

Auto-Med Trolley: An IOT-Based Medical Delivery and Patient Monitoring Robot with Automated Sanitization Dock

Asmabi V.¹, Shamma Fathima², Nadhima P.³, Muhammed Ameen P. P.⁴, Muhammed Shifin⁵,
Yadhukrishnan H.⁶

¹Associate Professor, Dept. of Electronics and Communication Engineering, Al Ameen Engineering College, Palakkad, India

²³⁴⁵⁶Student, Dept. of Electronics and Communication Engineering, Al Ameen Engineering College, Palakkad, India

Abstract - This project presents the design and development of a smart Internet of Things based medical delivery and patient monitoring robot for healthcare environments. The system integrates embedded hardware, sensor-based monitoring, and web technologies to enable automated medicine delivery and real time patient health tracking. The robot is controlled through a web interface developed using a Flask based server and allows users to manually control robot movement or navigate to predefined hospital rooms. The system is built using ESP32 and ESP8266 microcontrollers for wireless communication and device control.

When the robot reaches a patient room, integrated biomedical sensors measure important health parameters such as heart rate, blood oxygen saturation, and body temperature. These values are transmitted through a Wi-Fi based HTTP communication system and displayed on a web dashboard for monitoring. A servo driven compartment automatically opens to deliver medicines to the patient.

To improve hospital hygiene and reduce contamination risks, the system also includes an automated sanitization dock. An infrared sensor detects the robot and activates a mist-based disinfecting mechanism along with UV light for disinfection of the robot surface. The proposed system demonstrates an effective integration of robotics, Internet of Things technology, and healthcare monitoring to create a smart hospital assistance platform that improves efficiency, hygiene, and patient care.

Key Words: IoT Healthcare, Medical Delivery Robot, Patient Monitoring, ESP32, Web Based Control, Sanitization System, Flask Server, Health Sensors.

1. INTRODUCTION

Healthcare environments require efficient systems for medicine delivery, patient monitoring, and infection control. With the rapid development of Internet of Things technologies and embedded systems, it is possible to design intelligent robotic platforms that assist medical staff while improving patient safety and hospital hygiene. Automated medical delivery systems can reduce human contact, minimize workload for healthcare workers, and enable faster service within hospital environments.

This project presents the development of a smart Internet of Things based medical delivery and patient monitoring robot with an automated sanitization dock. The system integrates robotics, web technologies, biomedical sensors, and wireless communication to create a multifunctional healthcare assistance platform. The robot is capable of delivering medicines to predefined hospital rooms while simultaneously monitoring patient vital parameters such as heart rate, blood oxygen saturation, and body temperature. The robot is controlled through a web-based interface developed using Flask and HTML, which allows users to manually control robot movement or send the robot to predefined destinations such as Room 1 or Room 2. Communication between the web interface and the robot is achieved using Wi-Fi based HTTP communication through ESP32 and ESP8266 microcontrollers. The robot follows a predefined time-based navigation algorithm to reach the selected location.

To improve hygiene and reduce contamination risk, the system also includes an automated sanitization dock. When the robot reaches the dock, an infrared sensor detects its presence and automatically activates a mist-based disinfecting mechanism along with an UV light. This ensures that the robot is sanitized before and after interacting within hospital environments.

1.1 Problem Statement

In many hospitals, medicine delivery and patient monitoring are still performed manually by healthcare workers. This process can be time consuming and increases the risk of contamination, especially in environments where hygiene and infection control are critical. Additionally, manual monitoring of patient vital parameters may delay real time reporting. Therefore, there is a need for an automated system that can deliver medicines, monitor patient health parameters, and ensure proper sanitization to maintain a hygienic healthcare environment.

1.2 Objectives of the Project

The main objectives of the proposed system are:

1. To design a web controlled medical delivery robot.
2. To monitor patient vital parameters such as heart rate, blood oxygen saturation, and body temperature.
3. To enable remote robot control through a web interface.
4. To implement predefined navigation for hospital rooms.
5. To develop an automated sanitization dock for disinfecting the robot.
6. To improve hospital hygiene and reduce manual workload.

2. LITERATURE REVIEW

Recent advancements in healthcare automation have led to the development of intelligent systems for patient monitoring and medical assistance. Various research works have focused on integrating Internet of Things (IoT) technology with robotic systems to improve efficiency and reduce human intervention in hospital environments.

