BioGecko Vol 12 Issue 03 2023

ISSN NO: 2230-5807

IOT BASED SMART SECURITY AND SMART HOME AUTOMATION

M.CHINABABU[1], GANESH CHOUDHARY[2], K.YASWANTH REDDY[3], V.BHANUPRAKASH REDDY[4]
[1]Assistant Professor, Department of CSE, **Teegala Krishna Reddy Engineering College**, Hyderabad, Telangana, India.
[2] UG Scholars, Department of CSE, **Teegala Krishna Reddy Engineering College**,Hyderabad, Telangana, India.

ABSTRACT:
The main goals of this work are to manage home appliances and create a smart wireless home security system that leverages Wi-Fi for communication. Home automation is the focus of the project, which focuses on controlling lights and fans. We have created a low-cost Web of Points-based system that enables the early detection of house fires and gas leaks. By calculating temperature and gas concentration, we are recreating a situation in which we notice a rising risk of house fire in a kitchen atmosphere. In this job, we suggest a smart security entire lot system that combines a NodeMCU microcontroller with a variety of sensors, such as a DHT11, flame sensor, and gas detecting unit. The idea behind the Internet of Things (IoT) is the remote connection and monitoring of physical objects (things) via the Internet. This concept can be effectively applied to our home to make it smarter, safer, and more automated. This Internet of Things project focuses on creating a smart wireless home security system that alerts the owner over the web in the event of a break-in and activates an alarm system as well. Additionally, the same may be used for home automation by utilising the same set of sensors.

Key words: Node MCU microcontroller**,** DHT11, flame sensor, and gas sensor**.**

I INTRODUCTION

1.1 MOTIVATION

Wireless The two components of this work are home automation and also home security. The system's present design alerts the owner via voice calls made over the Internet if any type of human activity is detected close to his front door and may also sound an alarm at the customer's option. The system also has a provision for delivering direct communications to concerned security personnel in case of a critical scenario. Instead of activating the security alarm system, the user/owner can plan to open the door and turn on numerous inside appliances that are also connected to and controlled by the microcontroller in the system to welcome his visitor if, on the other hand, the owner realises that the person entering his home is not a burglar but rather an unplanned visitor. The same can be done when the user enters the area personally, and thanks to the system, he can also make setups from his doorstep so that once he enters his home, he can make himself completely at ease without having to manually turn on the electricity or, for example, his favourite TV channel. Therefore, the dual issues of home security and dwelling automation may be addressed on a complimentary basis by using the same collection of sensors. The user may check the alerts and status of the IoT system from any location, even if Web connectivity is not readily available (considering that just having access to Wi-Fi is necessary; having a mobile phone linked to the internet is not required). The current infrared (IR) or Bluetooth remote controllers available on the market are still generally device-specific and cannot be used back-to-back. It is impossible to control Bluetooth-enabled mobile phones that are connected to electric appliances from a distance. Thus, using such devices would make it impossible to do tasks like turning on an air conditioner while coming home. On the other hand, our approach provides a simple, low-cost alternative to wired home automation and security systems. This work attempts to address the issue that existing home security and surveillance systems have with providing information about the situation to users who are not at home.
written works Survey

IOT-based smart home automation and safety and security Different types of cordless communication technologies, such as ZigBee, Wi-Fi, Bluetooth, GSM, etc., can be used with the Home Automation. Due to the short array in which they work, these current approaches have drawbacks. We will put this task—"IOT based Smart safety and Smart home automation"—into practise to overcome these drawbacks. The mission focuses on providing smart security by sending out a snapped photo with anail to the owner via the internet when an item is noticed as well as regulating lighting and fans, commonly known as home automation. We're going to carry out this task by using the "Node MCU" Component. Those who are handicapped or elderly would undoubtedly benefit more from this.HOME security and safety SYSTEM USING IOT Design and Implementation of an IoT-Based Smart Home Security System Integrated House Protection and Monitoring System based on IoTSmart wireless home protection system that is IoT based.

