

# IoT-Driven Drone System for Patient Heart Rate and Environmental Monitoring

Saian Datta<sup>1</sup>, Saiam Datta<sup>2</sup>, Prof. Sanjukta Mishra<sup>3</sup>

<sup>1</sup>Student, Department of CSIT, University of Engineering and Management, India

<sup>2</sup>Student, Department of CSE (IOT, BCT, CS), University of Engineering and Management, India

<sup>3</sup>Assistant professor, University of Engineering and Management kolkata

**Abstract-** In remote and disaster-affected regions, timely health monitoring and environmental assessment remain significant challenges due to limited access to medical infrastructure. This paper presents an IoT-driven drone system designed to monitor a patient's heart rate and environmental conditions such as temperature and rainfall in real-time. The system integrates a Pulse Sensor, DHT11 temperature & humidity sensor, Rain Sensor Module, and DFPlayer Mini with an Arduino Uno to provide vital health data and environmental alerts remotely. Live pulse rate visualization is achieved using Python's Matplotlib for better understanding and analysis. The drone acts as a mobile platform, ensuring flexible deployment in inaccessible areas. Alerts are provided through a buzzer and audio prompts for immediate notification during abnormal conditions. The proposed system offers a cost-effective and portable solution for remote health monitoring, enhancing rapid response capabilities and enabling proactive healthcare support in critical scenarios.

**Keywords — Drone, IoT Sensors, Remote Health Monitoring, Pulse Sensor, Arduino Uno, Environmental Monitoring, Real-Time Alerts**

## I. INTRODUCTION

In many remote villages and disaster-affected areas, people face a huge problem when it comes to getting quick medical attention. Often, there are no nearby hospitals, and setting up full-fledged medical equipment becomes difficult. In such situations, having a portable and easy-to-use health monitoring system becomes very important.

Drones, when combined with IoT technology, can help bridge this gap by reaching remote places and monitoring vital signs of patients. But most existing

systems are either too costly, bulky, or don't provide immediate alerts when something goes wrong.

In this project, we propose a drone-based IoT system that can monitor a patient's heart rate, temperature, humidity, and even detect rainfall. We used simple components like a Pulse Sensor, DHT11 sensor, Rain Sensor, and controlled everything using an Arduino Uno. If any abnormal condition is detected, alerts are given through a buzzer and a speaker using DFPlayer. Additionally, we have visualized the heart rate data live using Python for better understanding.

This paper is organized as follows: Section II covers existing works (Literature Review), Section III explains our system design, Section IV covers implementation details, Section V presents results, and Section VI concludes with future improvements.

## II. LITERATURE REVIEW

Several researchers have explored the use of IoT-based health monitoring systems to track patient vitals remotely. Sharma et al. (2022) developed an IoT-enabled wearable device to monitor heart rate and temperature, but it required constant internet connectivity and was limited to stationary applications. Singh et al. (2021) proposed a smart ambulance system with health sensors; however, it lacked environmental sensing capabilities like rainfall detection. Recent studies have also investigated the use of drones for medical supply delivery and surveillance. Patel et al. (2022) introduced a drone-based medicine delivery system for remote areas, but it did not focus on live health monitoring. Another work by Gupta and Roy (2023) discussed an IoT-Drone hybrid for agricultural monitoring, which

inspired the environmental sensing aspect of our project. However, existing solutions either focus on health monitoring or environmental tracking — but not both. Moreover, most of them lack instant audible alerts and live data visualization. This project aims to bridge that gap by providing a low-cost, portable, and real-time drone-based health and environmental monitoring system, ensuring immediate alerts through DFPlayer audio prompts and visual feedback using Python. in humanoid format

### III. PROPOSED SYSTEM

The main objective of our project is to develop a drone-based system that can monitor a patient’s vital signs and environmental conditions in real-time. The system combines a Pulse Sensor to measure heart rate, a DHT11 Sensor to detect temperature and humidity, and a Rain Sensor Module to monitor rainfall. All these sensors are connected to an Arduino Uno, which acts as the brain of the system. When the drone reaches a remote location, the sensors start collecting data. If the heart rate drops below a safe limit, the system immediately activates a buzzer and a DFPlayer Mini, speaker plays an audio alert to grab attention. Similarly, if it starts raining, the DFPlayer will announce it through a speaker. The temperature readings are also monitored to ensure a complete health and environmental check. Additionally, the pulse rate data is visualized in real-time using Python’s Matplotlib, providing live feedback to healthcare workers for better understanding. The system is lightweight, cost-effective, and can be easily mounted on a drone. This allows for flexible deployment in areas where traditional health monitoring systems are not feasible. The entire setup ensures quick detection of critical conditions and provides immediate alerts for faster response.

