

# Multi-Control Smart Office Automation System: Arduino IoT Cloud App, Web Browser, IR Remote and Manual Switches

Pranali P. Pisal<sup>1</sup>, Pratik P. Pisal<sup>2</sup>, Suraj P. Bagal<sup>3</sup>, B.B. Godbole<sup>4,\*</sup>

## Abstract

*This paper presents the design and development of a smart office automation system that can be controlled in multiple ways: using the Arduino IoT Cloud or a web browser, an IR remote, and on/off switches. The system allows users to control office devices such as lights, fans, and other electric appliances. In this system, six relay modules are used to operate six electric appliances. Of these six appliances, four are operated through a dashboard created in the Arduino IoT Cloud mobile app, which also displays temperature sensor data. These four appliances are also operated using an IR remote and manual switches. The remaining two appliances are operated using a remote and manual switches. A temperature sensor is included to provide real-time office temperature readings. The system can be operated from anywhere via the internet using a mobile app; or, if the mobile app is not available, it can also be accessed and controlled through a web browser. Additionally, users can control the system offline using the IR remote (effective up to 8 meters) or direct switches operated manually, making it versatile and internet-independent when needed. The system features fast response times: about 400 ms–1 sec for cloud or web control, and 200 ms–300 ms for IR remote and manual switches. An Arduino board manages all inputs and devices to ensure smooth and efficient functioning. This project offers an easy, fast, flexible, and affordable solution for automating efficient workplace environments.*

**Keywords:** Multi-Control, office automation, Node MCU ESP32, IR remote, Arduino IoT Cloud app, web browser, relay module, DHT22 sensor, TSOP receiver, ON/OFF switches

## INTRODUCTION

Today, smart office systems help offices work faster and more smoothly, helping to save time and reduce energy use. Most automation systems use only one method to control devices, such as Wi-Fi-based apps or on/off switches. However, relying on only one method can cause problems if the internet is down, if remote control failures occur, or if users need quick local access. To address these issues, this paper presents a Multi-Control Smart Office Automation System that combines four control methods: the Arduino IoT Cloud App, Web Browser Control (if the mobile app is not available), IR Remote Control, and On/Off Switches. With this system, users can manage office appliances from anywhere using their smartphones or web browsers, control them locally with an IR remote, or operate them directly using on/off switches. This multi-control setup makes the system more flexible,

### \*Author for Correspondence

B.B. Godbole

E-mail: bhalachandra.godbole@sknscoe.ac.in

<sup>1-3</sup>Student, Department of Electronics and Telecommunication Engineering, SKN Sinhgad College of Engineering, Korti, Pandharpur, Maharashtra, India

<sup>4</sup>Professor, Department of Electronics and Telecommunication Engineering, SKN Sinhgad College of Engineering, Korti, Pandharpur, Maharashtra, India

Received Date: June 14, 2025

Accepted Date: July 10, 2025

Published Date: October 06, 2025

**Citation:** Pranali P. Pisal, Pratik P. Pisal, Suraj P. Bagal, B.B. Godbole. Multi-Control Smart Office Automation System: Arduino IoT Cloud App, Web Browser, IR Remote and Manual Switches. Journal of Microcontroller Engineering and Applications. 2025; 12(3): 1–10p.

reliable, and easy to use because users can still turn off devices when not needed, even without internet access. The aim of this research is to build a smart office system that ensures continuous device control under all conditions and provides a better user experience compared to existing single-method automation systems [1, 2].

## LITERATURE SURVEY

The literature review section in the paper discusses the limitations of existing office automation systems that rely on single control methods. It points out that Wi-Fi-based systems fail when the internet is down, GSM-based systems are slow and costly, and Raspberry Pi systems are complex, costly, and expensive. To address these limitations, the paper proposes a multi-control smart office automation system.

Abubakar et al. (2021) [1] described an IoT office automation system that used only Wi-Fi to control appliances through a website. If the internet failed, control was lost. The proposed system is more reliable because it has backup methods like IR remote and manual switches.