1. Bluetooth-based home automation system using mobile phones: In a Bluetooth-based home automation system, relays are used to connect household gadgets to an Arduino BT board at input and output ports. The Arduino BT board's programme is built using the highly interactive C language of microcontrollers, and the connection is established using Bluetooth. Only authorised customers are granted access to the home appliances due to the password protection that is provided. For cordless communication, a Bluetooth link is created between an Arduino BT board and a phone. The Python script is used in this system and can be installed on any Symbian OS environment, making it portable. For receiving phone remarks that indicate the tool's status, one circuit is created and put into use.
2. A mobile phone-based Zigbee home automation system
The system was developed and also implemented using Zigbee to monitor and manage the home appliances. Network planners track and evaluate the tool's performance. For this, a simple cordless ADSL contemporary router with 4 button ports—the Wi-Fi network—is employed. Both the network SSID and the WiFi security criteria have been setup. When the digital home algorithm declares a communication secure, it goes through a second encryption process before being sent on to the actual network device of the house. Zigbee controller transmitted signals to the end across the Zigbee network. the security, safety, and privacy of every communication that the virtual residence algorithm receives. Zigbee communication is used to reduce the cost of the system and the intrusiveness of the related installation of the system.
3. GSM-based home automation system utilising a mobile: Due to GSM and mobile advancement, GSM-based home automation is appealing for research. Twin tone multi regularity (DTMF) based home automation, GPRS-based home automation, and SMS-based home automation were the main solutions we considered for GSM communication. The work of A. Alheraish is rationally represented in number, which also illustrates how GSM and SIM (client identity module) are used by home sensors and devices to interface with the home network. The system makes use of a transducer to turn mechanical functions into electrical signals that are sent to the microcontroller. The system's sensor components convert physical characteristics like sound, temperature, and wetness into other numbers like voltage. The microcontroller analyses each signal and transforms it into a command that the GSM component can comprehend. Depending on the command that was received from the GSM component, choose the most appropriate communication method among text, GPRS, and DTFC.

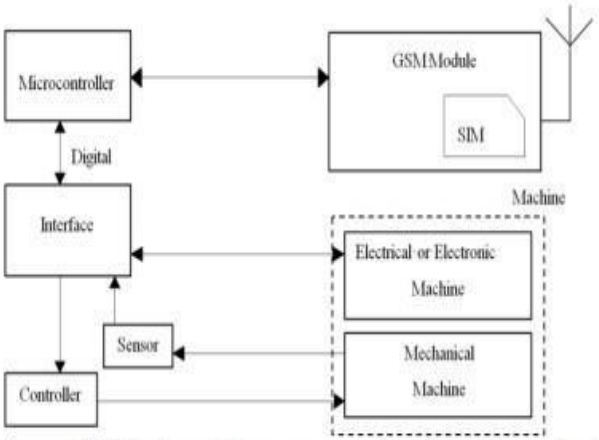


Figure. Mobile-based home automation from the work of A. Alheraish

Serial no.	System	Communication Interface	Controller	User Interface	Applications	Merits
1	Wi-Fi based using Arduino microcontroller through IOT	Wi-Fi	Arduino	Web Application and android App	Temperature and motion detection, monitoring and controlling appliances	Low cost, Secure, Remotely controlled
2	Smart Task Scheduling Based using Arduino and Android	Wired X10 and Wireless Zig bee	Arduino	Android Application	Energy Management and task scheduling with power and cost	Energy-efficient and Highly scalable
3	Web service and android app Based using Raspberry pi	Web server and interface card	Raspberry pi	Android application	Controlling shutter of window	Autonomous, and Quite scalable
4	Cloud Based Using Hadoop System	Cloud based data server uses Hadoop technology	Home gateway and router	Smart device	Monitoring and Controlling Home Appliances	Effectively manage Semi structured and unstructured data, Reduce computational burden of smart devices
5	Cloud Based Using Zig Bee Microcontroller	Zig bee wireless Network	Smart Socket	PC or Android Phone	entrance control management, monitoring the power consumption, temperature and humidity	Convenience, safety, and Power-saving
6	Wireless Sensors Based with mobile Technology	cloud-based data server	PCB circuits	Mobile Application	monitor the home conditions and power consumption of appliance	Low power consumption And system cost efficiency.
7	Android based using Arduino	Micro Web Server	Arduino Mega 2560 and the Arduino Ethernet shield	Android App	Light switches, Temperature, Humidity sensors, Intrusion detection, Smoke/Gas sensor	Feasibility and Effectiveness

II EXISTING SYSTEM

Currently, ZigBee-based Wireless House Automation System, Voice Recognition-based House Automation System, and Bluetooth-based Wireless House Automation System are used.One of the main drawbacks of a Bluetooth-based home automation system is that its limited range of 10 metres prevents it from controlling appliances in the house if the smartphone runs out of range.Domoticz is a Raspberry Pi-based home automation solution. But the cost is enormous.

