A

MINI PROJECT SYNOPSIS

ON

IOT based Patient health monitoring system

Submitted by

1) Suraj Kundlik Nikam (22410085)
2) Advita Ratnakar Vankore (22410087)
3) Anushka Moreshwar Ingale (22410090)

For

T.Y. B.Tech.(Electronics Engineering)

Under the guidance of

Dr.B.G.Patil

Department of Electronics Engineering,

Walchand College of Engineering, Sangli

DEPARTMENT OF ELECTRONICS ENGINEERING

WALCHAND COLLEGE OF ENGINNERING ,SANGLI

2024-25

Outline

- Introduction
- Literature review
- Problem statement
- Block Diagram
- Working
- Components required
- References

1. BRIEF INTRODUCTION :

Nowadays Health-care Environment has developed based on Wireless-Sensing node Technolgy. Patients are facing a problematic situation of unforeseen demise due to the specific reason of heart problems and attack which is because of nonexistence of good medical maintenance to patients at the needed time. This is for specially monitoring the old age patients and informing doctors and loved ones. So we are proposing a innovative project to dodge such sudden death rates by using Patient Health Monitoring that uses sensor technology and uses internet to communicate to the loved ones in case of problems.

2. LITERATURE REVIEW :

In view of current research area various journals are referred for up-to-date technical content some of them are discussed in brief as follows:

1. Prajoona Valsalan from Dhofar University (2020) detailed in the paper "IoT Based Health Monitoring System" (Journal of Critical Reviews, April 2020), has demonstrated that:
According to recent advancements in IoT-based health monitoring systems, significant impacts have been made in personal health management through the integration of various sensors and wireless communication technologies. These systems enable real-time monitoring of vital health parameters, such as heart rate and body temperature, and facilitate remote data analysis. IoT systems enhance continuous health tracking and early detection of health anomalies. This research underscores the potential for reducing the burden on healthcare professionals and facilities by providing a more personalized and efficient remote health monitoring solution.

2. Jayakumar S. et al. (2021) In their research titled "IoT Based Health Monitoring System" (2021), IoT technology plays a critical role in improving patient health monitoring by enabling real-time data collection and remote access to vital signs such as heart rate, body temperature, and ECG readings. The system utilizes sensors connected to an Arduino microcontroller, which transmits the data to a cloud platform like ThingSpeak for further processing and storage.The study emphasizes that this approach reduces the burden on healthcare professionals by providing continuous monitoring and alerting caregivers in case of emergencies. It also states that such systems are particularly useful in rural areas, where access to healthcare services is limited, helping patients receive timely medical attention. The research underscores the potential for IoT to revolutionize healthcare by offering a low-cost, scalable, and efficient solution for remote health monitoring.