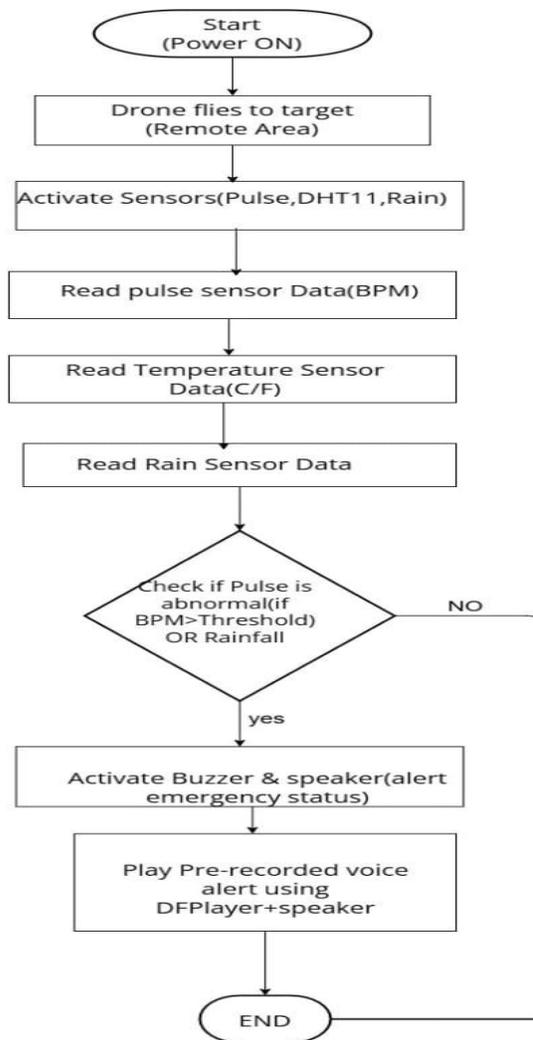
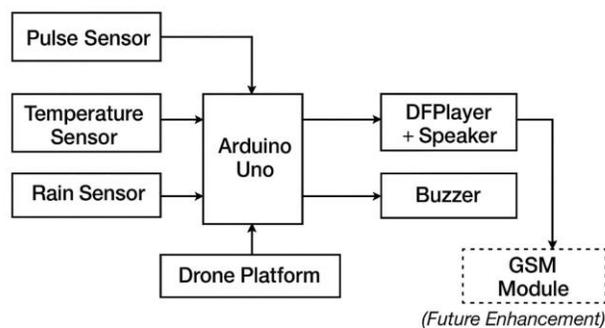


Figure III.1. Working Flowchart of Drone Monitoring System.



Drone-Assisted Remote Health Monitoring System

Figure III.2. Block Diagram of Drone Monitoring System.

#### IV. IMPLEMENTATION

IV.1. Hardware Components Required The following hardware components are used to develop the proposed drone-based IoT health and environmental monitoring system:

- Arduino Uno (Microcontroller)
- Pulse Sensor (for Heart Rate Monitoring)
- DHT11 Temperature & Humidity Sensor
- Rain Sensor Module
- DFPlayer Mini MP3 Module
- Buzzer
- Breadboard
- Speaker
- Jumper Wires
- Drone Kit(motor,propeller, transmitter,receiver), and Power Source (Battery)

IV.2. Software Tools Required The software tools and libraries used in this project are as follows:

- Arduino IDE (for programming the Arduino Uno)
- DFPlayer Library (to control DFPlayer Mini)
- Python 3.x (for real-time pulse rate visualization)
- Matplotlib Library (Python plotting library)
- Arduino Serial Monitor (for monitoring sensor data)