Kute et al. (2021) [2] utilized NodeMCU with Blynk but faced issues if Wi-Fi failed. The proposed system improves on this with IR backup and offline manual control.

Divyashree et al. (2020) [3] and Pandey et al. (2018) [4] also described the importance of energy saving and reliability. The proposed system is better because users have more control options, preventing unwanted use.

Another system [4] (Wi-Fi and Arduino UNO) used only Wi-Fi control through a mobile app. It worked well when the internet was available but failed if the Wi-Fi was disconnected. In contrast, the proposed system allows users to control devices even without Wi-Fi by using an IR remote or manual switches, making it more reliable.

A GSM-based system [5] used SMS and motion sensors to monitor office appliances. Although GSM provides wide coverage, it is slower because of SMS delays and requires ongoing costs for mobile network services. The proposed system is faster and does not require extra communication charges after setup.

Another system used Raspberry Pi and IoT technology for real-time control and monitoring. However, Raspberry Pi is more costly and complex compared to Arduino-based solutions. The proposed system achieves similar smart features using ESP32, an IR remote, and switches at a lower cost and with simpler hardware.

Arduino IoT Cloud Remote App, <https://cloud.arduino.cc>.

Technical Data Sheet Infrared Remote-control ReceiverModule:[https://www.mouser.com/datasheet/2/143/evers01884\\_1-2266839.pdf](https://www.mouser.com/datasheet/2/143/evers01884_1-2266839.pdf)

## DESIGN AND IMPLIMENTATION

Figure 1 illustrates the proposed block diagram of the office automation system. The block diagram consists of a NodeMCU ESP32, a temperature sensor (DHT22), a TSOP receiver, a relay, a mobile phone with the Arduino IoT Cloud app, a 12V DC power adapter, and a remote. When the connection is established, the system will begin operating and reading the temperature sensor data. The sensor data is sent to the web server application. The data can be analyzed anywhere, anytime [6]. The office automation system can control the lights and fans (on or off), operate other office parameters, and monitor the office temperature.

A 12V DC adapter is used to power the smart office automation system. It supplies enough voltage and current for the relay module and other components. It provides a stable power source for reliable system operation.

### Node MCU ESP32

In this project, the NodeMCU ESP32 (Figure 2) serves as the main controller for the smart office automation system. The ESP32 provides Wi-Fi for connecting to the Arduino IoT Cloud, enabling remote control of office devices. The board has multiple GPIO pins to connect the IR receiver, relays, and manual switches. It was programmed using the Arduino IDE, which made development easier [7]. The board runs on 3.3V, powered through a USB cable connected to a power adapter. Its built-in Wi-Fi, multiple I/O pins, and fast processing speed made it ideal for combining all control methods into one device.

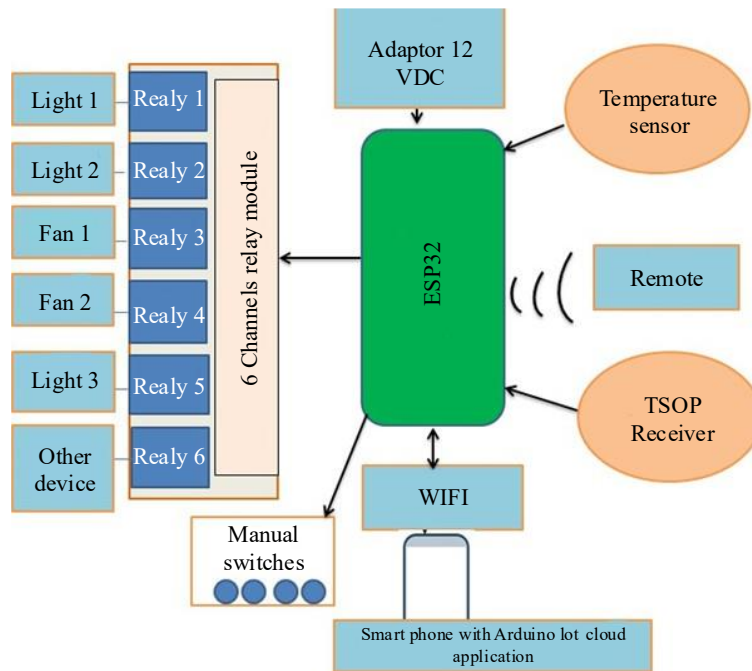


Figure 1. Block diagram of proposed system power supply.