These systems primarily focus on reducing the workload of healthcare staff and improving service speed. However, most existing systems lack integrated sanitization mechanisms, which are crucial in maintaining hygiene in sensitive environments such as hospitals. IoT-based healthcare monitoring systems have also been widely developed to track vital parameters such as heart rate, blood oxygen saturation (SpO₂), and body temperature in real time. These systems enable remote monitoring and timely medical intervention. However, many of these solutions operate independently and are not integrated with robotic delivery systems. In addition, mobile robots used in hospitals are generally controlled through predefined paths or manual control methods. While these robots improve operational efficiency, they often lack features such as real-time data transmission, automated decision-making, and user-friendly web-based interfaces. To address these limitations, the proposed system integrates medical delivery, patient monitoring, and automated sanitization into a single platform. The inclusion of a touchless IR-based hand sanitization mechanism and a sanitization dock with mist and UV disinfection provides an added advantage over existing systems by ensuring hygiene and safety. Thus, this project presents a more comprehensive and efficient solution by combining robotics, IoT communication, healthcare monitoring, and automated disinfection in a unified system.

3. SYSTEM ARCHITECTURE

The proposed system integrates a robotic delivery platform, biomedical monitoring sensors, a web-based

control interface, and an automated sanitization station. The system uses ESP32 and ESP8266 microcontrollers for robot control and wireless communication.

The system also includes an automated sanitization dock for robot disinfection. The web interface developed using Flask allows users to control the robot, monitor health parameters, and trigger robot movements. Commands are sent through WI-FI using HTTP communication. The ESP8266 receives these commands and forwards them to the ESP32 microcontroller which controls motors, sensors, and other actuators.

The system uses a local network-based communication setup where the web interface is accessed through a Telegram bot-generated IP address.

The robot is capable of navigating to predefined hospital rooms, delivering medicines using servo-controlled compartments, and collecting patient health data. After completing its tasks, the robot returns to a sanitization station where an automatic mist generator disinfects the robot before the next operation.

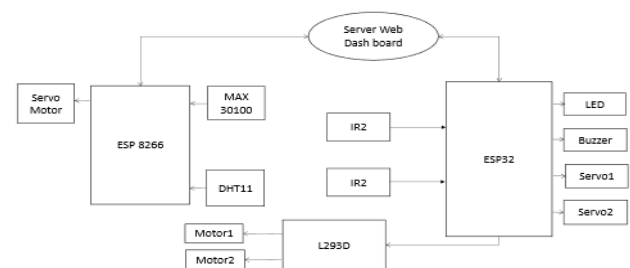


Fig -1: Block Diagram of Robot

4. HARDWARE DESIGN

The robot movement is controlled using the L293D motor driver module. The motor driver receives signals from the ESP32 and controls two DC motors which enable the robot to move forward, backward, left, and right. The system uses time-based movement to navigate the robot to predefined locations such as Room 1 and Room 2.

4.1 Microcontroller Unit

The system uses ESP32 microcontroller for control and communication. The ESP32 acts as the main processing unit that controls motors, servo mechanisms, sensors, and other hardware components. The ESP8266 is responsible for WIFI communication and connects the robot to the web server. Commands from the web interface are transmitted to the ESP32 through the ESP8266 module.

4.2 Motor Driver and Robot Movement

The robot movement is controlled using the L293D motor driver module. The motor driver receives

signals from the ESP32 and controls two DC motors which enable the robot to move forward, backward, left, and right. The system uses time-based movement to navigate the robot to predefined locations such as Room 1 and Room 2

4.3 Medicine Delivery Mechanism

Servo motors are used to control the opening and closing of medicine compartments. When the robot reaches the patient room, a buzzer indicates arrival and the servo motor opens the medicine box for the patient. After a fixed time interval, the compartment automatically closes.

4.4 Patient Health Monitoring Sensors

The system measures important patient health parameters using biomedical sensors. The MAX30100 sensor is used to measure heart rate and blood oxygen saturation (SpO₂). The DHT11 sensor is used to measure temperature. These sensors send data to the ESP8266 which then transmits the information to the web interface where the data is displayed in real time.

4.5 Obstacle Detection

Infrared sensors are used to detect obstacles during robot navigation. If an obstacle is detected, the system can stop or adjust movement to avoid collision.