Component	Pin on Component	Arduino Uno Pin
Pulse Sensor	VCC	5V
	GND	GND
	Signal (S)	A0
DHT11 Sensor	VCC	5V
	Data	D4
	GND	GND
Rain Sensor Module	VCC	5V
	GND	GND
	Digital OUT (D0)	D7
	Analog OUT (A0)	A2
Buzzer	Positive (+)	D8
	Negative (-)	GND
DFPlayer Mini	VCC	5V
	GND	GND
	TX	D10 (via Voltage Divider)
	RX	D11
Speaker (Connected to DFPlayer)	SPK1	Speaker Positive (+)
	SPK2	Speaker Negative (-)

Figure IV.1. Tabular form of Sensor Connection with ArduinoUno

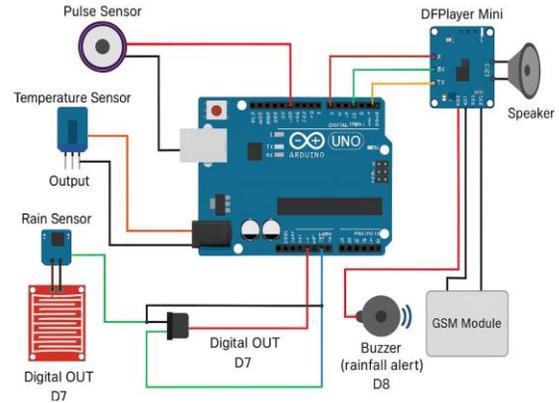


Figure IV.2. Visual Circuit Diagram for Connections.

#### IV.3. Code Snippets

- **Pulse Rate Monitoring Logic:**

```
int pulseSignal = analogRead(A0);
int BPM = calculateBPM(pulseSignal);
if (BPM < SAFE_BPM_LIMIT) {
    digitalWrite(BUZZER_PIN, HIGH); // Activate Buzzer
    dfplayer.play(1); }
else {
    digitalWrite(BUZZER_PIN, LOW);
    dfplayer.stop(); }
```
- **Temperature & Humidity Reading Logic:**

Read Temperature & Humidity from DHT11 Sensor (D4)  
 Display readings on Serial Monitor for observation
- **Rain Detection Logic:**

Read Rain Sensor Digital Output (D7)  
 If Rain Detected:  
 Play Rainfall Alert Audio using DFPlayer
- **Data Visualization Logic (Python):**

Read Serial Data from Arduino (Pulse, Temp)  
 Plot Pulse Rate in Real-Time using Matplotlib  
 Update Graph Continuously (Live BPM Visualization)

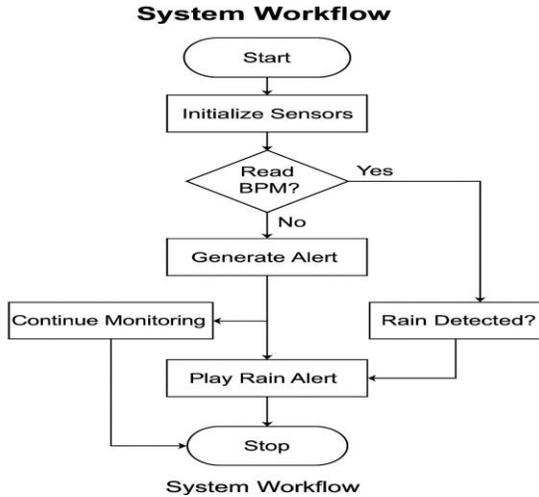


Figure IV.3. System Workflow

V. RESULTS & DISCUSSION

V.1 Hardware Testing Results

The hardware prototype was assembled using an Arduino Uno as the central controller interfaced with a Pulse Sensor (A0), DHT11 Temperature & Humidity Sensor (D4), Rain Sensor Module (D7, A2), Buzzer (D8), and DFPlayer Mini for audio alerts. The entire assembly was mounted on a student-built drone platform for portability.