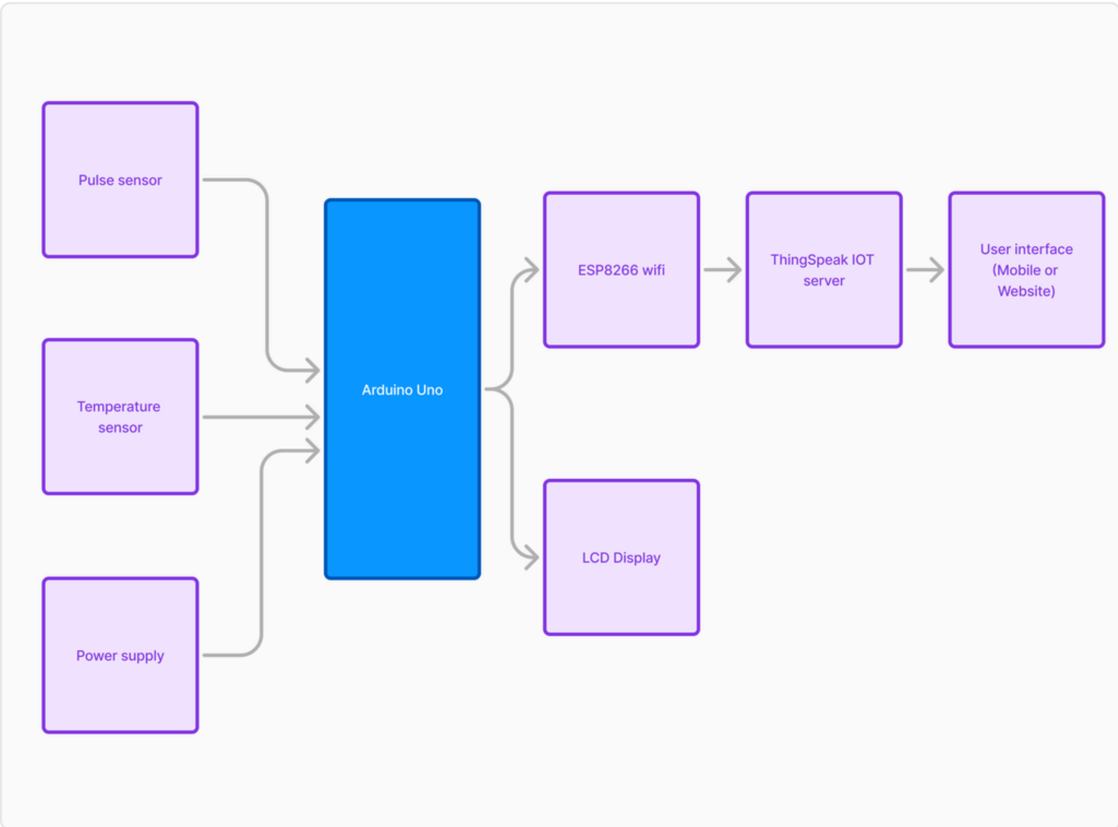
3. Studies by Islam et al. (2015) and Ullah et al. (2019) support the abstract's claims about IoT's ability to improve healthcare efficiency and reduce costs. The emphasis on 24/7 monitoring echoes work by Majumder et al. (2017), while the benefits of remote monitoring and emergency response are explored by Hassanalieragh et al. (2015). The concept of family/guardian monitoring mentioned in the abstract is reflected in research by Almotiri et al. (2016). While the abstract doesn't provide specifics on device comparisons, such analyses are common in the field, as demonstrated by Thota et al. (2018). Overall, the abstract effectively summarizes key themes in IoT healthcare research, emphasizing its potential to enhance patient care and reduce healthcare costs.

4.Abinaya Inbamani(2022) and Saranya.L.'s smart healthcare paradigm leverages sensors, processors, and communication technologies to enhance service quality and reliability. IoT enables real-time data extraction and remote advisory support, with collected information stored for further analysis. Meanwhile, ML algorithms improve predictability and enable earlier diagnosis, with feedback modules enhancing future predictions. However, the implementation of these technologies faces significant challenges, particularly in protocol standardization and data security. The merger of IoT and ML in healthcare focuses on specific applications such as diabetes management, blood pressure monitoring, and heart rate tracking. To address security concerns arising from this integration, measures such as authentication, authorization, user control, and data provenance are proposed. The development of intelligent, standardized protocols is crucial for ensuring accessibility and feasibility for all users. As research in this field progresses, it is essential to explore the current state of IoT protocols in healthcare, applications of ML in medical diagnosis, successful case studies of IoT and ML integration, standardization efforts, security measures for patient data protection, comparative analyses of ML techniques for health monitoring, and the ethical considerations surrounding AI-driven healthcare solutions.

3. PROBLEM STATEMENT :

To design an effective solution for monitoring and remote analysis of patient health parameters using IOT based health monitoring system, including an user interface ('Thing Speak') for real time monitoring of health parameters.

