



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

# IJRSET

International Journal of Innovative Research in  
**SCIENCE | ENGINEERING | TECHNOLOGY**

# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 12, Issue 5, May 2023

**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA

Impact Factor: 8.423

9940 572 462

6381 907 438

[ijrset@gmail.com](mailto:ijrset@gmail.com)

[www.ijrset.com](http://www.ijrset.com)

# IoT-enabled Military Operations: Progress in real-time Decision Making

Kalathma M.K<sup>1</sup>, Lavita Joyce<sup>2</sup>, Sampreeth M.P<sup>3</sup>, Kusmitha H.D<sup>4</sup>, Sharmila M.K<sup>5</sup>

<sup>1</sup>Assistant professor, Department of Computer science and Engineering, ATME College Of Engineering, Mysore, India

<sup>2,3,4,5</sup>UG student, Department of Computer science and Engineering, ATME College Of Engineering, Mysore, India

**ABSTRACT**-IOT-enabled Military operations: Progress in real-time Decision-making article intends to design an IoT-based military operating system. In order to increase situational awareness and facilitate field decision-making. The system may connect more sensors and components as necessary and is intended to offer real-time data on environmental conditions, possible threats, and troop movements. The process entailed employing a combination of hardware and software components to design and create the system. The ESP32 microcontroller was used to combine the system's different sensors and components, and Thingier.io's cloud platform will be used to see the data the sensors collect. Real-time visualization and analysis of the data gathered by the sensors are made possible by the platform in a safe and effective manner. The project's objectives include improving the safety and efficiency of military operations by providing real-time data on environmental conditions, potential threats, and troop movements. The results showed that the system was successful in collecting and transmitting real-time data on environmental conditions and potential threats.

**KEYWORDS**--Real-time, multi-purpose, decision making, Risk management, military operations.

## I. INTRODUCTION

Modern military operations require real-time situational awareness to make informed decisions and ensure the safety and effectiveness of troops. With the advent of the Internet of Things (IoT) technology, it is now possible to collect and transmit real-time data on environmental conditions, potential threats, and troop movements, which can greatly enhance the effectiveness of military operations. This project aims to develop an IoT-based military operation system that utilizes multiple sensors and components to improve situational awareness and decision-making in the field. The system incorporates various sensors, including air quality sensors, fire sensors, temperature and humidity sensors, light sensors, PIR sensors, and GPS sensors, along with an ESP32 microcontroller, a robot with a servo, Arduino Nano, motor driver, wheels with ESP32 cams, and a transmitter with a keypad, camera rotation, Arduino Nano, and joystick.

The system is designed to be flexible and adaptable to various environments and situations, with the ability to integrate additional sensors and components as needed. The project's objectives include improving the safety and efficiency of military operations by providing real-time data on environmental conditions, potential threats, and troop movements. The data collected by the system can be viewed and analyzed through the cloud platform, Thingier.io, which provides a secure and efficient way to visualize and analyze the data collected by the sensors in real-time. The platform's features, including data logging, dashboard creation, and device management, can be useful for monitoring and controlling the system remotely.

### A. Objectives

The objectives of the IOT enabled military operations: progress in real time decision making are as follows:

1. Integration of multiple sensors: The system should incorporate multiple sensors such as air quality sensors, fire sensors, temperature and humidity sensors, light sensors, PIR sensors, and GPS sensors, which can collect real-time data on environmental conditions, potential threats, and troop movements.
2. Integration of hardware components: The system should include hardware components such as an ESP32 microcontroller, a robot with a servo, Arduino Nano, motor driver, wheels with ESP32 cams, and a transmitter with a keypad, camera rotation, Arduino Nano, and joystick, which can process and transmit the data collected by the sensors.

3. Integration with cloud platform: The system should be integrated with a cloud platform such as Thingier.io, which can provide a secure and efficient way to visualize and analyze the data collected by the sensors in real-time.
4. Testing and evaluation: The system should be tested in a simulated military operation scenario to assess its effectiveness in providing real-time data and enhancing situational awareness.
5. Flexibility and adaptability: The system should be designed to be flexible and adaptable to various environments and situations, with the ability to integrate additional sensors and components as needed.