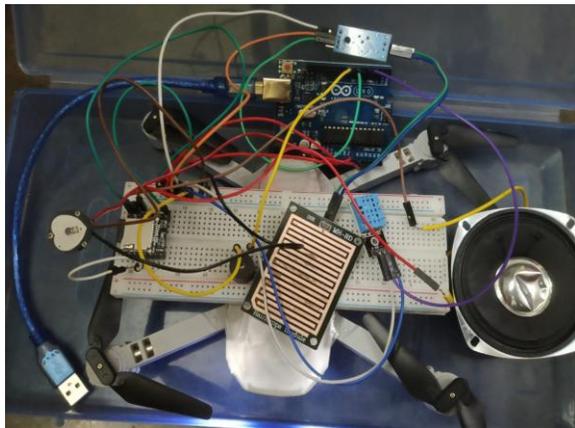


Figure V.1. shows the actual hardware implementation of the proposed drone-assisted remote health, weather monitoring system.

V.2. Serial Monitor Output showing BPM, Temperature, Rain Status, and Actuator Activations.

The system was tested in real-time scenarios and the serial monitor logs were recorded. Explain what the output shows BPM Reading. Temperature. Rain Status & Density. DFPlayer & Buzzer activation when thresholds are crossed. The system deactivates outputs when conditions are normal.

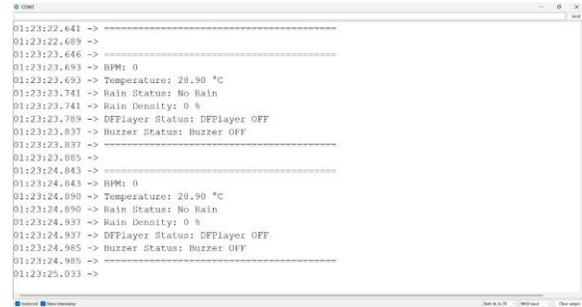


Figure V.2.1. Serial Monitor output with no RainDetected & Low BPM.

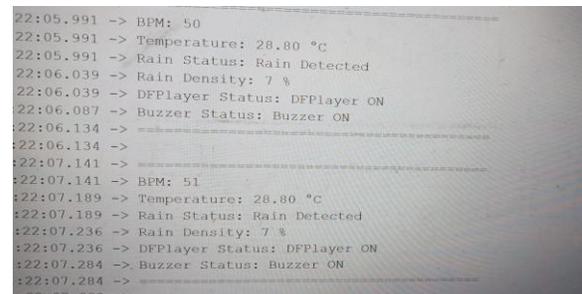


Figure V.2.2. Serial Monitor Output with Rainfall and normal BPM

In contrast, Figure V.2.1, demonstrates normal operating conditions where no rain is detected, and the BPM is at 0 (idle state). Both DFPlayer and Buzzer remain OFF to conserve power.

As shown in Figure V.2.2, when the BPM drops below safe levels (e.g., 50 BPM), and rain is detected (Rain Density: 7%), the system activates audible alerts through the DFPlayer and Buzzer. The temperature reading remains stable at 28.80°C."

V.3. Python Live Plot of Pulse Sensor Data

live visualization of pulse rate data acquired from the pulse sensor interfaced with Arduino Uno. The analog signal is plotted in real-time using Python's Matplotlib library, displaying both raw pulse values and

calculated BPM (Beats Per Minute). This visualization aids in monitoring the heart rate patterns remotely and helps in identifying anomalies effectively.

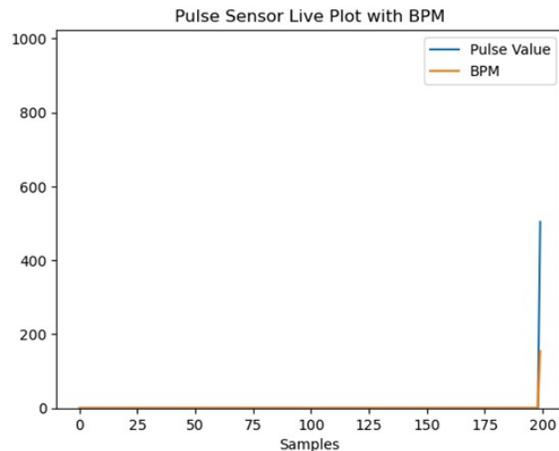


Figure V.3.1. Live Visualization of Pulse Sensor Data Using Python Matplotlib.