4. BLOCK DIAGRAM :



This is a simple block diagram that explains the IoT Based Patient Health Monitoring System using ESP8266 & Arduino. Pulse Sensor and LM35 Temperature Sensors measure BPM & Environmental Temperature respectively. The Arduino processes the code and displays it to 16*2 LCD Display. ESP8266 Wi-Fi module connects to Wi-Fi and sends the data to IoT device server. The IoT server used here is Thingspeak. Finally, the data can be monitored from any part of the world by logging into the Thingspeak channel.

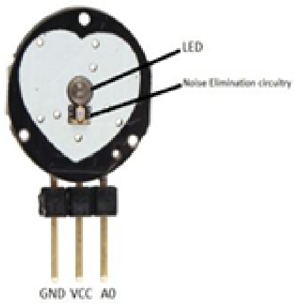
5. WORKING :

The IoT-based Patient Health Monitoring System utilizes an ESP8266 and Arduino to track vital signs like BPM (heart rate) and body temperature using a pulse sensor and an LM35 temperature sensor, respectively. The data is processed by the Arduino and displayed on a 16x2 LCD screen for local monitoring. Additionally, the ESP8266 Wi-Fi module connects to the internet and sends the data to an IoT server using the ThingSpeak platform. This setup allows the health data to be accessed remotely from anywhere in the world via the ThingSpeak channel, ensuring real-time monitoring and analysis.

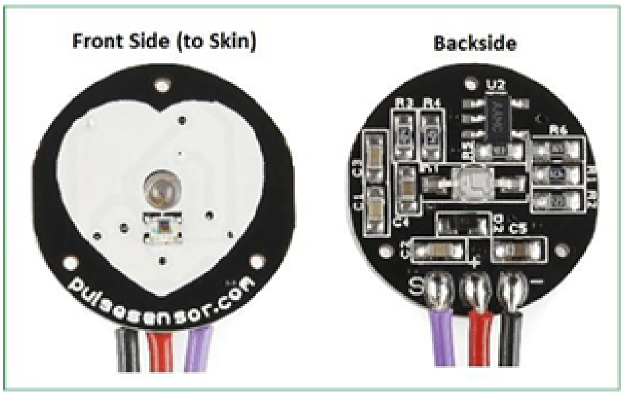
6. COMPONENTS REQUIRED :

Sr.No.	Component Name	Quantity
1	Arduino Uno Board	1
2	ESP8266-01 WiFi Module	1
3	16x2 LCD Display	1
4	Pulse Sensor	1
5	LM35 Temperature Sensor	1
6	Resistors	2
7	Wires, Breadboard	-

Pulse Sensor:

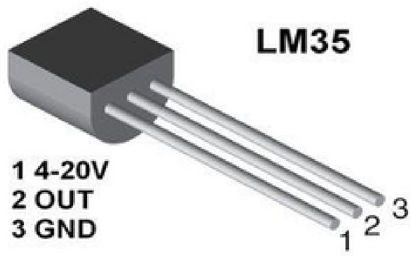


The pulse sensor has three pins: VCC, GND & Analog Pin. The Pulse Sensor is a plug-and play heart-rate sensor for Arduino. Clip the Pulse Sensor to our earlobe or fingertip and plug it into our Arduino, we can ready to read heart rate. Also, it has an Arduino demo code that makes it easy to use.

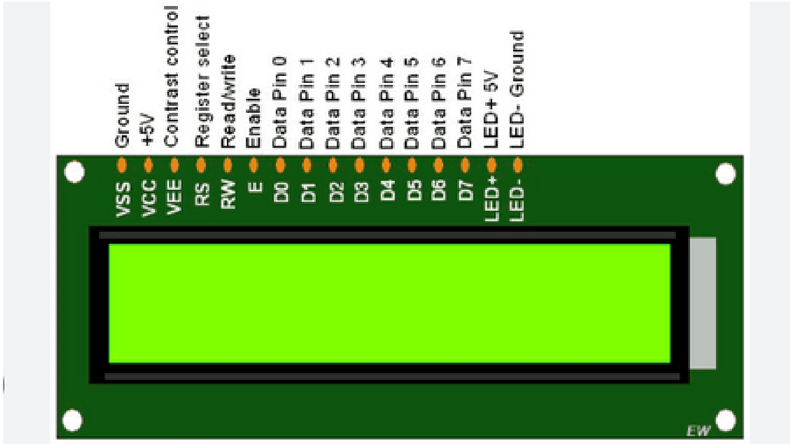


LM35 Temperature Sensor:

The LM35 series are precision integrated-circuit temperature devices with an output voltage linearly proportional to the Centigrade temperature. The LM35 device has an advantage over linear temperature sensors calibrated in Kelvin, as the user is not required to subtract a large constant voltage from the output to obtain convenient Centigrade scaling.

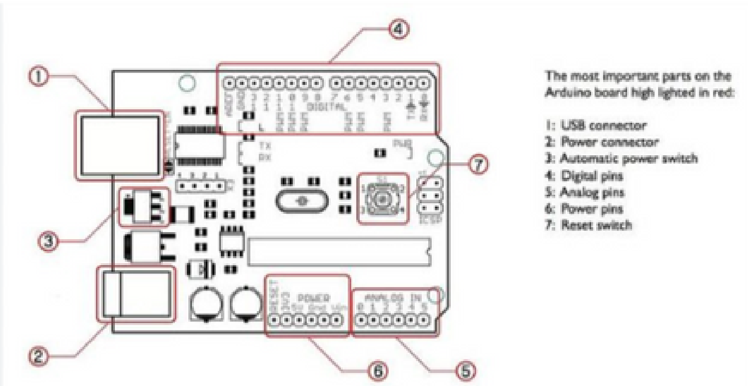


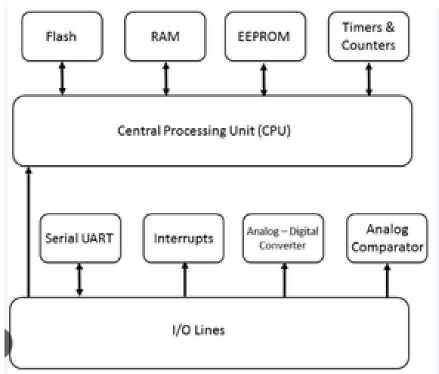
LCD:



A 16x2 LCD (Liquid Crystal Display) is a basic electronic display module capable of displaying 16 characters per line on 2 lines, totaling 32 characters at once. Each character is displayed in a 5x7 pixel matrix format. This type of display operates on 5V power supply and can display alphanumeric characters, symbols, and custom characters. It is widely used in embedded systems, electronics projects, and various appliances due to its low cost, ease of programming, and readability.

Arduino:

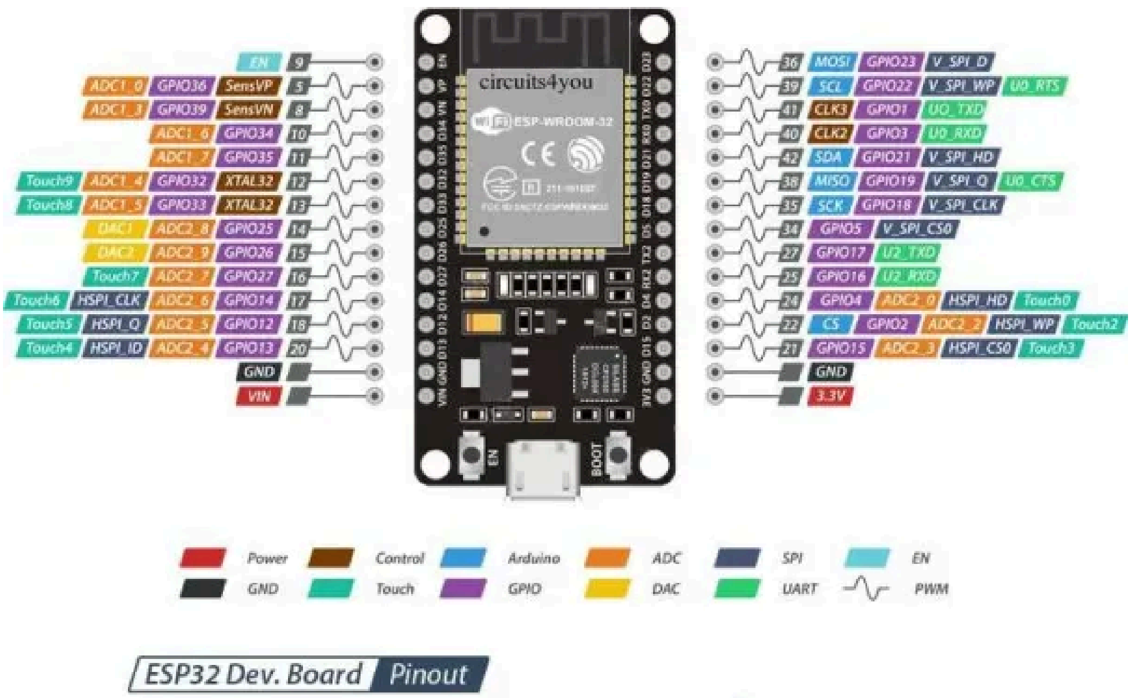




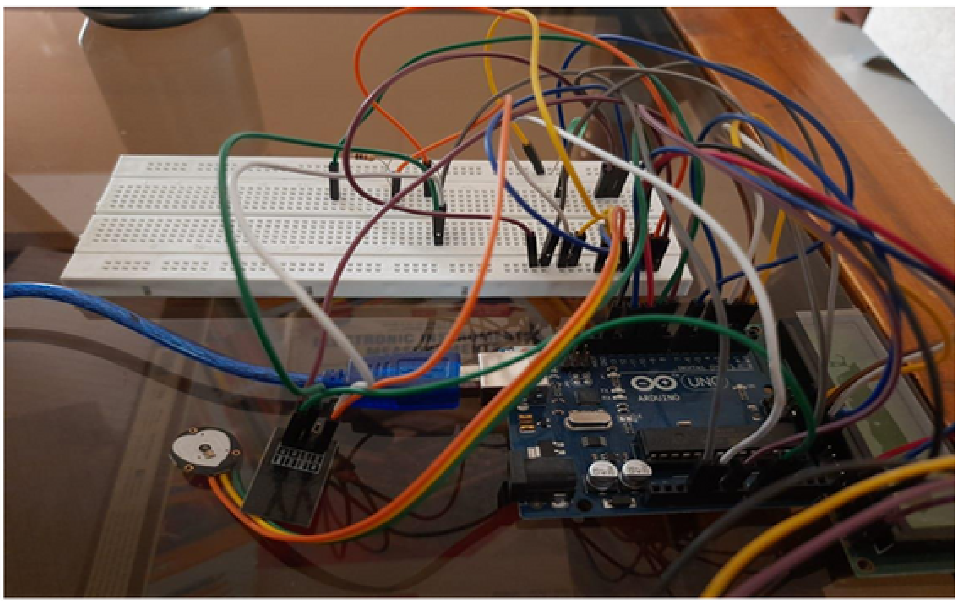
Arduino is an open-source electronics platform that combines user-friendly hardware and software, making it ideal for creating interactive projects and prototypes. The platform is built around microcontroller boards that can read inputs (like sensors, buttons, or digital signals) and convert them into outputs (such as activating motors, turning on LEDs, or displaying information). Its simplified C++ programming environment, extensive library support, and active community have made Arduino a cornerstone of DIY electronics, STEM education, and rapid prototyping in fields ranging from hobbyist projects to industrial applications.