5. HYGIENE AND SANITIZATION SYSTEM

The proposed system incorporates a dual-level sanitization mechanism to ensure hygiene and prevent contamination in healthcare environments. This includes a sanitization dock system and a touchless hand sanitization unit integrated within the robot.

5.1 Sanitization Dock Operation

The sanitization dock is designed as a dedicated station where the robot undergoes automatic disinfection before and after completing its tasks.

-An IR sensor detects the presence of the robot.

-The sensor signal is processed by an Arduino Nano controller.

A relay module is activated to control high-power devices.

-The ultrasonic mist generator produces disinfectant fog.

-UV LED is used for disinfection by reducing microbial contamination, thereby enhancing overall sterilization efficiency.

This ensures complete surface-level disinfection of the robot.

5.2 Arduino Nano Controller

The sanitization station uses an Arduino Nano microcontroller to control the mist generator and LED indicator. The Arduino receives signals from an infrared sensor which detects the presence of the robot inside the sanitization station.

5.3 IR Sensor Detection

The IR sensor detects when the robot enters the sanitization area. Once detection occurs, the sensor sends a signal to the Arduino Nano.

5.4 Relay Controlled Sanitization System

The Arduino activates relay modules which control high power devices such as the mist generator and UV Light LED. The relay acts as a switch that allows the low voltage microcontroller to control higher voltage components.

5.5 Ultrasonic Mist Generator

The ultrasonic mist generator produces disinfectant fog which covers the robot surface and eliminates microorganisms. This ensures proper sanitization before the robot continues operation.

5.6 UV Blue LED

A UV blue LED is used in the sanitization dock to enhance the disinfection process. It emits light in the ultraviolet/blue spectrum, which helps in reducing microbial contamination by damaging the DNA of microorganisms. When combined with the mist-based disinfectant.

5.7 Touchless Hand Sanitization System

To ensure user hygiene during medicine collection, a touchless sanitization mechanism is integrated into the robot.

-An IR sensor is placed near the sanitizer unit.

-When a hand is detected:

-Sanitizer is dispensed automatically.

-Signal is sent to ESP32.

The system allows medicine access only after sanitization. This creates a condition-based access control mechanism, improving infection prevention.

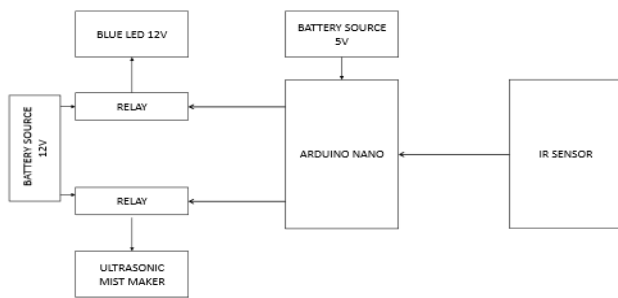


Fig -2: Block Diagram of the Sanitization Dock

6. SOFTWARE IMPLEMENTATION

The system software consists of embedded programming and a web-based control interface.

6.1 Arduino IDE

Arduino IDE is used to program the ESP32 and Arduino Nano microcontrollers. The firmware controls robot movement, sensor readings, servo motors, and communication with the web interface.

6.2 Web Interface using Flask

The web interface is developed using Python Flask framework.

The interface allows users to control the robot movement, start delivery sequences, and monitor patient health parameters. The dashboard displays real time values of heart rate, SpO₂, and temperature. It also provides manual navigation controls, room selection buttons, emergency commands, and UV control options. The web dashboard is accessed through a dynamically generated URL provided via a Telegram bot. The Flask-based web server operates on a local Wi-Fi network, and the dashboard is accessible only when the user device and the embedded system are connected to the same network. The Telegram bot generates the IP-based URL, which is then opened in a web browser to access the control interface. This ensures secure and controlled local communication between the user and the robot.

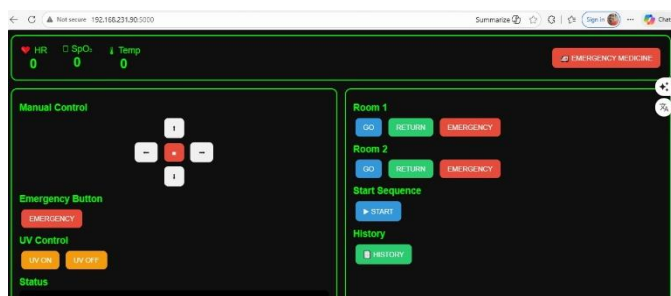


Fig -3: Web Dashboard

6.3 HTTP Communication

The system uses HTTP protocol for communication between the web interface and the robot. Commands from the dashboard are sent through Wi-Fi to the ESP8266 module which forwards them to the ESP32 for execution.