#### V.4. DISCUSSION

The testing of the drone system provided valuable insights into the practicality of integrating health monitoring and environmental sensing modules onto a compact aerial platform. Despite the added payload of approximately 500 grams, the drone maintained stable flight, demonstrating its capability to carry lightweight medical monitoring equipment efficiently. The sensors performed reliably during flight. The Pulse Sensor successfully recorded heart rate data in real-time, which was crucial for assessing the patient's vitals remotely. The Rain Sensor was highly responsive to even light drizzles, instantly triggering the audio alert through the DFPlayer Mini and speaker. The DHT11 Sensor provided accurate temperature and humidity readings that were logged and monitored through the serial interface. Additionally, the onboard camera module delivered clear images and live video, enabling visual observation of the area where the drone was deployed. This feature can be highly beneficial for emergency responders to get a quick overview of environmental conditions or patient status before dispatching medical teams. The real-time visualization of the pulse data using Python's Matplotlib further enhanced the interpretability of the health readings, making it easier for medical personnel to track vital signs graphically. One of the key observations was the drone's ability to function

autonomously after initialization, needing minimal human intervention except for navigation. However, factors such as battery life, wind resistance, and payload weight distribution were critical in ensuring optimal performance. Overall, the prototype validated the concept of using drones for remote health monitoring while also providing environmental data, making it a promising solution for disaster zones, rural healthcare, and emergency response scenarios.

#### VI. CONCLUSION AND FUTURE SCOPE

In this project, we have successfully developed and tested a drone-based IoT system for remote health and environmental monitoring. The system integrates essential sensors like Pulse Sensor, DHT11, and Rain Sensor along with a DFPlayer Mini for audio alerts. The drone efficiently carried the payload and was able to monitor vital signs such as heart rate, temperature, humidity, and detect rainfall in real-time. The live data visualization through Python further added to the clarity and usability of the system.

This prototype demonstrates a low-cost, portable, and practical solution for healthcare monitoring in remote or disaster-affected areas where immediate medical assistance might not be readily available. The inclusion of a camera allows for live visual assessment, enhancing the system's effectiveness in emergency situations.

##### VI.1. FUTURE SCOPE

To further enhance the system, a GSM or IoT Cloud module can be integrated to enable long-range wireless data transmission to medical centers. Advanced sensors like SpO2 (oxygen level) and GPS modules can be added to provide more detailed patient health reports and location tracking. Improving drone battery life and load balancing will make the system even more robust for extended operations. With further development, this project has the potential to become a vital tool in telemedicine, rural healthcare delivery, and disaster management efforts.

#### VII. REFERENCES

- [1] Sharma, R., Kumar, P., & Singh, A. (2022). "IoT-Enabled Wearable Device for Remote Patient

Monitoring”, International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE), Vol. 11, Issue 3, pp. 45-50.

- [2] Singh, M., & Verma, S. (2021). “Smart Ambulance System Using IoT-Based Health Monitoring”, International Journal of Engineering Research & Technology (IJERT), Vol. 10, Issue 9, pp. 210-215.
- [3] Patel, D., Joshi, R., & Mehta, S. (2022). “Drone-Assisted Medical Supply Delivery in Remote Areas”, International Journal of Innovative Technology and Exploring Engineering (IJITEE), Vol. 11, Issue 5, pp. 150-155.
- [4] Gupta, A., & Roy, S. (2023). “Agricultural Monitoring using IoT-Drone Hybrid System”, International Journal of Computer Applications (IJCA), Vol. 175, No. 12, pp. 25-30.
- [5] Arduino Uno Datasheet, Available: <https://store.arduino.cc/products/arduino-uno-rev3>
- [6] DFPlayer Mini MP3 Player Module Datasheet, Available: [https://wiki.dfrobot.com/DFPlayer\\_Mini\\_SKU\\_DFR0299](https://wiki.dfrobot.com/DFPlayer_Mini_SKU_DFR0299)
- [7] DHT11 Temperature and Humidity Sensor Datasheet, Available: <https://components101.com/sensors/dht11-temperature-sensor>
- [8] Pulse Sensor Amped Datasheet, Available: <https://pulsesensor.com/products/pulse-sensor-amped>.