Journal of Advanced Research in Operating System Development and Evolution
Volume 1, Issue 2 - 2025, Pg. No. 5-8
Peer Reviewed Journal

Review Article

A Secure and Cost-Effective IoT Framework for Scalable Smart Lighting and Appliance Control

Arjun V Bala[1], Kalpesh H Surati[2], Bhushan D Joshi[3], Drashti D Ruparelia[4]

[1,3,4]Student, Darshan University, Rajkot, India
[2]Student, Recstech Technology, India
DOI: https://doi.org/10.24321/3051.4282.202502

I N F O

Corresponding Author:
Arjun V Bala, Darshan University, Rajkot, India
E-mail Id:
arjun.bala@darshan.ac.in
Orcid Id:
https://orcid.org/0009-0008-5505-634X
How to cite this article:
Bala A V, Surati K H, Joshi B D, Ruparelia D D. A Secure and Cost-Effective IoT Framework for Scalable Smart Lighting and Appliance Control. *J Adv Res Oper Syst Dev Evol* 2025; 1(2): 5-8.

Date of Submission: 2025-08-15
Date of Acceptance: 2025-09-06

A B S T R A C T

An IoT-based smart lighting and appliance control system using an Arduino Mega, relay modules, and motion sensors is designed and implemented with very Low cost. The system allows remote operation of lights, fans, and air conditioners through a web application developed with Angular (frontend) and Node.js (backend). Communication between the hardware and server is handled via the MQTT protocol. To enhance security, a selfhosted MQTT server was developed, ensuring controlled access, encrypted communication, and authentication for connected devices. Self-hosted MQTT server prevents unauthorized commands and data interception, addressing common vulnerabilities found in public MQTT brokers. Additionally, the system incorporates motion sensor-based automation, enabling lights to turn on only when movement is detected, thereby improving energy efficiency. Testing confirmed the system's ability to deliver secure, fast, and reliable control, making it suitable for residential and commercial applications, with potential for future scalability.

Keywords: IoT, Arduino Mega, MQTT, PIR, PHP, Angular, Node.js

Introduction

The Internet of Things (IoT) is transforming everyday devices into intelligent, interconnected systems capable of automation, remote operation, and seamless communication. In the realm of smart home automation, IoT offers significant benefits such as improved energy efficiency, enhanced user convenience, and real-time monitoring[1],[2-3]. However, traditional lighting and appliance control systems still dominate in many settings, relying on manual operation through physical switches. This results in energy wastage from unattended devices, lack of automation to respond to occupancy or environmental changes, and no remote access for users. Furthermore, IoT implementations using public platforms without robust security measures are vulnerable to unauthorized access and manipulation.[4] To overcome these limitations, this paper presents an IoT-based smart lighting and appliance control system that integrates the Arduino Mega, relay modules, and motion sensors with a secure, selfhosted MQTT server.

The proposed system employs motion-based automation to reduce unnecessary energy consumption, while providing remote and manual control through a web application built using Angular for the frontend and Node.js for the backend. The self-hosted MQTT server replaces public brokers to ensure secure communication via encryption and authentication[5,6] mitigating vulnerabilities associated with open IoT networks. This architecture not only enhances operational efficiency and safety but also offers scalability for residential, commercial, and industrial environments, making it a robust solution for modern smart infrastructure.

Components of the system

Arduino mega 2560 and it's peripherals

The Arduino Mega 2560 is the powerhouse of our IoT-based lighting and appliance control concept.[7,8] Built around the ATmega2560 microcontroller, it is designed for projects where multiple devices and sensors need to operate seamlessly together. With 54 digital I/O pins, 16 analog inputs, and 4 dedicated serial ports, it offers significantly more expansion capacity than most boards in its category. Coupled with 256 KB of flash memory and 8 KB of SRAM, it can handle complex automation logic without straining resources. This makes it ideal for tasks like reading motion sensor inputs, controlling multiple relays for lights, fans, and AC units, and delivering real-time responses to user commands from an application interface.

The decision to select Arduino Mega over alternatives such as NodeMCU or Arduino Uno was guided by practical scalability and reliability requirements. NodeMCU, although strong in built-in Wi-Fi capabilities, has a very limited number of GPIO pins, restricting the number of directly connected devices. Arduino Uno, while reliable and widely used, offers fewer pins and less memory, making it unsuitable for managing multiple appliances and sensors simultaneously without external expanders.[5] In contrast, the Arduino Mega accommodates a large number of peripherals directly, avoiding the need for complex expansion circuits and simplifying the overall hardware architecture.

