig.3 depicts the implementation photo of Node 2. Fig.3. display's the implemented prototype of the ESP8266 smart Home of MQTT and Node-RED where MQ2 Gas sensors, Buzzer are interfaced with the NodeMCU ESP8266 to get the desired experimental result.

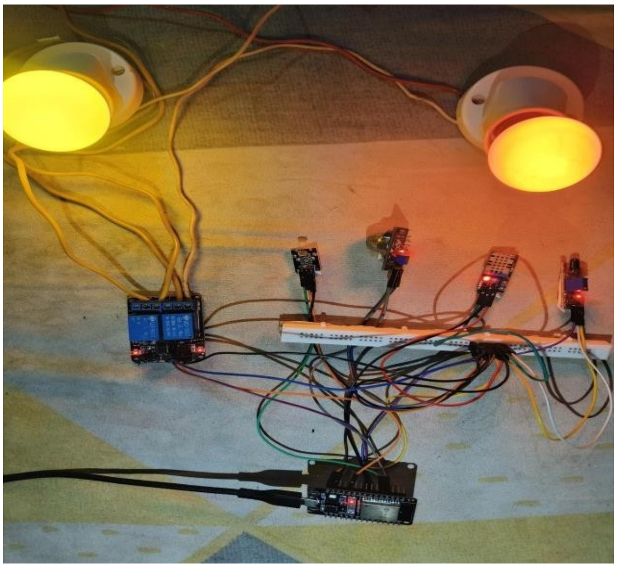


Fig.3. Implemented Node 2.

Node 3 mirrors Node 2 but is designed for indoor monitoring. It uses the same set of sensors to measure indoor environmental conditions. Fig. 4 depicts the implementation photo of Node 3, showcasing the ESP32 and its associated sensors configured for indoor use. The use of Pagekite in the system facilitates remote access and monitoring, while the ESP32 nodes and sensors enable efficient environmental monitoring and control of lighting and appliances.

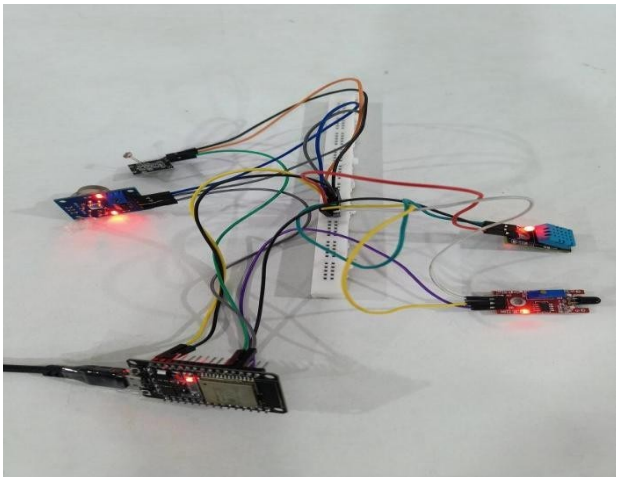


Fig.4. Implemented Node 3.

Network Setup:

The network setup, a crucial aspect, facilitates seamless uplink and downlink communication between the ESP32 nodes and the Raspberry Pi via the MQTT protocol. Each ESP32 node connects to the home Wi-Fi network and publishes sensor data to the HiveMQ broker using three separate topics: <Hm1>, <Hm2>, and <HM3>. The HiveMQ broker, hosted in the cloud, manages data exchanges, ensuring efficient transmission between the ESP32 nodes and the Node-RED server on the Raspberry Pi. This setup empowers the Raspberry Pi to receive data from the nodes (uplink) and send control commands back to Node 1 for managing lights and appliances (downlink) through the dashboard interface.

Hardware Setup:

Fig. 5 illustrates the network architecture, depicting the ESP32 nodes publishing data to the HiveMQ broker and the Raspberry Pi subscribing to these topics, enabling real-time data exchange and control capabilities.