Proposed System's Values

The proposed system uses the DHT11 sensing unit to detect changes in temperature and humidity. It then sends the data to the NodeMCU, which processes it and activates a follower to regulate the temperature and humidity levels. The system also employs a flame sensor to detect any potential fires of any kind, sounding the alarm and showing the alert message on the attached display screen.It is very dependable and offers real-time monitoring and control of temperature, humidity, fire, and gas. It can be easily linked into any security system and managed online from a different location. The system may also be modified and expanded to include additional sensors and components to meet certain security requirements.

DISAVANTAGES & OPPORTUNITIES

Disadvantages

Vulnerabilities in terms of safety: IoT devices are susceptible to security breaches and hacking. These devices have the potential to become access points for cybercriminals to access your home network or endanger your privacy if they are not adequately protected.

Personal privacy issues: Smart home tools frequently gather a lot of information about your routines, habits, and personal preferences. This data has the potential to be tampered with or accessed by unauthorised parties, causing privacy issues.

Dependability issues: For an Internet of Things device to work effectively, there must be a constant online connection. Your smart security and home automation devices' reliability and performance may be impacted if your internet connection is interrupted or fails.

Benefits

- This low-cost solution takes care of home automation and home security with only a few prerequisites.
- This home security system employs numbers from the phone's keypad instead of a smartphone application or other type of user interface; the system is system
- autonomous and hence accessible from a variety of phones running different operating systems.
- The optional mobile application takes into account the possibility that the consumer could also want to control his household appliances without setting off detecting devices.

PROPOSED SYSTEM

The proposed system uses the DHT11 sensor to detect changes in temperature and humidity. It then sends the information to the NodeMCU, which processes it and activates a follower to control the temperature and humidity levels. The system also employs a flame sensor to detect any possible fires, activating the alarm, and displaying the alert message on the attached display screen.It offers real-time tracking and management of temperature, moisture, fire, and gas and is very effective. It can be simply integrated into any security system and is also web-based remote controllable. Additionally, the system may be modified and expanded to include different other sensing components and parts to meet specific security requirements.

GOAL:

IoT-based smart security and smart home automation systems are designed to provide convenience, improve safety and security, and also increase energy efficiency in residential settings.

Boost benefit: IoT devices can automate many home duties like regulating lighting, heating, and home entertainment systems, enabling people to easily maintain and also customise their living environment.

Boost safety and security: Smart safety systems may include components like activity detection units, door/window sensors, surveillance cameras, and smart locks that enable remote tracking. real-time warnings and control over who has access to the house. This increases home security and gives homeowners happiness.

TECHNIQUE IV

To implement a smart security and home automation system, start by identifying your unique needs and goals. Determine the areas of your home that require automation as well as the security features you want.

Research available technologies: Look at the many IoT devices, systems, and treatments that are currently on the market. To narrow the options that meet your needs, consider criteria including compatibility, protection features, integrity, as well as client reviews.

Create a system design plan that lists all the components and devices needed for your intelligent home automation and security system. Consider the need for connectivity, such as Wi-Fi or Bluetooth, as well as the potential for tool integration.

Choose and buy devices: Choose the IoT devices that best meet your needs based on your study and system design approach. This might include security cameras, motion sensors, smart locks, smart thermostats, lighting controls, and other automation devices. Consider features like integrity, protection, and also ease of use, as well as make sure the devices are compatible with one another.

EXECUTIONS AND RESULTS OF V

Implementing IoT-based smart security and home automation requires combining many tools and systems to create a connected environment. The steps to put in place such a system are as follows:

Define your requirements: To begin, decide the particular automation and security features you want in your house. Security cameras, activity sensors, door/window sensors, smart locks, smart lighting, thermostat control, etc. may fall under this category.

Choose IoT devices: Do some research and pick IoT devices that suit your requirements. Make sure the devices you choose can be integrated into a single system and that they can communicate with one another. Consider factors including device compatibility, connection protocols (Wi-Fi, Zigbee, Z-Wave, etc.), and user reviews.

Create a central hub: The central hub serves as the brain of your smart home system. It connects and controls all IoT devices, allowing them to interact with one another. Examples of well-known hubs are Google Nest Hub, Amazon Mirror Plus, and Samsung SmartThings.