In the proposed IoT-based smart lighting and appliance control system, the Mega serves as the central control unit, managing all switching operations and processing sensor data with high stability. A LAN shield is connected to the Arduino Mega, enabling a stable wired internet connection for reliable communication with the server. Through this connection, Arduino Mega connect to MQTT Server (MQTT protocol). Arduino Mega is a subscriber to MQTT Server, subscribing to a specific topic to receive control commands and status updates in real time. This tight integration between hardware and communication protocol ensures that even if network conditions fluctuate, the automation logic continues to function without interruption, making the system both dependable and scalable. The 16 Relay Module is connected with the Arduino Mega to control the appliances (Fig 1).

MQTT Protocol and Publish–Subscribe Model

The Message Queuing Telemetry Transport (MQTT) protocol is a lightweight, reliable, and cost-effective communication standard designed specifically for Internet of Things (IoT) applications.[3,6] In the IoT ecosystem, where devices often operate with limited processing power, memory, and bandwidth, MQTT shines because it can function with minimal resources and without demanding complex hardware. Running over TCP/IP, it enables seamless communication between devices and backend services, making it ideal for transporting telemetry data such as sensor readings and actuator commands. Its low overhead and event-driven nature make it suitable even for small devices like Arduino boards, yet powerful enough for large-scale IoT networks. At the heart of MQTT lies publish–subscribe (pub/sub) communication model, which replaces the direct device-to-device connection with a flexible and decoupled structure.[6] In this model, a publisher sends messages on specific topics to a broker, which acts like a central switchboard. Subscribers register their interest in certain topics, and the broker ensures that only relevant messages reach them. This approach offers three major advantages: it decouples senders and receivers so they can operate independently, it supports one-to-many communication where a single message can be sent to multiple subscribers at once, and it allows the subscriber list to change dynamically depending on the context. If no subscriber matches a message's topic, the broker simply discards it; if one or more subscribers match, each receives the message instantly. For Every device status change the application generates the string message, the one which will be published to particular topic. Later the message is decoded by the Topic subscriber, the Arduino Mega. Figure 2 illustrates the basic publish–subscribe model of MQTT.

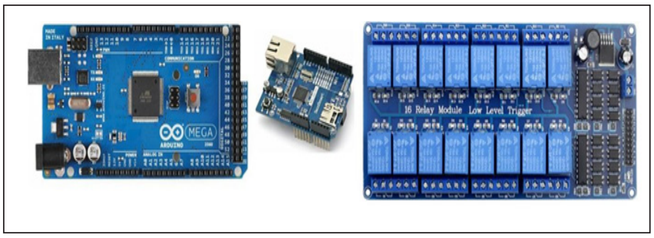

Figure 1.Arduino Mega 2560, Ethernet Shield and 16 Relay Module

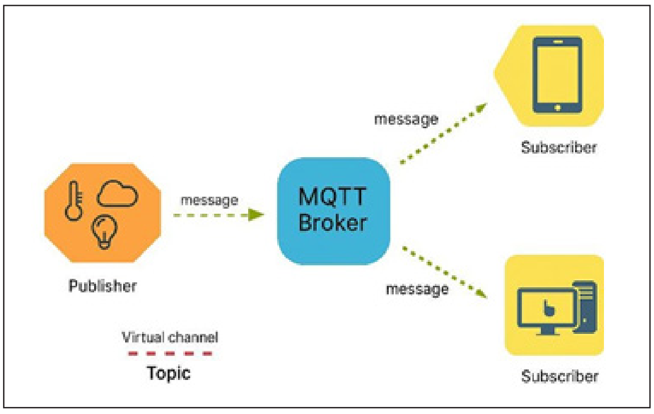

Figure 2.Publish- Subscribe Model of MQTT Protocol

In the proposed IoT-based smart lighting and appliance control system, MQTT is the backbone of communication. The Arduino Mega, equipped with a LAN shield for internet connectivity, subscribes to specific topics on a selfhosted

ISSN: 3051-4282
DOI: https://doi.org/10.24321/3051.4282.202502

MQTT server. This setup ensures secure, low-latency, and event-driven control of devices such as lights, fans, and air conditioners. By leveraging the publish–subscribe model, commands from the application backend are efficiently routed through the broker to the Arduino Mega, enabling real-time, reliable, and scalable automation without requiring the devices to remain online simultaneously. This architecture not only enhances performance but also adds an extra layer of security by avoiding reliance on third-party MQTT services.

Architecture Of The Application

The architecture of the proposed IoT-based Smart Lighting and Appliance Control System is designed to provide seamless interaction between the hardware, communication protocols, and the user interface.[7,8,9] At its heart lies the Arduino Mega, acting as the central controller that coordinates the flow of data between sensors, actuators, and the network. Connected to the Arduino Mega is a LAN shield, which enables internet connectivity. Through this link, the system communicates with an MQTT broker, allowing messages to be exchanged efficiently between devices and the web application (Fig 3).