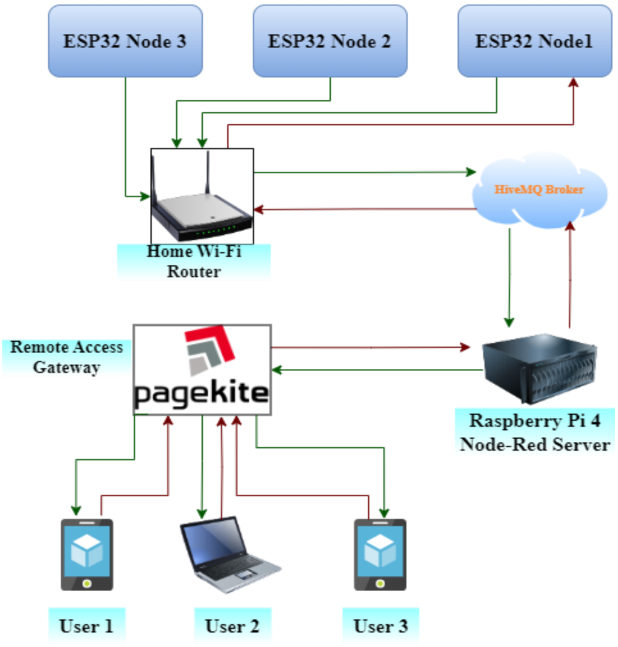


Fig.5. EPS 32 Network Architecture.

As shown in the figure above, hardware composes of the following:

i. ESP32 Microcontroller:

ESP 32, which is a low-cost microcontroller that replaces the previous version ESP8266, is at the heart of the architecture of hardware. Along with integrated Wi-Fi, ESP 32 offers a variety of functionalities compatible with a wide range of sensors for a different use, including IoT and streaming its specification. It as well offers programmers a powerful toolkit with dual cores, 240 MHz, 520 Kbyte SRAM and peripherals including: I2C, DAC, ADC, I2S, SPI, UART, 34 physical GPIO pins (Fig. 5). The power management unit and the lower power controller enable the ESP32 to run lower than 1mA in deep sleep mode. These advantages make the ESP32 a better device for low power applications.

ii. ESP32-CAM: The ESP32-CAM is a low-cost card that includes an ESP32 microcontroller and an OV2640 camera. One of the most interesting features of the ESP32 is the ability to communicate by Wi-Fi, an obvious use of the ESP-32 CAM is to transmit live video images via Wi-Fi (surveillance camera, etc.). In addition to an OV2640 camera, the module is equipped with a micro-SD card reader that can be used to store images or video sequences.

Software Setup:

The software setup revolves around the Raspberry Pi, which hosts the Node-RED server and employs PageKite for data processing and visualization. Node-RED processes incoming data from the HiveMQ broker and updates the dashboard with real-time visualizations and control options. Users can interact with the dashboard to monitor environmental conditions and manage devices. To facilitate remote access, PageKite is utilized to generate a static URL for the Node-RED dashboard. This setup allows users to securely access and controls their home automation system from any location with internet connectivity.

The ESP32 nodes use the MQTT protocol to publish sensor data at regular intervals. The MQTT protocol is lightweight, ensuring efficient data transmission, minimizing latency, and maximizing responsiveness. The proper integration of network and software systems ensures seamless operation of the home automation system, delivering real-time insights and control over the home environment to users. By incorporating Protocol-based and local components, the system becomes more adaptable and scalable, making it the optimal solution for modern smart homes.

In order to evaluate the proposed home automation system's effectiveness in monitoring and controlling environmental parameters and appliances, it was thoroughly tested. The Node-RED dashboard was utilized to visually represent data and manage device interactions, providing users with real-time control and insights.

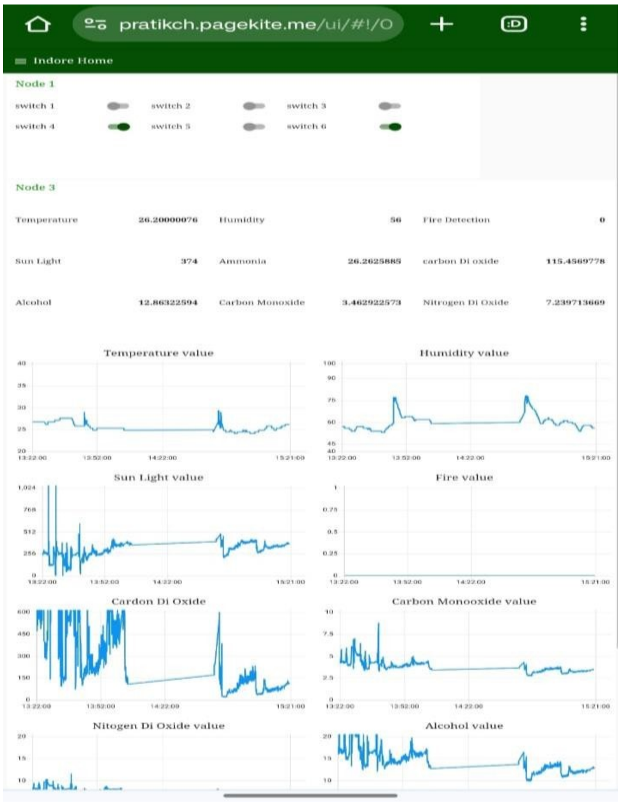
Monitoring and Control:

The Node-RED dashboard for ESP32 Node 1 in Fig. 6 controls six relay modules that regulate lights and other appliances in the common hall. Users can manually toggle each relay and also benefit from automated control based on preset schedules or conditions. The system's responsiveness guarantees smooth user interaction, enhancing the overall convenience of appliance management.

Fig. 6 displays the dashboard for ESP32 Node 3, which is responsible for outdoor environmental monitoring. This node includes DHT11, MQ135, KY- 018, and flame sensors, providing comprehensive data on temperature, humidity, gases concentrations, and light intensity. The dashboard displays these parameters in real-time, enabling users to effectively monitor changes. The two relays connected to this node operate based on light intensity, automatically turning outdoor lights on or off as per ambient conditions, showcasing an effective edge IoT implementation.

This system monitors light intensity, temperature, humidity, and pressure values. If the recorded light intensity value goes beyond the set high or low value of the threshold level, the controller will generate the signal to turn the LED OFF/ON. Then the user will be notified by the LED status on the dashboard, this feature is presented in Fig. 6 and 7.

The ESP32 controller subscribes to the published command, created by the end-user through the web interface, and transmitted by the Node-Red. Based on the received message, ESP32 sends a command to the LED to turn it ON/OFF. The implemented system is tested to evaluate its capability. Results provided in Fig. 6 show the recorded values of temperature, humidity, and light density, and pressure, which are acquired and displayed over time through a data-viewing dashboard.



Node 1

switch 1	○	switch 2	○	switch 3	○
switch 4	●	switch 5	○	switch 6	●

Node 3

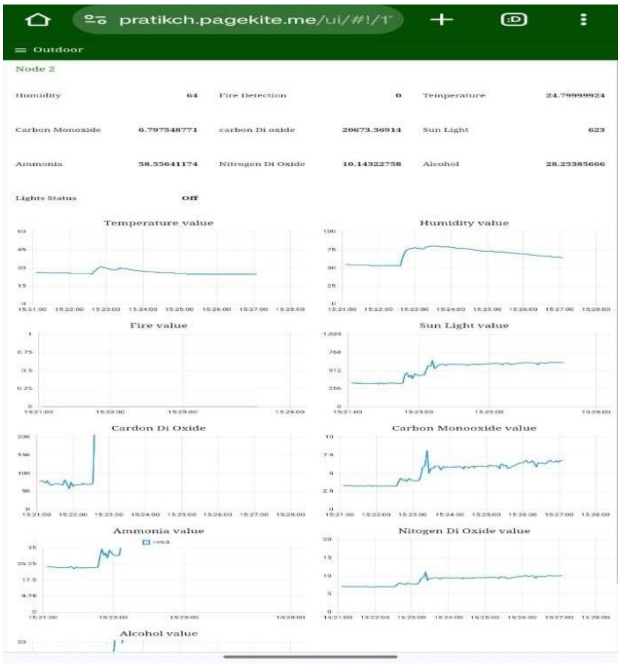
Temperature	26.20000076	Humidity	56	Fire Detection	0
Sun Light	374	Ammonia	26.2625885	carbon Di oxide	115.4569778
Alcohol	12.86322594	Carbon Monoxide	3.462922573	Nitrogen Di Oxide	7.239713669

Fig. 6. Web Dashboard Node 1 & 3

Fig. 7 depicts the Node-RED dashboard for ESP32 Node 2, which is designed for indoor monitoring using the same sensor suite as Node 2.

The data can be collected by ESP32- S2 microcontroller using the sensors output pins and onwards transferred to the locally installed Node-Red IoT server. NodeRed IoT server receives energy demand form the web interface developed to initiate trading sessions and start energy transfer process automatically.