Install sensors and tools: Install the chosen sensing components and tools in the appropriate locations. Area security cameras are strategically placed to cover vulnerable areas, movement sensors are placed close to entrances, and door/window sensors are mounted on accessible apertures. For installation and configuration, according to the producer's instructions.

Connect tools to the hub: Connect each device to the central hub in accordance with the manufacturer's instructions. Typically, this entails using a web-based interface or a mobile app to connect the tools to the hub. Ascertain that every device is securely connected and reachable from the hub. Create automation rules depending on your preferences. For instance, you may set the system up to automatically lock the doors at a specific moment or change the thermostat according on the number of occupants when motion is detected. Create these guidelines using the control user interface for the hub.

Enable remote access: Many smart home systems let you manage and keep an eye on your home from a distance. Connect your centre to your home's Wi-Fi network and create an account on the corresponding application or website to enable remote access. This gives you the ability to control both your smart safety and

Use a smartphone or computer to access your automation system from anywhere. Thoroughly test the system to ensure that all devices and automation rules work as intended. To improve the system's performance, make any necessary adjustments or improvements and take care of any problems that arise.

Prioritise security measures since IoT devices may be vulnerable to cyberattacks. Change the factory default passwords, keep your hardware and firmware up to date, and utilise secure Wi-Fi techniques. Consider activating two-factor verification as well as using a different network just for IoT devices.

Broaden and tailor: As you gain experience with your smart home automation and security system, you may expand its functionality by incorporating new devices or other systems. Configure the system to meet your unique requirements and preferences. Keep in mind to consult the user manuals and documentation provided by the manufacturers of your IoT devices for comprehensive instructions and detailed instructions on setup and configuration.

To implement an IoT-based smart safety and home automation system using C++, you'll need to take the following factors into account:

Gadgets in an IoT system must communicate with both the central hub and one another. Boost.Asio and POCO Net are only two of the many collections and network interface frameworks that C++ offers. These collections may be used to provide channels of communication both inside the hub and between tools.

Assimilation of Sensing Units: Depending on the devices and gadgets you select, you may need to include them into your C++ programme. The majority of IoT technologies offer APIs or SDKs in a variety of programming languages, including C++. These APIs allow you to interface with the sensing devices and obtain essential data from them.

Information processing: You must process and also analyse the data you get from the sensors. For processing and manipulating data, C++ provides strong collections like the Standard Design Template Collection (STL). The data from the sensing device may be stored, processed, and even examined within your application using STL containers and formulae.

Applying automation policies necessitates the use of a guideline engine, which analyses issues and initiates actions based on predetermined rules. By utilising object-oriented ideas, a rule engine may be designed and implemented in C++. Use a decision-making algorithm to analyse the rules and prompt the best actions after identifying policy items with issues and activities.

User Interface: You may create a graphical user interface (GUI) using C++ libraries like Qt or wxWidgets to provide a user interface for controlling and monitoring the smart security and home automation system. These collections include components and gadgets to create visually pleasing interfaces for users to interact with the system.

Safety and security considerations are essential for Internet of Things technologies. To ensure information privacy and stability, use secure communication protocols like Transport Layer Safety And Security (TLS). Adhere to the recommended practises for safe coding, such as input recognition, buffer overflow protection, and secure storing of sensitive information.

Checking and Debugging: To create unit tests for certain components of your C++ programme, use testing frameworks like Google Test or Catch2. Checking various situations and edge cases will ensure the system functions properly. Use debugging tools like gdb or integrated development environment (IDE) debuggers as well to find and also repair any errors.

Deployment and Integration: When releasing the system, take into account the hardware requirements as well as the target devices' compatibility. For some designs, you might need to cross-compile your C++ code. For further functionality and remote access, think about connecting your system with cloud services or third-party APIs.

Do not forget to consult the documentation and resources provided by the collections, structures, and APIs you select to employ in your C++ programme. Make sure you have a solid grasp of both the hardware and

software domains because IoT growth typically requires a combination of both. Asynchronous Shows: IoT systems often include managing several concurrent activities, such as gathering data from sensing units, processing events, and communicating with devices. To handle asynchronous programming, C++ provides collections like Boost.Asio or C++ 20's coroutines, allowing you to manage these jobs properly without disturbing the main thread.