7. SENSOR SPECIFICATIONS

The performance of the proposed system depends on the accuracy, reliability, and response time of the sensors used for patient monitoring and obstacle detection.

The MAX30100 sensor is used for measuring heart rate and blood oxygen saturation (SpO₂). It operates using optical sensing techniques and provides stable and reliable readings with good accuracy. This makes it suitable for continuous and non-invasive health monitoring in basic healthcare applications.

The DHT11 sensor is used for temperature measurement. Although it offers moderate accuracy compared to advanced sensors, it is sufficient for general patient monitoring in non-critical environments.

Infrared (IR) sensors are used for both obstacle detection and touchless hand sanitization. These sensors work based on the reflection of infrared light and provide digital output when an object or hand is detected. Their fast response time ensures real-time operation.

Table -1: Sensor Specifications

Sensor Specifications				
Sensor Name	Parameter measured	Range	Accuracy	Resolution
MAX30100	Heart Rate	30-240 bpm	±2 bpm	1 bpm
MAX30100	SpO ₂	70-100%	±2 %	1%
DHT11	Temperature	0-50°C	±2°C	1°C
IR SENSOR	Obstacle/ Robot/ Hand	2-30 cm	High	Digital Output

8. SYSTEM ALGORITHM

8.1 Algorithm for System Initialization and Sanitization

When the system is powered ON, all microcontrollers including ESP32, ESP8266, and Arduino Nano are initialized. The system establishes a Wi-Fi connection and activates all sensors, actuators, and communication modules. The robot initially remains positioned at the sanitization dock. The sanitization dock ensures that the robot is disinfected before and after operation. When the robot enters or returns to the dock, an infrared (IR) sensor detects its presence. Upon detection, the Arduino Nano processes the signal and activates a relay module. The relay switches ON the ultrasonic mist generator and UV light. The mist generator produces a fine disinfectant fog that uniformly covers the robot surface, while the UV light provides additional microbial disinfection. After completion, the Arduino Nano deactivates the relay, turning OFF both the mist generator and UV light. The robot is now considered safe for operation and waits for user commands from the web interface.

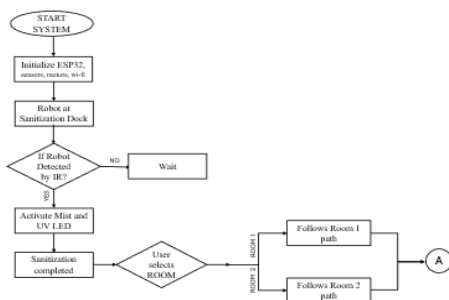


Fig -4: System Initialization Algorithm

8.2 Algorithm for Robot Navigation using Web Control

The user accesses the web dashboard and selects a command such as Room 1, Room 2, Manual Control, or Emergency. Based on the selected command, the web server sends an HTTP request through the ESP8266 module, which forwards the command to the ESP32 microcontroller. The ESP32 controls the motor driver and initiates robot movement toward the selected destination using time-based navigation. During movement, infrared sensors continuously monitor for obstacles to ensure safe and collision-free operation.

Upon reaching the selected room, the robot stops and a buzzer is activated to indicate arrival. Before allowing access to the medicine compartment, a hygiene verification step is enforced through a touchless hand sanitization system. An IR sensor placed near the sanitizer

detects the presence of the user's hand. When a hand is detected, the sanitizer mechanism is automatically activated, dispensing sanitizing liquid without physical contact. This ensures a hygienic interaction between the user and the robot.

Only after successful hand detection and sanitization, the system proceeds to the next step. The ESP32 then activates a servo motor to open the medicine compartment. The compartment remains open for a predefined duration, allowing the user to collect the medicine, and then automatically closes.

If an emergency command is triggered at any time, the robot immediately stops its current operation and returns to its initial position or sanitization dock.