It can be installed on a Linux- based platform and it provides a browser-based editor that makes it easy to write together flows using various nodes in the Platte that can be deployed to run time and This setup allows for the tracking of indoor environmental conditions, enhancing safety and comfort by warning users of potential hazards, such as gas leaks or fires. The consistent and accurate sensor data enables precise indoor climate control.



Node 2

Humidity	64	Fire Detection	0	Temperature	24.79999924
Carbon Monoxide	6.797548771	carbon Di oxide	20673.36914	Sun Light	623
Ammonia	58.55641174	Nitrogen Di Oxide	10.14322758	Alcohol	28.25385666
Lights Status	Off				

Fig.7. Web Dashboard Node 2

Performance Evaluation:

The system utilizes the MQTT protocol to ensure efficient and dependable communication between the ESP32 nodes and the Raspberry Pi. Data is transmitted with minimal latency, enabling real-time updates on the Node-RED dashboard. The use of HiveMQ as an MQTT broker enhances data flow and robustness, ensuring uninterrupted service. Remote access to the system is facilitated by Pagekite, which provides a static URL for users to access the Node-RED dashboard from any location. This feature significantly improves user convenience by offering control and monitoring capabilities beyond the local network.

Discussion:

The results confirm the effectiveness of the proposed home automation system in providing remote monitoring and control capabilities. Its architecture, leveraging lightweight MQTT communication and versatile ESP32 nodes, offers a scalable and cost-effective solution for smart home applications. The integration of sensor-based automation, particularly for lighting control, demonstrates substantial potential for energy efficiency improvements. By adjusting lighting based on environmental conditions, the system reduces unnecessary energy consumption, contributing to sustainability.

IV. CONCLUSION

This research designed and developed a home automation system that employs ESP32 microcontrollers, MQTT protocol, and a Node-RED server on a Raspberry Pi 4, presenting an adaptable and efficient solution for smart home administration. The deployment of sensors and relay modules across three ESP32 nodes offers extensive monitoring and control capabilities for both indoor and outdoor settings. The developed system effectively showcases real- time environmental monitoring and appliance management, enhancing user convenience and energy efficiency. The utilization of open-source components ensures cost-effectiveness and flexibility, making it a feasible choice for various applications. The proposed system emphasizes the potential of IoT-based systems to revolutionize household management by delivering practical insights and automated control. Future enhancements could involve advanced sensor integration and machine learning for predictive analytics, further improving the capabilities and impact of home automation systems. Once the message is sent through Node-Red along with the MQTT protocol to the microcontroller, it will be executed by ESP32 and turns the devices ON/OFF. The developed system cost is low, simple to operate, and is simply embedded with home devices.

Future work could explore integrating additional sensors and expanding the system's control capabilities, as well as employing advanced data analytics to generate more insightful and actionable information from the collected data.

REFERENCES

[1] ulhafz Ahmed Mustofa, Yohannes Alemayehu Dagnew, Prabhakar Gantela ,and M. Javed Idrisi, “SECHA: A Smart Energy-Efficient and Cost-Effective Home Automation System for Developing Countries” in Journal of Computer Networks and Communications, 2023.

[2] Shubham Gupta , Saurav , Vidit Patira , Nishant Jain and Dr. Vijay Kumar, “Home Automation Solution Using Node-Red and MQTT” in International Journal of Advanced Research in Computer and Communication Engineering, 2021.

[3] Md. Sadad Mahamud, Md. Saniat Rahman Zishan, Syed Ishmam Ahmad, Ahmed Rezaur Rahman, Mehedi Hasan and Md.Lutfur Rahman, “Domicile - An IoT Based Smart Home Automation System” inInternational Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), 2019.

[4] V. Pravalika and Ch. Rajendra Prasad “Internet of Things Based Home Monitoring and Device Control Using Esp32” in International Journal of Recent Technology and Engineering (IJRTE), July 2019.

[5] Kodali, Ravi Kishore, and SreeRamya Soratkal. "MQTT based home automation system using ESP8266." 2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC). IEEE, 2016.

[6] Arun Kumar, A., & Vaigandla, K. K. (2022). Review of Internet of Things (IoT) for Future Generation Wireless Communications. *International Journal for Modern Trends in Science and Technology*, *8*(03), 01-08.

[7] D. K. (2022). Investigation on Internet of Things (IoT):Technologies, Challenges and Applications in Healthcare. International Journal of Research , XI (II), 208-218.