Actuators like lights, locks, and thermostats may need to be controlled, depending on the tools you incorporate into your system. C++ can communicate with actuators using a variety of protocols, including MQTT, CoAP, and HTTP. Use command- and actuator-controlling libraries like Eclipse Paho MQTT or cpp-httplib to deliver messages to the actuators linked to your smart home system.

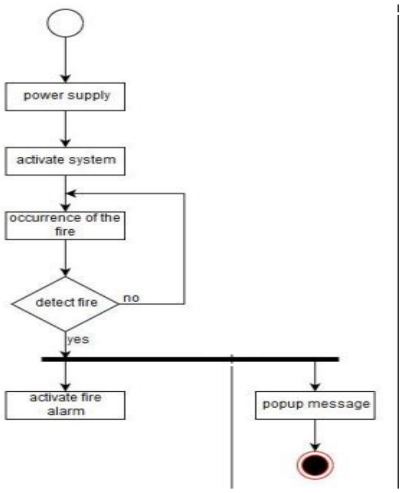
Data Analytics and Storage: Monitoring sensor data can yield valuable information for both automation and security goals. Consider storing and retrieving data obtained from sensing equipment using databases like SQLite or MySQL. For complex data processing and artificial intelligence algorithms, you may also use data analytics sets like TensorFlow or Apache Glow.

Combining AI and ML: AI and machine learning techniques can enhance the capabilities of your smart home automation and security system. You may do functions like item finding or facial recognition using C++'s libraries for computer system vision tasks, such as OpenCV. Additionally, you may use frameworks such as TensorFlow or PyTorch to integrate professional versions for cutting-edge analytics or anomaly detection.

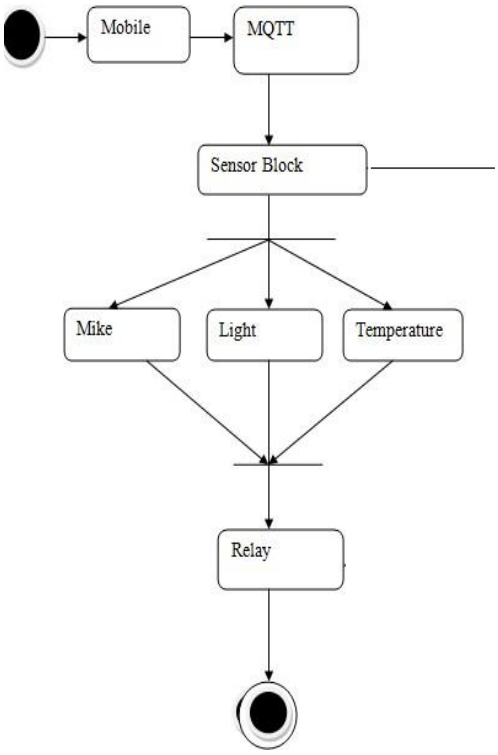
Real-Time Interaction: Consider adopting techniques like WebSockets or the Real-Time Publish-Subscribe (MQTT) protocol for scenarios that call for real-time interaction. These methods allow for dependable, low-latency communication between devices and the main hub, enabling real-time monitoring and management of your smart home system.

Implement energy-efficient ways to increase power consumption and also prolong battery life for IoT devices. Utilising sleep settings, planning tasks, or low-power cordless protocols like Zigbee or Bluetooth Low Energy (BLE) are some examples of how to do this. Consider collections like Contiki or TinyOS, which provide protocols and energy-efficient networking stacks. In other cases, you might need to develop firmware for IoT devices or microcontrollers that are integrated into your system. When developing firmware, C++ may be used to programme and communicate with hardware using frameworks like Arduino or ARM mbed.