### *B. Problem Statement*

The requirement for better situational awareness and decision-making in military operations serves as the project's issue statement. In contemporary combat, troops work in intricate and dynamic contexts where they must contend with a variety of difficulties, such as possible threats, shifting troop movements, and environmental risks. Real-time information on environmental conditions, possible dangers, and troop movements is required in order to make wise decisions and guarantee the safety and effectiveness of soldiers. Traditional data collecting and processing techniques are frequently sluggish and ineffective, making it challenging to react rapidly to changing circumstances. By creating an Internet of Things (IoT)-based military operating system that combines several sensors and parts to gather and communicate real-time data on environmental conditions, possible threats, and troop movements, this project seeks to address this issue. The technology will give soldiers insightful information about their surroundings, empowering them to decide wisely and act fast in changing circumstances.

## **II. EXISTING SYSTEM**

The current system of military operations has human troops doing a variety of risky, physically demanding, and psychologically draining jobs on the battlefield. Human troops are limited in terms of stamina, quickness of decision-making, and quick detection and reaction to threats.

Human mistake can also result in undesirable outcomes like harm or fatality under the current system of military operations. Robotic soldiers have assisted soldiers in the past with jobs including bomb disposal, reconnaissance, and surveillance. These robots couldn't make their own judgements and were frequently commanded remotely.

Additionally, they had a limited range of motion and were not connected to IoT elements that may have improved their performance. The current military operating system is based on conventional weapons and technology, including guns, tanks, and drones. While these technologies can be useful in some circumstances, their capabilities can be constrained, and they frequently can't adapt to shifting battlefield circumstances.

A drone, for instance, may be helpful for monitoring, but it might not be able to recognize and address threats in real time.

In general, the current system of military operations is to some extent successful, but it has flaws that can be fixed by incorporating IoT technology into military robots. Military robots could be able to independently identify and respond to threats in real-time by adding IoT components like sensors, cameras, and GPS, as well as carry out activities that would be risky or impossible for human soldiers to complete. Additionally, military commanders might receive useful data and analytics from IoT-enabled military robots, enabling them to make better informed choices in real-time. The efficacy and safety of military operations might be greatly improved by integrating IoT technology with military robots, even if the current system of military operations has historically proven successful. However, as was covered in the preceding response, there may also be hazards and drawbacks to using military robots with IoT capabilities. To guarantee that these robots are employed in a safe, moral, and efficient manner, much thought and preparation must be made.

## **III. PROPOSED SYSTEM**

The proposed statement about an IoT military operations robot using the IoT components mentioned focuses on the potential future developments of these technologies in military operations. It suggests that the integration of these components into military robots can revolutionize the way in which military operations are conducted and can lead to new innovative solutions to current challenges.

The MQ 135 Air Quality Sensor, Fire Sensor, and DHT11 Temperature and Humidity Sensor can be integrated with machine learning algorithms to create a predictive model that can forecast environmental conditions, hazardous materials, and potential risks, providing early warning and allowing for timely responses. The IDR Module and ESP32 microcontroller can facilitate the integration of the robot with existing military networks, providing real-time information sharing and enabling a higher level of coordination between robotic systems and human operators. This can lead to more efficient decision-making processes and improved outcomes in military operations.

The PIR Sensor and NEO 06 GPS can be integrated with autonomous navigation systems, allowing the robot to navigate complex environments and autonomously patrol designated areas. This can enhance the safety of military personnel and increase the effectiveness of security operations. The ESP32 Camera and Servo Motor can be integrated with facial recognition and object detection algorithms, providing the robot with the ability to recognize and track targets, making it ideal for reconnaissance and surveillance missions. The L298 Motor Driver can be integrated with obstacle detection and avoidance systems, allowing the robot to move smoothly through complex environments and avoid obstacles. The NRF24LO1 RF module can be used to establish a mesh network of interconnected robotic systems, enabling more complex and coordinated actions, such as convoy operations or search and rescue missions. Finally, the Joystick can be used as a backup control mechanism in the event of a communication failure or when manual control is needed.