Figure 2. Node MCU ESP32.

### Relay Module

The six-channel relay module (Figure 3) is used to control up to six different electrical devices, such as lights, fans, and other appliances. It supports switching up to 250V AC at 10A or 30V DC at 10A, making it suitable for common office appliances. The module is compact and allows multiple devices to be controlled at the same time from one board [8].

### IR Remote

An IR (Infrared) remote control (Figure 4) was used as one of the input methods to control appliances wirelessly. It works by sending infrared light signals to an IR receiver module connected to the NodeMCU ESP32. The remote operates at a standard frequency of 38 kHz. The IR remote transmitter

requires direct line-of-sight to the IR receiver, and its effective range is typically up to 8 meters. The IR system is reliable, lightweight, and simple to integrate with the automation system [9].



**Figure 3.** 6 Channel relay modules.



**Figure 4.** IR Remote and TSOP receiver.

### Arduino IoT Cloud App

The Arduino IoT Cloud is used in this project to enable remote monitoring and control of office devices through the internet. It allows the NodeMCU ESP32 to connect to a web dashboard or mobile app, where users can control appliances like lights and fans from anywhere. The Arduino IoT Cloud platform provides a simple interface to create virtual switches, buttons, and data displays without complex coding [10]. The ESP32 is programmed using the Arduino IDE with IoT Cloud libraries, making the setup easy. This integration helps automate tasks and improves convenience by allowing wireless control through Android phones or computers/PCs.

### METHODOLOGY

The proposed smart office automation system (Figure 5) is designed to control various office appliances through multiple controls: the Arduino IoT Cloud mobile application, an IR remote control, and manual switches. The NodeMCU ESP32 microcontroller serves as the central processing unit, handling all inputs and controlling the appliances through relays [10].