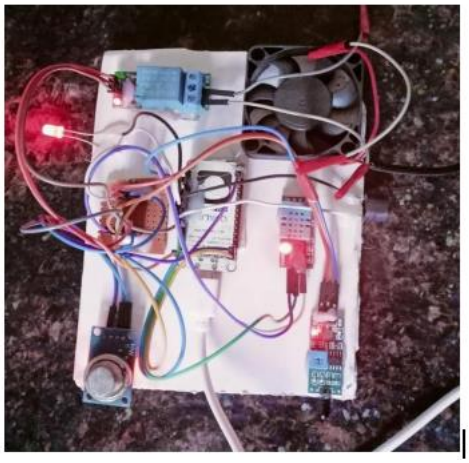
Strength and Error Handling: Implement robust error handling and healing systems in your C++ programmes to handle network outages, sensor malfunctions, or unforeseen occurrences. Use methods like handling exemptions, retries, or fallback mechanisms to ensure sure the system can gracefully recover from mistakes and continue operating. Remember to keep protective procedures in mind as you go with your project. This includes encrypting sensitive data, using secure verification and access control methods, and routinely upgrading software and firmware to address security vulnerabilities. IoT development is a broad field, and there are many frameworks, topologies, and platforms readily available for different use cases. Consider researching and selecting those that best fit your unique needs, equipment, and the ecological community.



Activity diagram for smart



Results :



Blynk. Console is a feature-rich online application that serves many user kinds. The platform's configuration of linked devices, including application settings, is one of its primary features. Management of devices, data, users, groups, and locations. Remote device monitoring and management Applications created using Blynk are now available to users. Whether they are members of your family, coworkers, or customers, they can quickly download the app, link their devices, and begin using it with your devices.

a little service that makes possible:
Device ownership claims made by users and organisations

supplying WiFi credentials to devices so they may connect to end-user WiFi networks.
Blynk.Apps's Managing Authentication Tokens UX flow will support end users through the claiming and provisioning processes.

CONCLUSION AND FUTURE ENHANCEMENT

To sum up, the IoT-based home monitoring system created in this project offers a trustworthy and dependable means of monitoring fire, gas levels, moisture, and temperature within a home. Homeowners receive fast signals and real-time information thanks to the integration of several sensor units, the NodeMCU ESP8266 microcontroller, and the BLYNK IoT application, which increases security and assurance. The DHT sensor device accurately measures temperature and moisture, enabling home owners to maintain a comfortable living environment and also take necessary actions to resolve any issues. The MQ2 gas sensing device in particular uses a gas sensor to detect the presence of LPG gas, shielding residents from potential gas leaks and associated

dangers.

The fire sensing unit is essential for the early detection of fires because it sends out alert signals and makes it possible for home owners to take the necessary precautions to stop the fire from spreading. Combining the relay, exhaust follower, LED, and buzzer improves the automation of the system and provides visible and audio warnings, ensuring prompt feedbacks to potential dangers. The main hub, the NodeMCU ESP8266 microcontroller, facilitates seamless communication between the sensing devices and the BLYNK IoT application. Through the programme, house owners can easily monitor the health of their residence from a distance, increasing access and control. The system's potential future range offers a number of opportunities for expansion and improvement.

Additional sensing devices, such carbon monoxide or movement detectors, might be included to offer a far more thorough monitoring service. Machine learning algorithms and other advanced information analysis approaches can help the system become more adept at spotting and predicting anomalies. The system may be integrated with current home automation systems to increase efficiency and convenience while providing automatic control of various home appliances depending on environmental conditions. The BLYNK IoT application may be improved to provide even more in-depth data visualisation, historical analysis, and notification customisation options.

REFERENCES

1.Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. IEEE Communications Surveys & Tutorials, 17(4), 2347-2376.

2.Bandyopadhyay, D., & Sen, J. (2011). Internet of things: Applications and challenges in technology and standardization. Wireless Personal Communications, 58(1), 49-69.

3.Roy, N., Mukherjee, A., & Chakraborty, U. (2020). IoT-based home automation: A comprehensive review. Computers, Materials & Continua, 63(2), 1111-1135.

4.Patil, S., & Shaikh, A. (2016). IoT based smart home automation and security system using Arduino and Wi-Fi. International Journal of Engineering and Computer Science, 5(9), 18082-18088.

5.Sharma, S., & Khanna, A. (2019). Internet of Things based home automation system: A review. International Journal of Computer Science and Mobile Computing, 8(6), 177-186.

6.Singh, P., & Gautam, P. (2018). Design and development of smart home automation and security system using Arduino and IoT. International Journal of Computer Applications, 181(7), 27-33.

7.Gupta, P., & Saini, M. L. (2019). Smart home automation system using IoT. International Journal of Recent Technology and Engineering, 8(2S4), 448-453.

8.Choudhury, N., Das, S., Biswas, A., & Gupta, A. (2020). IoT-based smart home automation system using NodeMCU. International Journal of Advanced Research in Computer Science, 11(1), 32-35.