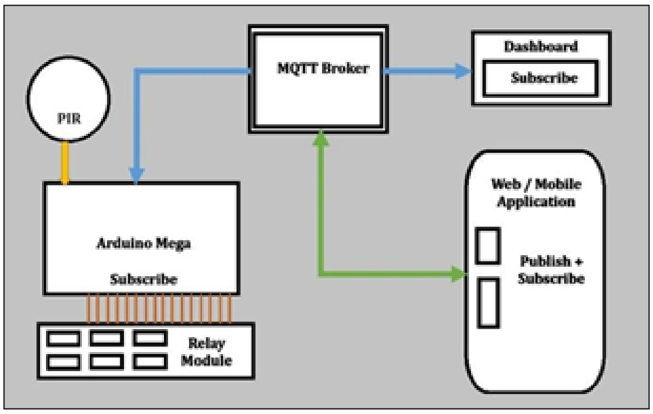

Figure 3.Architecture of the Smart Lighting and Appliance control system

The hardware layer comprises sensors for detecting environmental parameters and a relay module for controlling electrical appliances. The relay module acts as a switch, receiving commands from the Arduino Mega to turn devices on or off based on the user's instructions or automated logic. Eight PIR sensors used to detect the motion inside the Room, PIR sensor directly connected to the Arduino Mega for the detection of motion. Sensor Mode is used to save the power inside the room. If no motion found within 5 minutes, system enters into power saving mode with turn off all fans and lights. Using the publish–subscribe model, the web application (publisher) sends control messages to specific topics via the MQTT broker, and the Arduino Mega (subscriber) listens to these topics, instantly acting on any relevant commands to operate the relay and manage connected appliances. This decoupled communication ensures that messages are delivered reliably and efficiently, even if the sender and receiver are not active simultaneously.

On the software side, the web application serves as the primary user interface. Users can log in through a PHP-based authentication system, while Angular handles the dynamic and interactive elements of the dashboard. The application also supports QR codes placed on switchboards, enabling users to scan and directly trigger appliance control without opening the full interface. The web application then publishes these QR-triggered commands, alongside dashboard interactions, to the MQTT broker—ensuring real-time communication between the user and the devices. This architecture ensures a responsive, flexible, and scalable control system, making it easy to integrate additional devices and features in the future.

Web And Mobile Application

The web application serves as the primary user interface for the IoT-based smart lighting and appliance control system[10,11] providing a seamless way to monitor and operate connected devices from anywhere with internet access. Its design focuses on simplicity, responsiveness, and security, ensuring that even non-technical users can easily control appliances such as lights, fans, and air conditioners. The application offers an intuitive dashboard where each appliance is represented by a virtual switch, enabling instant ON/OFF control. To make interaction even more convenient, the system also supports QR codes placed on switchboards—by simply scanning a code, users can directly access and control the corresponding appliance without navigating through menus.[12] This interactive interface not only enhances user experience but also provides real-time feedback, allowing users to monitor the status of their devices without delay. Mobile Application is developed with the flutter and web view is imported (fig 4,5).