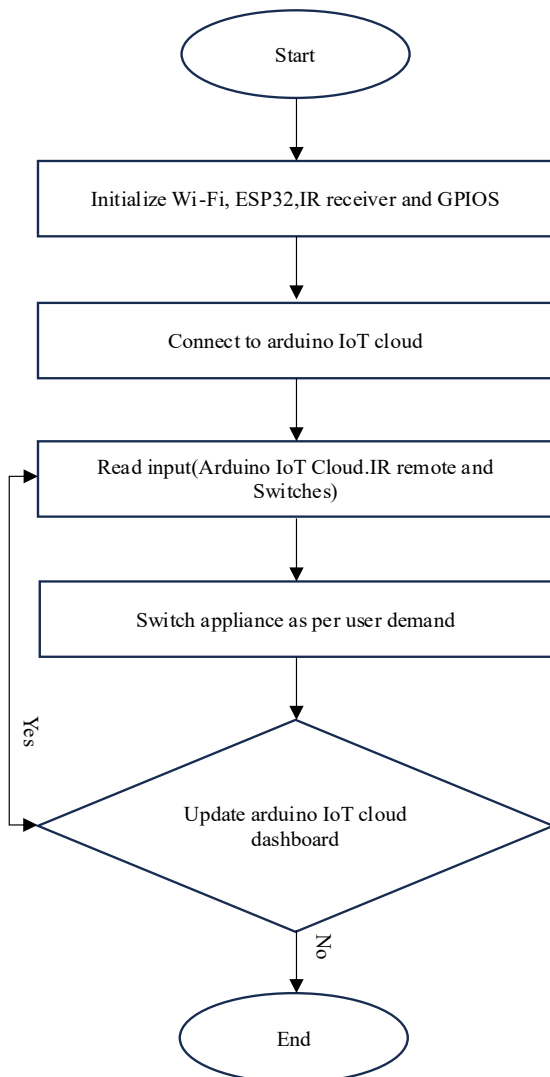
#### Control via Arduino IoT Cloud App

The Arduino IoT Cloud is used to create a custom dashboard accessible via the mobile application or web browser (Figure 6). This dashboard includes control widgets, such as buttons and data displays. When the user interacts with the dashboard, the app sends commands over the internet to the ESP32, which is connected to the Wi-Fi network. Upon receiving these commands, the ESP32 processes them and activates the relay to control appliances like lights, fans, and other appliances. In addition, a DHT22 temperature sensor is used to monitor the ambient temperature, where it is displayed on the dashboard for real-time monitoring.

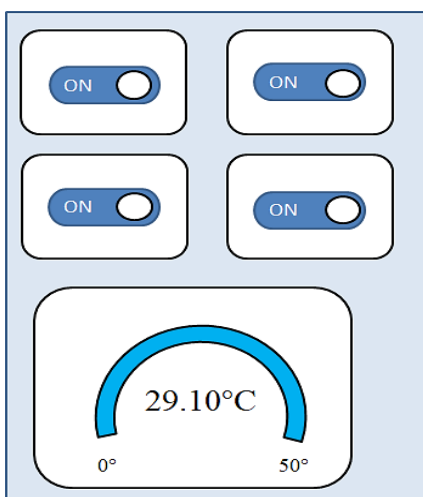
#### Control via Web Browser

The Arduino IoT Cloud is used to create a custom dashboard accessible via the mobile application or web browser. This dashboard includes control widgets, such as buttons and data displays. When the user interacts with the dashboard, either through the mobile app or, if the app is not available, through

a web browser, the system sends commands over the internet to the ESP32, which is connected to the Wi-Fi network [11].



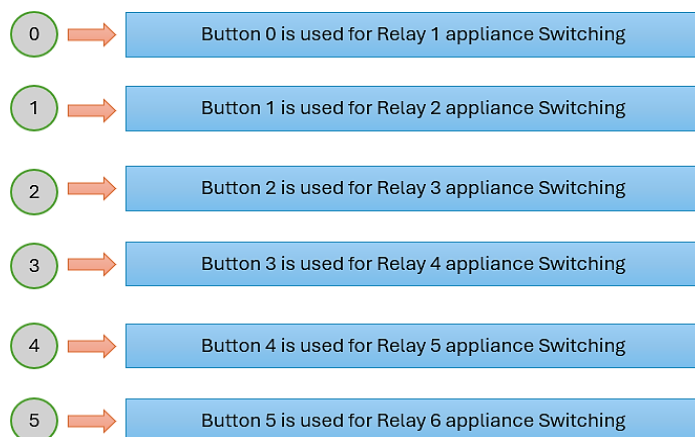
**Figure 5.** System operation.



**Figure 6.** Arduino IoT cloud dashboard.

### Control via IR Remote

The system also supports control through an IR remote. An IR receiver is connected to the ESP32, which listens for specific IR codes sent from the remote. Each button on the remote is assigned a unique IR code. When a button is pressed, the IR receiver on the ESP32 detects the code and triggers the corresponding action, such as turning a light or fan on or off. This method provides short-range control and does not require internet connectivity, making it useful in situations where internet access is unavailable (Figure 7) [12].