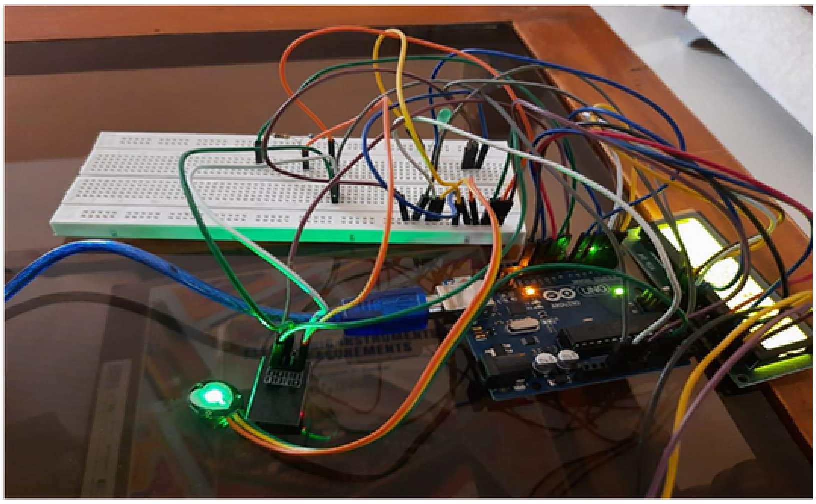
ESP32(Wifi Module:

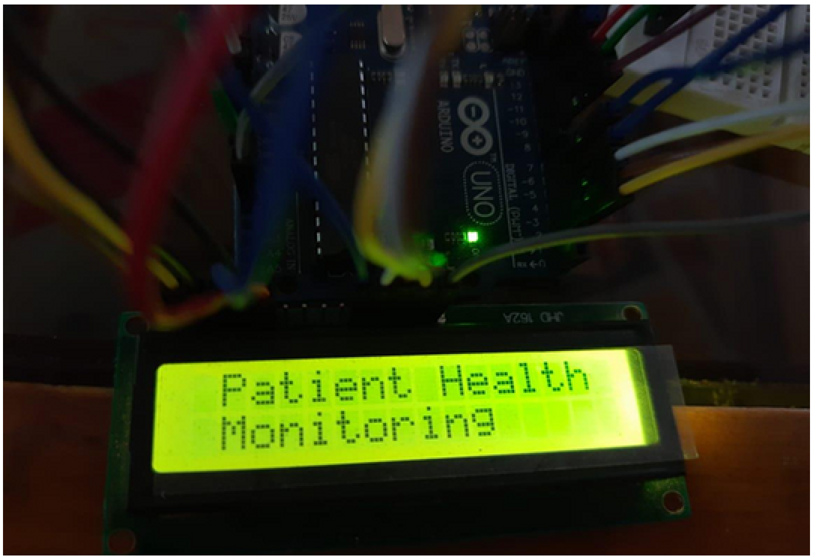
The ESP32 is a very user-friendly and low-cost device to provide internet connectivity to our projects. The module can work both as an Access point (can create hotspot) and as a station (can connect to Wi-Fi), hence it can easily fetch data and upload it to the internet making the Internet of Things as easy as possible.