#### IV. METHODOLOGY

The use of IoT (Internet of Things) technology in military operations has the potential to significantly enhance situational awareness, decision-making, and operational efficiency. Here is a methodology for applying IoT in military operations:

1. Define the operational requirements: The first step is to determine which operational needs may be met by IoT technology. This may entail determining the kinds of sensors and devices needed, the kinds of data that must be gathered, and the kinds of insights needed.
2. Develop an IoT architecture: The next stage is to create an IoT architecture that can enable the deployment of sensors and devices after the requirements have been established. This may entail creating a network architecture, choosing suitable communication protocols, determining the kinds of gateways needed, and determining additional infrastructure requirements.
3. Select appropriate sensors and devices: The next stage is to choose the proper sensors and equipment to satisfy the operating needs. This might entail choosing sensors for tracking workers, monitoring the environment, and other uses.
4. Deploy sensors and devices: It is necessary to deploy the chosen sensors and gadgets in the field. This may entail placing them on people, cars, and other resources.
5. Collect and process data: Large amounts of data must be gathered and analyzed instantly by the sensors and equipment. This can entail processing data closer to the source using edge computing techniques.
6. Analyze and interpret data: To deliver useful insights, the data gathered from the sensors and devices has to be analyzed and interpreted. This could entail utilizing machine learning and other cutting-edge analytical methods.
7. Visualize insights: Once the insights have been produced, it is necessary to visualize them so that military people may readily understand them. This can entail creating dashboards and other visualizations that offer situational awareness in real time.
8. Take action: The final stage is to act in response to the insights the IoT system has produced. This might entail sending troops or resources to address a danger or taking other steps in accordance with the knowledge gained.

## V. SYSTEM DESIGN

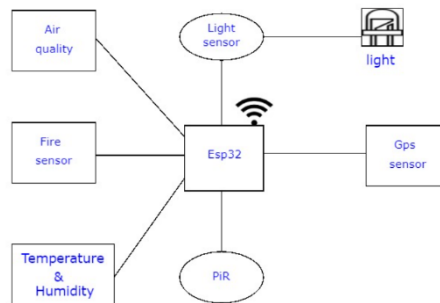
### A. Introduction

System design is the stage that effectively closes the gap between the issue domain and the current system. In this stage, the question "how to implement?" is the key focus. This stage determines how the system will function and converts the SRS document into an implementable format. The complicated activity of system development is broken down into a number of smaller sub-activities during this phase.

The process of establishing a system's architecture, parts, modules, interfaces, and data in order to meet predetermined criteria is known as system design. It entails drawing out a strategy or blueprint to build a system. System design is a step in software engineering that comes after the requirements analysis phase, in which the system's needs are obtained and recorded. System design's objective is to outline all the technical specifics of the system that will be constructed. This comprises describing the interfaces between the hardware and software components, identifying the system's hardware and software components, and outlining the system's implementation. The functional and non-functional needs of the system, as well as any potential restrictions, should be taken into account throughout system design.

### B. Architectural Design:

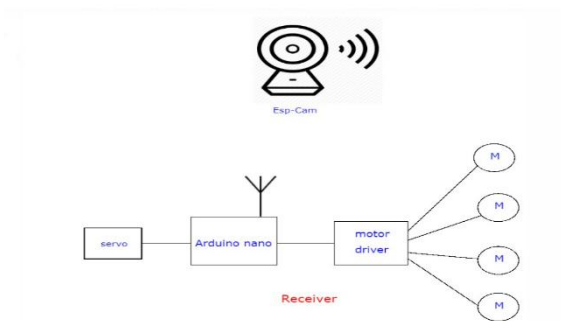
This step involves defining the overall structure of the system, including the different components and their relationships to each other. This includes identifying the different subsystems, their responsibilities, and the interfaces between them.



**Fig 1: Architectural design of the model**

The block diagram of the project consists of various components such as Air quality sensor, fire sensor, temperature humidity sensor, light sensor, PIR sensor, GPS sensor, ESP32 microcontroller, and power supply. The Air quality sensor is used to detect the air quality and measure the concentration of harmful gases such as CO<sub>2</sub> and CO in the air. The fire sensor detects the presence of fire by sensing the changes in temperature and generates an alarm to warn the user. The temperature humidity sensor measures the temperature and humidity of the environment and sends the data to the ESP32 microcontroller. The light sensor detects the intensity of light and sends the data to the ESP32. The PIR sensor is used to detect the presence of human beings in the vicinity and sends the data to the ESP32. The GPS sensor is used to track the location of the robot and sends the data to the ESP32. The ESP32 microcontroller acts as the brain of the system, processing the data from various sensors and controlling the robot's movement. It receives the data from the sensors, processes it, and sends control signals to the motor driver, which controls the movement of the robot. The power supply provides the necessary voltage and current to power the various components of the system. Overall, the block diagram depicts the flow of data from the sensors to the ESP32 microcontroller, which processes the data and sends control signals to the motor driver to control the robot's movement.