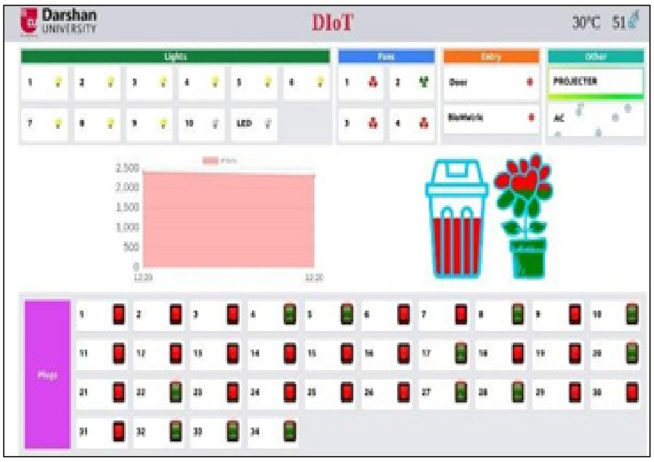

Figure 4.Web Dashboard

ISSN: 3051-4282
DOI: https://doi.org/10.24321/3051.4282.202502

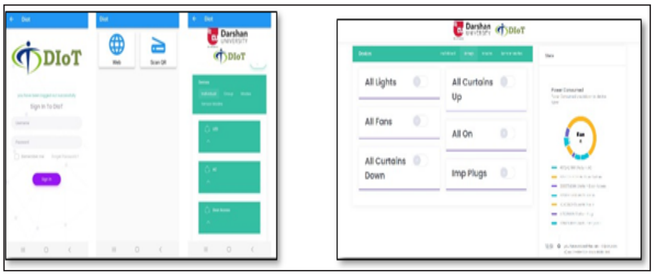

Figure 5.Mobile View and Web view of Control Panel

The web application combines multiple technologies to achieve both efficiency and scalability. PHP is utilized for managing the core login authentication process, ensuring that only authorized users can access the system. Once the credentials are verified, control is handed over to an Angular-based dashboard, which offers a dynamic, single-page interface for device control. The Node.js backend bridges the communication between the dashboard and the MQTT server, sending user commands securely to the Arduino Mega via the publish-subscribe model. In this architecture, the MQTT server acts as a message broker, relaying control commands to subscribed devices and returning status updates to the web dashboard.

Choosing a web application for the initial implementation offered several advantages over a dedicated mobile application. It provides platform independence, allowing access from any device with a browser, without the need for installations or updates. Furthermore, the integration of PHP, Angular, and Node.js ensures that the system remains modular, making it easy to upgrade individual components in the future. By combining robust backend authentication, a responsive frontend, and real-time IoT communication, the web application becomes a powerful yet user-friendly control hub for the entire smart lighting and appliance system.

In the Control Panel several features has been added to enhance the User experience. Group mode made possible to turn on/off all same type of the devices at the same time. Theater Mode has the feature to deal with particular devices for Low light and low sound theater like environment, which turn off regular lights and turn on warm light, pull curtains down, turn on the projector and A.C.

Future Scope

A multilingual voice assistant could take this system beyond touch and clicks, enabling natural, language-free barriers control, while AI predictions make your home respond before you even ask. Accurate power consumption measurement can be done through the current sensor.

Conclusion

The proposed system integrates Arduino Mega, MQTT, and a web dashboard to deliver seamless appliance control with real-time responsiveness. While the relay module guarantees precise operation of devices, the publish–subscribe communication ensures smooth interaction between hardware and users. Overall, the architecture not only demonstrates efficiency and flexibility but also lays a strong foundation for future enhancements, making smart living both practical and accessible.

References

1. Alsamhi SH, Ma O, Ansari MS, Meng Q. Greening internet of things for smart everythings with a green-environment life: A survey and future prospects. arXiv preprint arXiv:1805.00844. 2018 May 2.
2. Al-Fuqaha A, Guizani M, Mohammadi M, Aledhari M, Ayyash M. Internet of things: A survey on enabling technologies, protocols, and applications. IEEE communications surveys & tutorials. 2015 Jun 15;17(4):2347-76.
3. Jaswal K, Choudhury T, Chhokar RL, Singh SR. Securing the Internet of Things: A proposed framework. In2017 International conference on computing, communication and automation (ICCCA) 2017 May 5 (pp. 1277-1281). IEEE.
4. Gubbi J, Buyya R, Marusic S, Palaniswami M. Internet of Things (IoT): A vision, architectural elements, and future directions. Future generation computer systems. 2013 Sep 1;29(7):1645-60.
5. Ye X, Huang J. A framework for cloud-based smart home. InProceedings of 2011 international conference on computer science and network technology 2011 Dec 24 (Vol. 2, pp. 894-897). IEEE.
6. Vyas DJ, Rudani NN. MQTT & IOT Based Control and Monitoring of Smart Green House. International journal of Engineering development and research. 2018 Jan;6(1):235-40.
7. Pescatore J, Shpantzer G. Securing the internet of things survey. SANS Institute. 2014 Jan:1-22.
8. Vishwakarma SK, Upadhyaya P, Kumari B, Mishra AK. Smart energy efficient home automation system using IoT. In2019 4th international conference on internet of things: Smart innovation and usages (IoT-SIU) 2019 Apr 18 (pp. 1-4). Ieee.
9. Singh H, Pallagani V, Khandelwal V, Venkanna U. IoT based smart home automation system using sensor node. In2018 4th International Conference on Recent Advances in Information Technology (RAIT) 2018 Mar 15 (pp. 1-5). IEEE.
10. Yassein MB, Shatnawi MQ, Aljwarneh S, Al-Hatmi R. Internet of Things: Survey and open issues of MQTT protocol. In2017 international conference on engineering & MIS (ICEMIS) 2017 May 8 (pp. 1-6). Ieee.
11. Razzaque MA, Milojevic-Jevric M, Palade A, Clarke S. Middleware for internet of things: a survey. IEEE Internet of things journal. 2015 Nov 9;3(1):70-95.
12. Ashton K. That 'internet of things' thing. RFID journal. 2009 Jun 22;22(7):97-114.