<u>Experimental Setup :</u>







Source Code :

```
#include <LiquidCrystal.h>
#include <SoftwareSerial.h>
#include <PulseSensorPlayground.h> // Include PulseSensor library
// LCD and SoftwareSerial setup
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
SoftwareSerial mySerial(9, 10); // RX, TX (Arduino communicates with ESP32)
// Pin and variable declarations
int tempPin = A1;
float temp = 0;
int Threshold = 550; // Threshold for pulse detection
const int PulseWire = A0; // PulseSensor connected to analog pin A0
const int LED = LED_BUILTIN;
// Create PulseSensor object
PulseSensorPlayground pulseSensor;
```

```
void setup() {
 lcd.begin(16, 2); // Initialize LCD
 pinMode(tempPin, INPUT); // Set temperature sensor pin as input
 Serial.begin(9600); // For debugging
 mySerial.begin(9600); // Communication with ESP32
 lcd.clear();
 lcd.setCursor(0, 0);
 lcd.print("IOT based Health Monitoring System");
 delay(2000);
 lcd.clear();
 // Configure PulseSensor
 pulseSensor.analogInput(PulseWire);
 pulseSensor.blinkOnPulse(LED); // Blink built-in LED with pulse
 pulseSensor.setThreshold(Threshold);
 // Check if PulseSensor object starts correctly
 if (pulseSensor.begin()) {
 Serial.println("PulseSensor initialized successfully!");
 }
}
void loop() {
 // Clear the display once at the start of each loop
 lcd.clear();
 // Read and display pulse
 int myBPM = calculateBPM();
 // Read and display temperature
 readAndDisplayTemperature();
 // Send structured data to ESP32 in JSON-like format
 sendDataToESP(myBPM, temp);
 delay(15000); // Delay for 15 seconds
}
// Function to read and calculate BPM
int calculateBPM() {
 int myBPM = 0; // Initialize BPM variable
 if (pulseSensor.sawStartOfBeat()) { // Check if a beat is detected
 myBPM = pulseSensor.getBeatsPerMinute(); // Get BPM value
 Serial.println("♥ A HeartBeat Happened! ");
 Serial.print("BPM: ");
 Serial.println(myBPM);
 }
 // Display BPM on LCD (row 0)
 lcd.setCursor(0, 0);
 lcd.print("BPM: ");
 lcd.print(myBPM);
 return myBPM;
}
```

```
// Function to read and display temperature
void readAndDisplayTemperature() {
 int tempValue = analogRead(tempPin);
 float temperatureC = (tempValue * 5.0 / 1024.0) * 100; // Convert to Celsius
 float temperatureF = temperatureC * 1.8 + 32; // Convert to Fahrenheit
 temp = temperatureF;
 // Print temperature on LCD (row 1)
 lcd.setCursor(0, 1);
 lcd.print("Temp: ");
 lcd.print(temperatureF);
 lcd.print(" F");
 // Print to Serial for debugging
 Serial.print("Temp: ");
 Serial.print(temperatureF);
 Serial.println(" F");
}
// Function to send data to ESP32
void sendDataToESP(int bpm, float temp) {
 // Send the data in a structured format (JSON-like)
 String dataToSend = "{\"BPM\": " + String(bpm) + ", \"TEMP\": " + String(temp) + "}";

 // Print to Serial Monitor (for debugging)
 Serial.println("Sending data to ESP32: " + dataToSend);

 // Send the data to ESP32 via SoftwareSerial
 mySerial.println(dataToSend);
}
```

12. REFERENCES:

1. Abinaya Inbamani, Saranya.L., Birundha.S., Janani.V., A.Siva Sakthi, Vidhya.H., and Viswak Sena.T., "Internet of Things Embedded in Healthcare Applications," 2022 6th International Conference on Computing Methodologies and Communication (ICCMC), pp. 382-489, 2022.
2. Abdulmalek S, Nasir A, Jabbar WA, Almuhaya MAM, Bairagi AK, Khan MA, Kee SH. IoT-Based Healthcare-Monitoring System towards Improving Quality of Life: A Review. Healthcare (Basel). 2022 Oct 11;10(10):1993. doi: 10.3390/healthcare10101993. PMID: 36292441; PMCID: PMC9601552.
3. Valsalan, Prajoona & Tariq, Ahmed & Hussain, Ali. (2020). IOT BASED HEALTH MONITORING SYSTEM. 2020. 10.31838/jcr.07.04.137.
4. Jayakumar S. et al. (2021),Department of EIE, KCG college of technology, Chennai, India. Series-Advances in Parallel Computing, Ebook-Volume 40: Advances in Parallel Computing Technologies and Applications