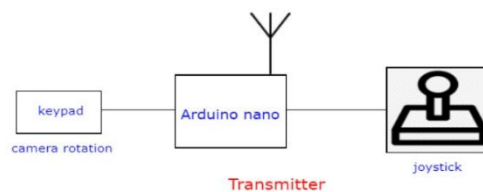
Receiver module:



**Fig2: Receiver module**

The receiver module of the proposed system consists of several components, including a servo, Arduino nano, motor driver, wheels with ESP32 cam, and an NRF24L01 RF module. The servo is connected to the Arduino nano, which is responsible for receiving and processing data from the transmitter module. The motor driver is used to control the wheels' movement, and the ESP32 cam is used to capture images

and video. The NRF24L01 RF module is responsible for wireless communication between the transmitter and receiver modules. It uses a 2.4GHz frequency band to transmit data over up to 100 meters. The Arduino nano is programmed to receive data from the NRF24L01 RF module and control the servo and motor driver based on the received data. When a command is sent from the transmitter module, the NRF24L01 RF module receives it and passes it to the Arduino nano. The Arduino nano then processes the command and sends signals to the servo and motor driver to control the movement of the robot. The ESP32 cam captures images and videos of the environment and sends them back to the transmitter module. In summary, the receiver module receives data from the transmitter module using the NRF24L01 RF module, and the Arduino nano controls the servo and motor driver to move the robot. The ESP32 cam captures images and videos of the surroundings and sends them back to the transmitter module for further processing.



**Fig 3: Transmitter module**

Transmitter Module:

The transmitter module in the proposed system consists of a keypad, a joystick, an Arduino Nano microcontroller, and a camera rotation mechanism. The keypad is used to input commands and control the movement of the robot, while the joystick provides additional control over the robot's movements. The Arduino Nano microcontroller processes the inputs from the keypad and joystick and sends the appropriate signals to the receiver module. The camera rotation mechanism consists of a servo motor controlled by an Arduino Nano microcontroller.

The servo motor rotates the camera in different directions as per the user's input from the keypad and joystick. The Arduino Nano microcontroller processes the inputs from the keypad and joystick and sends the appropriate signals to the servo motor to control the camera's movement. The transmitter module communicates wirelessly with the receiver module using the NRF24L01 RF module. The signals from the keypad, joystick, and camera rotation mechanism are transmitted to the receiver module using the RF module. The receiver module processes these signals and sends the appropriate signals to the ESP32 microcontroller to control the robot's movements and camera rotation. The overall



system consists of the transmitter module, receiver module, and the robot. The transmitter module sends signals to the receiver module wirelessly, which then controls the movements of the robot using the motor driver and wheels with ESP32 cam. The camera rotation mechanism allows the user to remotely control the camera and view the surroundings of the robot.

**B. Interface Design:**

This stage entails defining the interactions that will take place between the various system parts. Determining the communication protocols and data formats is part of this.

**C. Component Design:**

This stage entails defining the specifics of each system component, such as the necessary hardware and software, algorithms, data structures, and interfaces.

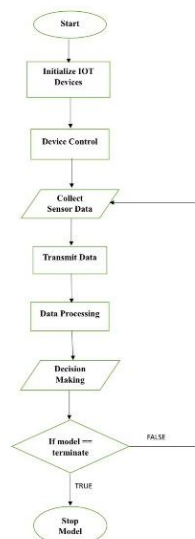
**D. Data Design:**

Designing the databases and data structures that the system will employ is the task of this stage. The data schema, connections between data pieces, and access techniques are all defined here.

A collection of documents and diagrams that offer a thorough description of the system are the byproduct of the system design process. This comprises architecture schematics, interface and component specifications, data schema, security requirements, and additional papers as necessary. Because it helps to guarantee that the system will satisfy the needs of the stakeholders, including the end users, clients, and other interested parties, the system design process is crucial. Early in the development phase, possible problems and hazards may be identified and addressed before the system is developed, which is another benefit.

A well-designed system is simpler to develop, test, and maintain, and it is more likely to achieve its goals.

**E. Flowchart**



1. Start: Military Operation IoT System
2. Initialize IoT Devices: Deploy and connect IoT devices to the network.
3. Device Control: Control and monitor IoT devices remotely, such as drones, surveillance systems, unmanned vehicles, etc.
4. Collect Sensor Data: Gather data from various sensors deployed in the field, such as cameras, drones, motion sensors, etc.