[8] Mr.RadhaKrishna Karne, M. M. (2022). Applications of IoT on Intrusion Detection System with Deep Learning Analysis. International Jourfor Innovative Engineering and Management Research , 11 (06), 227-232.

[9] Venu, D. N. (2022). Smart Agriculture Remote Monitoring System Using Low Power IOT Network. IJFANS International Journal of Food and Nutritional Sciences , 11 (6), 327-340.

[10] Venu, D. N. (2022). IOT Surveillance Robot Using ESP-32 Wi-Fi CAM & Arduino. IJFANS International Journal of Food and Nutritional Sciences , 11 (5), 198-205.

[11] Venu, D. N. (2022). IOT Based Speech Recognition System to Improve the Performance of Emotion Detection. IJFANS International Journal of Food and Nutritional Sciences, 11 (3), 92-102.

[12] Sowmya Jagadeesan, M. K. (2022). Implementation of an Internet of Things and Machine learning Based Smart Medicine Assistive System for Patients with Memory Impairment. IJFANS International Journal of Food and Nutritional Sciences , 1191-1202.

[13] Venu, D. N. (2023, September). IoT based Real Time Street Lights controlling on Motion Detection. *European Chemical Bulletin, 12*(9), 270-287. doi: https://www.eurchembull.com/uploads/paper/cdab7176a12303abe8ee8796b22267b6.pdf.

[14] N. Deshai, A. K. Arigela, S. Ashwini, N. N. Jose, V. Palanivel, "SecureSwipe Enhancing Card Transactions Through Gradient Boosted Fraud Detection," 2024 10th International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, India, 2024, pp. 394-399, doi: 10.1109/ICCSP60870.2024.10544369.

[15] Venu, D. N. (2023, September). IoT based Real Time Street Lights controlling on Motion Detection. *European Chemical Bulletin, 12*(9), 270-287. doi: https://www.eurchembull.com/uploads/paper/cdab7176a12303abe8ee8796b22267b6.pdf .

[16] N. Deshai, A. K. Arigela, S. Ashwini, N. N. Jose, V. Palanivel, "SecureSwipe Enhancing Card Transactions Through Gradient Boosted Fraud Detection," 2024 10th International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, India, 2024, pp. 394-399, doi: 10.1109/ICCSP60870.2024.10544369.

[17] N. N. Jose, A. K. Arigela, G. Vivekanandan, R. Sethuraman, S. B. T. Naganathan, "“Optimizing Payment Transaction Security: Utilizing Gradient Boosting Machines for Fraud Detection”," 2024 10th International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, India, 2024, pp. 720-725, doi: 10.1109/ICCSP60870.2024.10543774.

[18] S. R. Raja, H. Arya, Ramya.D, S. Sivakumar, A. K. Arigela, "Development of Green Computing through Power Management in Wireless Networks using Machine Learning," 2024 International Conference on Inventive Computation Technologies (ICICT), Lalitpur, Nepal, 2024, pp. 1911-1916, doi: 10.1109/ICICT60155.2024.10544499.

[19] B. Nancharaiah, S. B, A. Arun Kumar, P. Elizabeth Kalpana and A. C, "Analysis of Visual Expression of Light and Color in Movies Based on Wavelet Neural Network," 2023 Second International Conference On Smart Technologies For Smart Nation (SmartTechCon), Singapore, Singapore, 2023, pp. 245-249, doi: 10.1109/SmartTechCon57526.2023.10391687.

[20] B. Nancharaiah, A. Visva Sangeetha, A. Brahmareddy, R. M and R. Vijaya Kumar Reddy, "Analysis of Visual Communication Effect of Color in Print Advertising Design Based on Decision Tree Classification Algorithm," 2023 Second International Conference On Smart Technologies For Smart Nation (SmartTechCon), Singapore, Singapore, 2023, pp. 804-808, doi: 10.1109/SmartTechCon57526.2023.10391823.

[21] Reddy, A. V., Kumar, A. A & Reddy, R. V. K. (2022). On optimization efficiency of scalability and availability of cloud-based software services using scale rate limiting algorithm. Measurement: Sensors, 24, 100468.

[22] Swathi, R., Sarangi, Sanjaya Kumar, Subashini, V., Arulkumar, D., Ralhan, Shimpy, Debtera, Baru, Optimization of Hello Message Broadcasting Prediction Model for Stability Analysis, Wireless Communications and Mobile Computing, 2022, 2785810, 9 pages, 2022. https://doi.org/10.1155/2022/2785810.