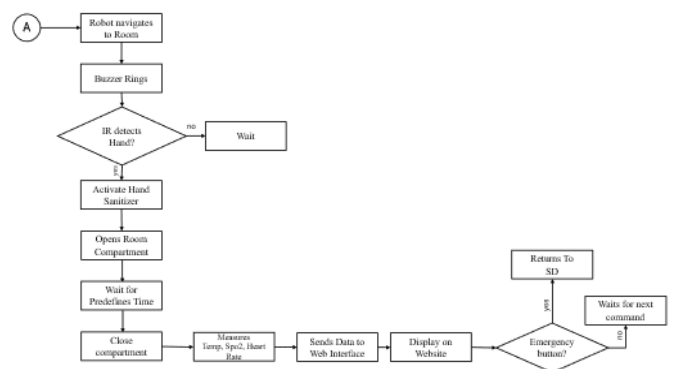


Fig -5: Robot Navigation Algorithm

8.3 Algorithm for Web Interface Operation

The system uses a Flask-based web interface to enable user interaction and control of the robot. The web dashboard is accessed through a dynamically generated URL provided by a Telegram bot.

The Flask server operates on a local Wi-Fi network, and the web interface can be accessed only when the user device and the robot system are connected to the same network. The Telegram bot generates an IP-based URL corresponding to the local server, which is then opened in a web browser to access the dashboard.

When the user selects a command on the web interface, an HTTP request is sent to the ESP8266 module. The ESP8266 acts as a communication bridge and forwards the request to the ESP32 microcontroller, which executes the corresponding operation such as robot movement, servo control, or sensor activation. Simultaneously, patient health data such as heart rate, SpO₂, and temperature are collected using biomedical sensors and transmitted to the web interface via Wi-Fi. These values are displayed in real time on the dashboard, enabling continuous monitoring.

The web interface also supports manual navigation, predefined room selection, and emergency control features, making the system flexible and user-friendly.

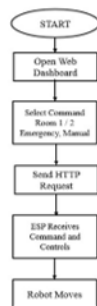


Fig -6: Web Interface Algorithm

9. SYSTEM WORKING

The overall working process of the system is as follows:

1. The user opens the web dashboard.
2. A command is selected such as Room 1, Room 2, or Manual Control.
3. The web server sends an HTTP request to the robot.
4. The ESP8266 receives the command and forwards it to the ESP32.
5. The ESP32 controls the motors and moves the robot to the selected location.
6. When the robot reaches the patient room, health sensors measure patient parameters.
7. Compartment opens only after successful hand sanitization detected by IR sensor
8. The sensor data is displayed on the web dashboard.
9. After completing delivery, the robot returns to the sanitization station.
10. The IR sensor detects the robot and activates the mist generator and UV Light to disinfect it.

This sequential operation ensures a fully automated, hygienic, and contactless medical delivery system with integrated safety validation.

10. RESULTS AND DISCUSSION

The proposed Auto-Med Trolley system was successfully developed and tested for medical delivery, patient monitoring, and automated sanitization in a controlled environment. The system demonstrated reliable performance in executing all major functions including robot navigation, sensor-based monitoring, and disinfection processes.

The robot was able to navigate to predefined locations such as Room 1 and Room 2 using time-based movement control. The integration of infrared sensors ensured effective obstacle detection, preventing collisions during movement. The addition of a touchless IR-based hand sanitization mechanism improved safety by ensuring that the medicine compartment opens only after proper hand sanitization.

The biomedical sensors used in the system, including MAX30100 and DHT11, provided stable and reasonably accurate readings for heart rate, SpO₂, and temperature. The data was successfully transmitted to the web interface using Wi-Fi-based HTTP communication and displayed in real time, enabling remote monitoring.

The sanitization dock performed effectively by automatically detecting the robot and activating the mist generator and UV disinfection system. This ensured proper sterilization before and after robot operation, significantly improving hygiene standards.

The web-based control system developed using Flask allowed users to easily control robot movement and monitor patient data. The interface was responsive and user-friendly, making the system suitable for practical healthcare applications.

Overall, the system achieved its objectives of reducing manual workload, improving hospital hygiene, and enabling real-time patient monitoring. The integration of multiple technologies into a single platform demonstrates the feasibility and effectiveness of IoT-based healthcare robotic systems.