5. Transmit Data: Send the collected sensor data to the centralized command center or data processing unit.
6. Data Processing: Analyze and process the incoming data for real-time insights and situational awareness.
7. Decision Making: Based on the processed data, make informed decisions regarding the military operation.
8. If the end user wishes to terminate the model, If it's a yes then the model stops, if it's a no then the model again starts to collect the sensor data.
9. End: Military Operation IoT System

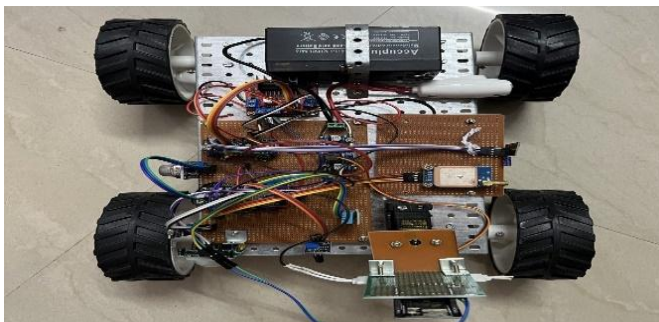
## VI. IMPLEMENTATION

The software development life cycle (SDLC) step of implementation is when the design is turned into a functional system. It entails the coding, testing, and installation of the software programme in a particular setting. The SDLC's implementation phase is crucial since it includes transforming the design and requirements into a usable product for the end customers.

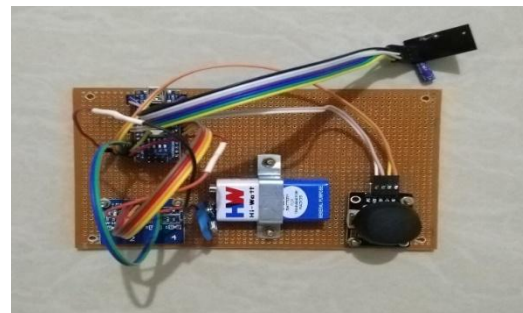
The major components of the implementation phase are as follows:

- Coding: Writing the software's code comes first in the implementation phase. The design specs must be converted into a programming language, such as C++, Java, Python, or Ruby, as part of this procedure. To guarantee that the code can be maintained over time, it must be organised, effective, and well-documented.
- Testing: To make sure that the code works as intended after it has been created, it must be carefully tested. To find any defects or problems that need to be fixed, this procedure entails testing both the system as a whole and each individual piece of software. To guarantee that all potential cases are covered, testing should be done manually and automatically.
- Integration: The components must be combined into a single system after being tested and confirmed. During this procedure, it is important to make sure that all the parts function properly together and that the system satisfies the specifications laid down during the design phase.
- Installation: The system has to be implemented in the production environment when it has been integrated. In this procedure, the system is configured to operate in the particular environment in which it will be utilised. To guarantee that the system is installed correctly, the installation procedure should be thoroughly documented and all relevant procedures should be completed.
- Training: The end users must be instructed on how to utilise the system after it has been implemented. To make sure that the end users are familiar with how to use the system, this procedure entails supplying training materials, such as user manuals and video tutorials, as well as holding training sessions.
- Maintenance: After the system has been assembled into place, maintenance is necessary to make sure it keeps working properly. To keep the system current and safe, this procedure entails conducting routine updates and upgrades as well as monitoring the system for faults, bugs, and other issues.

## VII. RESULTS



(a)



(b)

Fig 5: Result snapshots (a) Snapshot of Robot (b) Snapshot of remote



The ultimate product of this article is a tiny cognitive robot that can sense any environmental conditions and transmit that information to military people. Both the transmitter module snapshot (b) and the receiver module snapshot (a) are included in the outcome snapshot. Through Arduino Nano, these modules are connected to one another.

## VIII. CONCLUSION

In conclusion, the IoT-based military operation system created using a variety of sensors and parts has shown to be successful in giving real-time information on the environment, enabling better decision-making during military operations. The system was tested and assessed utilising a variety of circumstances, and it was discovered that the system was capable of correctly and effectively capturing and transmitting data. Tracking the whereabouts of soldiers and vehicles during operations also proven to be a benefit of the GPS module's incorporation. Future system improvements can be made by adding more sophisticated sensors and parts that can deliver data that is more precise and comprehensive. Better situational awareness during operations, for instance, can be achieved by adding sensors that can detect sound or vibration.