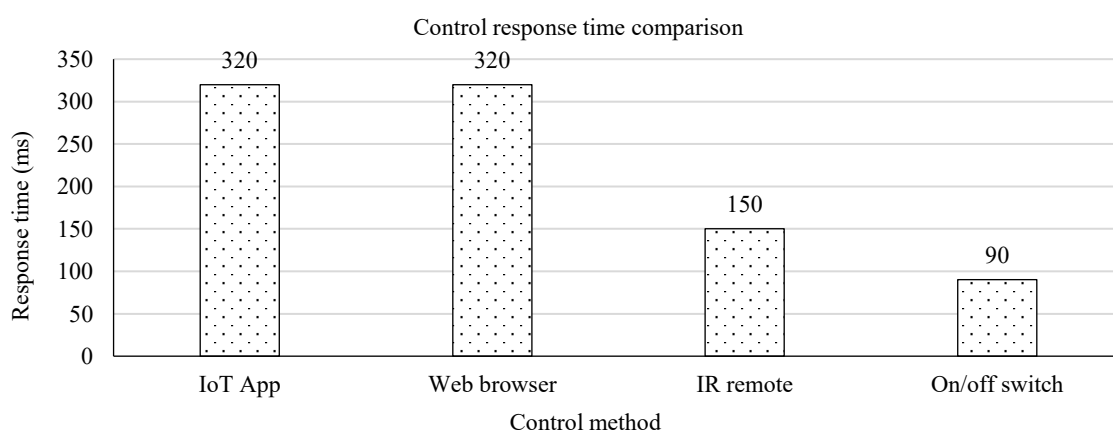
**Figure 7.** IR remote buttons.

### Control via Manual Switches

For local, physical control, the system is equipped with manual switches that are connected to the GPIO pins of the ESP32. Each switch corresponds to a specific appliance. When a switch is toggled, the ESP32 detects the change in input state (HIGH or LOW) and executes the corresponding logic to operate the associated relay, thus controlling the connected appliance [14]. This method ensures immediate, local control of devices without reliance on the internet or other external systems.

### Control Response Time Measurements

Comparison of average response times across different control methods (Figure 8).



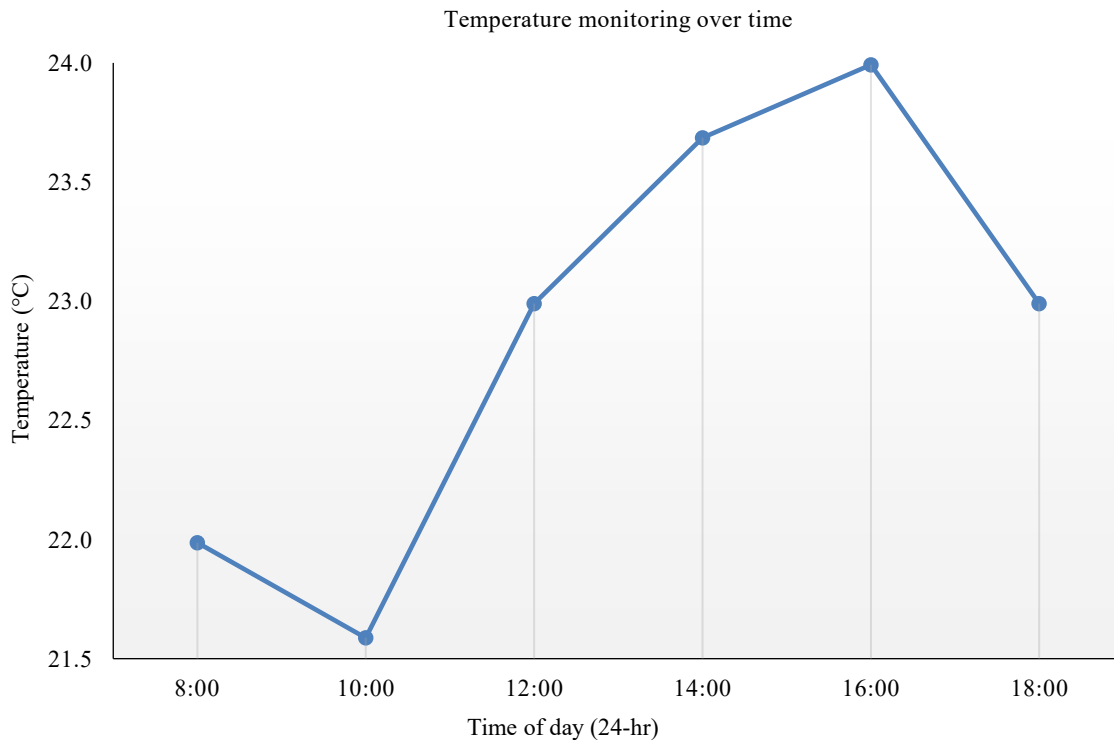
**Figure 8.** Control response time comparison graph.