11. ADVANTAGES

The proposed Auto-Med Trolley system offers several advantages in healthcare environments:

1. Reduces manual workload of healthcare staff by automating medicine delivery
2. Minimizes human contact, thereby reducing risk of infection spread
3. Ensures hygiene through dual sanitization: dock sanitization (mist + UV) and hand sanitization (touchless IR-based)

4. Provides real-time patient monitoring (heart rate, SpO₂, temperature)
5. Enables remote control through a web-based interface
6. Low-cost and easy to implement using readily available components
7. Improves operational efficiency and response time in hospitals
8. User-friendly interface with simple controls and real-time data display

2. Use of advanced sensors for more accurate patient monitoring
3. Cloud integration for remote monitoring beyond local Wi-Fi networks
4. Mobile application development for easier control instead of browser-based interface
5. AI-based health analysis and alert system for abnormal conditions
6. Camera integration for real-time video monitoring
7. Voice control or assistant-based interaction
8. Battery optimization and wireless charging for continuous operation
9. Expansion to multi-room or multi-floor hospital environments

12. PERFORMANCE ANALYSIS

The performance of the system was evaluated based on navigation accuracy, sensor response, communication reliability, and sanitization efficiency.

The robot demonstrated stable navigation using time-based control, successfully reaching predefined locations such as Room 1 and Room 2. Obstacle detection using IR sensors provided quick response, ensuring safe movement without collisions. The biomedical sensors (MAX30100 and DHT11) showed consistent performance with acceptable accuracy for non-critical monitoring applications. Real-time data transmission via Wi-Fi and HTTP protocol was reliable within the local network, with minimal latency.

The hand sanitization mechanism operated effectively, ensuring that the medicine compartment opened only after detecting proper user interaction. This adds an extra layer of hygiene validation.

The sanitization dock showed efficient performance by automatically activating mist and UV disinfection upon robot detection. The combined disinfection process improved overall cleanliness and reduced contamination risk. Overall, the system achieved reliable performance in automation, communication, and hygiene control. The integration of condition-based access control, real-time monitoring, and dual sanitization mechanisms demonstrates the system's effectiveness for practical healthcare deployment, especially in infection-sensitive environments.

13. FUTURE SCOPE

The proposed system can be further enhanced with the following improvements:

1. Integration of autonomous navigation using line-following or SLAM instead of time-based movement

14. CONCLUSIONS

The proposed system presents an effective IoT-based medical delivery robot integrated with real-time patient monitoring and advanced sanitization mechanisms. By incorporating both dock-based disinfection using mist and UV light, and touchless hand sanitization with condition-based access control, the system ensures high standards of hygiene and safety. The use of web-based control and local network communication enables efficient and reliable operation. Overall, the system reduces manual workload, minimizes human contact, and enhances operational efficiency in healthcare environments, making it a practical and scalable solution for modern hospitals.

REFERENCES

- [1] P. Joel Josephson, P. S. Kavhat, and H. U. Pawar, "Integrated Smart Trolley System for Patient Care and Isolation," IJARET, 2024.
- [2] M. Young, *The Technical Writer's Handbook*. Mill Valley, CA: University Science, 1989.
- [3] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for Smart Cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, 2014.
- [4] Espressif Systems, "ESP32 Technical Reference Manual," Espressif Systems, 2023.

BIOGRAPHIES

Asmabi V. - Associate Professor in the Department of Electronics and Communication Engineering at Al Ameen Engineering College, Palakkad. Her areas of interest include embedded systems, IoT, and communication technologies. She has guided several student projects in the field of automation and smart systems.



Shamma Fathima- Undergraduate Student in Electronics and Communication Engineering. Her areas of interest include IoT and embedded systems. She contributed to embedded programming, Python-based web development, microcontroller interfacing, and the implementation of the sanitization dock in this project.



Nadhima P. - an undergraduate student in Electronics and Communication Engineering with interests in sensor systems and embedded design. She contributed to sensor integration, circuit design, the mechanical body work of the robot and documentation work of the project.



Muhammed Ameen P. P. - is a student specializing in Electronics and Communication Engineering with a focus on robotics and IoT systems. He contributed to body integration and the integration of the health monitoring system.



Muhammed Shifin - an undergraduate student with an interest in embedded systems and automation. He contributed to body design, motor control, power supply connections, and the integration of the mist and UV sanitization system.



Yadhukrishnan H. - student in Electronics and Communication Engineering with an interest in communication systems and IoT. He contributed to body work and overall component integration, ensuring proper assembly and system functionality.