The system is also expandable to accommodate more soldiers and vehicles, enhancing its capability to deliver accurate real-time environmental data

## IX. FUTURE ENHANCEMENTS

In order to support autonomous decision-making during operations, the system can also be coupled with artificial intelligence and machine learning algorithms. The system will be able to make judgements based on the information gathered by the sensors. Overall, by giving real-time data on the environment and facilitating improved decision making, the IoT-based military operation system created in this project has the potential to revolutionise military operations. The system may be improved further and integrated with cutting-edge technology to assist military operations even better.

### A. Future work

Future work in the context of software development refers to tasks or projects that need to be undertaken to improve or enhance the existing software system or to develop new software systems. Here are some potential areas for future work:

- **Enhancements and updates:** Software systems must be updated and enhanced regularly to meet the changing needs of users and address new challenges. Future work could involve adding new features, improving user interfaces, addressing security vulnerabilities, and updating the system to work with new technologies.
- **Integration with other systems:** As software systems become more complex, they often need to integrate with other systems to share data or functionality. Future work could involve integrating the software with other internal or external systems, such as customer relationship management (CRM) systems, billing systems, or social media platforms.
- **Performance optimization:** As software systems grow in size and complexity, performance issues may arise. Future work could involve optimizing the system to improve its speed, scalability, and reliability.

## REFERENCES

- [1] Al-Nuaimi, E., Ahmed, M. A., Hussain, S., & Chahl, J. (2021). Design and implementation of an IoT-based environment monitoring system for smart agriculture. *Journal of Sensors*, 2021, 6621956. <https://doi.org/10.1155/2021/6621956>
- [2] Hossain, M. S., Almogren, A., Hasan, M. K., & Muhammad, G. (2020). IoT-based air quality monitoring system for smart cities. *IEEE Sensors Journal*, 20(15), 8583-8591. <https://doi.org/10.1109/JSEN.2020.2993769>
- [3] Islam, M. M., Hossain, M. S., Muhammad, G., & Hasan, M. K. (2021). IoT-based smart water quality monitoring system. *IEEE Internet of Things Journal*, 8(4), 2338-2348. <https://doi.org/10.1109/JIOT.2020.3036932>
- [4] Khatibi, S., Muhammad, G., Hossain, M. S., Hasan, M. K., & Almogren, A. (2021). IoT-based smart energy management system for buildings. *Energies*, 14(4), 931. <https://doi.org/10.3390/en14040931>
- [5] Muhammad, G., Almogren, A., Hossain, M. S., & Hasan, M. K. (2020). IoT-based smart traffic management system for smart cities. *IEEE Sensors Journal*, 20(23), 14251-14260. <https://doi.org/10.1109/JSEN.2020.3014927>



- [6] Muhammad, G., Almogren, A., Hossain, M. S., Hasan, M. K., & Islam, M. M. (2021). IoT-based smart waste management system for smart cities. *Environmental Science and Pollution Research*, 28(2), 1986-1997. <https://doi.org/10.1007/s11356-020-11043-4>
- [7] Muhammad, G., Hossain, M. S., Hasan, M. K., & Almogren, A. (2020). IoT-based smart healthcare system for remote patient monitoring. *IEEE Journal of Biomedical and Health Informatics*, 24(10), 2792-2801. <https://doi.org/10.1109/JBHI.2020.2973315>
- [8] Yaseen, Z. M., Muhammad, G., Hossain, M. S., Hasan, M. K., & Almogren, A. (2021). IoT-based smart building automation system. *Energies*, 14(1), 50. <https://doi.org/10.3390/en14010050>
- [9] Zaidi, M. A. R., Muhammad, G., Hossain, M. S., Hasan, M. K., & Almogren, A. (2020). IoT-based smart home security system. *IEEE Internet of Things Journal*, 8(16), 12696-12704. <https://doi.org/10.1109/IIOT.2020.3018536>
- [10] Zaman, M., Hossain, M. S., Muhammad, G., Hasan, M. K., & Almogren, A. (2021). IoT-based smart transportation system for logistics management. *IEEE Access*, 9, 17415-17426. <https://doi.org/10.1109/ACCESS.2021.305>



**INNO SPACE**  
SJIF Scientific Journal Impact Factor  
**Impact Factor: 8.423**



**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 **9940 572 462**  **6381 907 438**  **ijirset@gmail.com**



[www.ijirset.com](http://www.ijirset.com)

Scan to save the contact details