Table 1 shows how fast different control methods respond in a smart office setup. The manual switch reacts the quickest, around 80–150 ms, because it connects directly to the device. The IR remote is a bit slower, with a response time of 150–250 ms, as it uses short-distance signals. The Arduino IoT app or Web browser takes the most time, about 300 ms–1 sec, likely due to internet delays and app processing. Even though the response times vary, using multiple control options is better than using just one [15].

A multi-control system gives users more choices and flexibility, allowing them to control devices in the way that best fits their needs, whether they are nearby, want remote access, or prefer a fast manual option.

### Temperature Monitoring w.r.t. Time

Figure 9 shows how the updated temperature graph displays office temperature fluctuations at different times of the day, with increases and decreases. This pattern reflects natural variations in indoor conditions due to factors like weather, device usage, or room occupancy (Figure 9).



**Figure 9.** Temperature monitoring over time graph.

**Table 1.** Control response time.

Control method	Response time range (ms)	Experimental result
IoT App	~300 ms-1 sec	320
Web Browser	~300 ms-1 sec (same as IoT App, since both use cloud)	320
IR Remote	~150 ms-250ms	150
On/Off Switch	~90 ms-150ms	90

## RESULT AND DISCUSSION

Figure 10 shows a smart office automation system. An Android IoT cloud app on the phone displays a dashboard with virtual switches and temperature sensor data. This app communicates with an ESP32. LEDs on the breadboard likely indicate the status of controlled devices. A temperature sensor provides the environmental data shown on the app [16].

Figure 11 shows a smart office automation system controlled via an Android IoT cloud app and an IR remote and switches. The hand pressing a button on the IR remote indicates direct control of a device (likely through an IR transmitter connected to the ESP32). The remote control functionality is demonstrated by the user directly interacting with an IR remote to send commands to the system, which are then processed by the ESP32 to control connected devices [17–20].



**Figure 10.** System operate using Arduino IoT Cloud app dashboard.



**Figure 11.** System operate using IR remote.



**Figure 12.** System operate using switches.

Figure 12 shows the hand pressing a button on the manual switches demonstrates direct control of the system through input. The system also features on/off switches on the app, an IR remote, and LEDs indicating device status. Control via the manual switches is demonstrated by the user directly interacting with the switches to input commands, which are then processed by the ESP32 to trigger actions in the connected office devices.

## CONCLUSION

This study successfully developed and evaluated a multi-control smart office automation system using four integrated control methods: the Arduino IoT Cloud app, a locally hosted web server, an IR remote, and ON/OFF switches. Observational results demonstrated that the ON/OFF switches offered the fastest response time, ranging between 90–150 ms, due to direct hardware triggering. The IR remote showed reliable mid-range responsiveness at approximately 150–250 ms. The web server and Arduino IoT Cloud app, both dependent on Wi-Fi and processing overhead, responded within 300 ms–1 sec. Additionally, real-time temperature monitoring was successfully displayed across both the mobile dashboard and web server interface, offering insights into environmental conditions throughout the day. By incorporating both online and offline control options, the system ensures uninterrupted operation of essential office appliances regardless of network availability. The project demonstrates that integrating multiple control methods in a smart automation setup not only enhances user experience but also increases system robustness and operational flexibility, making it a practical solution for modern office environments. Our tests showed that the system works quickly, with the manual switch being the fastest, followed by the IR remote, and then the IoT app. Compared to other systems we studied that used only one control method, our system saves more energy and gives users more flexibility. Overall, this multi-control system is a good solution for making office automation easier, faster, and more dependable.

## REFERENCES

1. Abubakar IN, Ufot EG, Dodo UA, Okosun OE, Dodo MA, Babalola O, et al. IoT Office Automation. *Zaria J Electr Eng Technol.* 2021;10(1):1–5.
2. Kute UV, Darade S, Avhad V, Gavhane S. NodeMCU with Blynk for Smart Automation. *Int Res J Eng Technol.* 2021 Jan;8(1).
3. Divyashree, et al. Energy Management System for Smart Offices. *Int J Eng Res Technol.* 2020;2278–0181.
4. Pandey R. Review of IoT-Based Office Automation Systems. *JETIR.* 2018 Oct;5(10).
5. Abubakar IN, Ufot EG, Dodo UA, Okosun OE, Dodo MA, Babalola O. Internet of things (IoT) based office automation system. *Zaria J Electr Eng Technol.* 2021;10(1):1–5.
6. Pandey AK. Automation of office system based on Internet of Things (IoT): A Review. *JETIR.* 2018 Oct;5(10):JETIR06057.
7. Rodge PR, Prajapati J, Salve A, Sangle P. IoT based smart interactive office automation. *Int Res J Eng Technol.* 2017 Apr;3(4).
8. Sravani MBNCS, Geetha J. IoT based smart voice and app controlled office automation. *Dogo Rangsang Res J.* 2022 Dec;12(12):2347–7180.
9. Arduino IoT Cloud Remote App [Internet]. Available from: <https://cloud.arduino.cc>
10. Sahana HS, Sandeep R, Shweta R, Sowmya J, Krupa KS. Office automation system using Internet of Things. *Int Res J Eng Technol.* 2017 Jul;7(7).
11. Shweta KB, Deepthi MV. A survey on intelligent office automation system based on IoT. *IRE J.* 2022 Jul;6(1):ISSN 2456–8880.
12. Clark E, Brown R. Smart Home Automation for Elderly Care: A Review of Technologies and Applications. *J Ambient Intell Human Comput.* 2024.
13. AlHammadi A, AlZaabi A, AlMarzooqi B, AlNeyadi S, AlHashmi Z, Shatnawi M. Survey of IoT-based smart home approaches. In: 2019 Advances in Science and Engineering Technology International Conferences (ASET). IEEE; 2019. <https://doi.org/10.1109/icaset.2019.8714572>
14. Singh U, Ansari MA. Smart home automation system using Internet of Things. In: 2019 2nd Int Conf Power Energy, Environ Intell Control (PEEIC). IEEE; 2019. <https://doi.org/10.1109/peeic47157.2019.8976842>

- 
15. Aware A, Vaidya S, Ashture P, Gaiwal V. Home Automation using Cloud Network. *Int J Eng Res Gen Sci.* 2015;3.
  16. Mohammed ZKA, Ahmed ESA. Internet of Things Applications, Challenges and Related Future Technologies. 2017;126–148.
  17. Raj P, Raman AC. *The Internet of Things: Enabling Technologies, Platforms, and Use Cases.* Boca Raton: CRC Press, Taylor & Francis Group; 2017. ISBN: 978-1-4987-6128-4.
  18. Mohammed ZKA, Ahmed ESA. Internet of Things Applications, Challenges and Related Future Technologies. In: *Book Chapter.* 2017. p. 126–148.
  19. Zhang Y, Lee K. Design and Implementation of Smart Home System Based on IoT. *Mater Today Proc.* 2024 Apr;83:1211–6.
  20. Sharma A, Patel R. IoT-Based Smart Home Monitoring and Automation System Using Android Application. *Int J Eng Res Technol.* 2023 Feb;13(2):112–6.