iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** V **Month of publication:** May 2025

DOI: https://doi.org/10.22214/ijraset.2025.71112

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

IoT-Based Worker Safety System with Real-Time Health Monitoring and Emergency Rescue using RSSI

Dr. M. Newlin Rajkumar[1], Anush Kumar S[2], Subiya V M[3]
[1]Assistant Professor, [2, 3]Student, Department of EEE, Anna University Regional Campus, Coimbatore

Abstract: In the construction industry, worker safety is a critical concern, particularly during emergencies such as building collapses, where delayed response and inadequate health monitoring can result in severe injuries or fatalities. Traditional systems often lack real-time tracking and health monitoring capabilities, making it difficult to locate and assist affected workers promptly. To address these challenges, we propose a Smart Employee Health Monitoring and Emergency Rescue System utilizing Received Signal Strength Indicator (RSSI) technology. The system integrates ESP32 and Zigbee-based Wireless Sensor Network (WSN) modules to enable continuous health monitoring and precise indoor positioning of workers on construction sites. By leveraging RSSI-based location tracking and real-time communication with a central control unit, the system ensures low-power, reliable data transmission for efficient emergency alerts and worker status updates. This integration significantly enhances the speed and accuracy of rescue operations, while also enabling proactive health management, ultimately improving occupational safety and emergency response in high-risk environments.
Keywords: Control Centre Module, ESP32, Received Signal Strength Indicator, Worker Module, Zigbee

I. INTRODUCTION

In high-risk workplaces, such as construction sites and industrial facilities, employee safety is constantly threatened by unforeseen accidents, especially structural failures like building collapses. During such emergencies, time is a critical factor that can determine the survival of trapped workers. However, traditional emergency response methods often fall short in terms of speed, precision, and real-time monitoring, resulting in delayed rescues and increased casualties. Manual search techniques, visual inspections, or basic location estimation tools lack the ability to accurately detect and locate injured individuals, especially under heavy debris. To address these pressing concerns, this project introduces a Smart Employee Health Monitoring and Emergency Rescue System (SH-ERS), specifically designed to enhance worker safety and improve emergency response in the event of structural collapse. The proposed system leverages Received Signal Strength Indicator (RSSI) technology for accurate indoor positioning, while integrating wearable health sensors connected through a wireless sensor network (WSN) based on ESP32 and Zigbee modules. These devices constantly monitor the physiological parameters of employees and transmit real-time data such as body temperature, heart rate, and movement to a central control unit. In case of a collapse, the system identifies trapped individuals' exact locations using RSSI-based triangulation and sends immediate alerts to rescue teams. This not only speeds up rescue operations but also ensures that injured workers receive timely assistance based on their health status.

II. SYSTEM DESIGN

A. *Existing System*

Workers wearing GPS-enabled devices can be accurately tracked in real time, enabling improved monitoring and management of personnel on large and complex construction sites. These devices provide precise geographic coordinates, which are especially useful in open outdoor environments where satellite signals are strong and unobstructed. supervisors can use the real-time data to ensure that workers remain within designated zones, respond quickly to emergencies, and analyze movement patterns for efficiency and safety improvements. However, GPS-based tracking systems face significant limitations in indoor, underground, or densely built-up areas where signals may be weak, blocked, or reflected, leading to inaccurate positioning or signal loss. This limitation makes GPS unsuitable for certain construction scenarios, such as tunnel work or multi-story building interiors.

The alternative positioning technologies such as Time-of-Flight (ToF) systems and Bluetooth low energy (BLE) beacons are increasingly adopted. The Time of Flight system calculates a worker's location with respect to the worker's location using propogation delay based range methods.

While this method is generally useful for indoor positioning, it can suffer from low accuracy due to signal interference or reflection from metal surfaces common on construction sites. On the other hand, BLE beacon-based systems offer a more practical solution for indoor environments. Beacons placed throughout the site emit signals that are received by wearable devices, and the worker's location is determined by triangulating signal strength from multiple beacons. although the accuracy is limited to within a few meters, this method is effective for general location awareness and zone-based safety monitoring. These indoor tracking methods provide a viable supplement to GPS, creating a hybrid positioning system that ensures continuous worker tracking across various site conditions, thereby enhancing both operational oversight and worker safety.

B. *Proposed System*

The proposed Smart Employee Health Monitoring and Emergency Rescue System aims to address critical worker safety concerns on construction sites, particularly in emergency situations like building collapses. Traditional systems often fall short in providing real-time tracking and health monitoring, which can delay response times and lead to severe consequences for workers. Our system leverages Received Signal Strength Indicator to offer precise indoor positioning, even in complex construction environments, alongside continuous health monitoring through integrated sensors.

The system utilizes ESP32 microcontrollers and Zigbee-based Wireless Sensor Network modules to facilitate communication and data transfer. These modules enable the real-time collection of health metrics, such as heart rate, body temperature, and oxygen levels, while also tracking the worker’s precise location via RSSI-based location tracking. The low-power design ensures that the system can operate continuously without excessive battery drain, even in demanding construction settings.

The integration of RSSI technology ensures that workers’ locations are accurately tracked, even in multi-floor or obstructed areas where traditional GPS systems fail. If an emergency occurs, the system promptly communicates with a central control unit, which alerts emergency responders with precise information about the worker's condition and location. This functionality accelerates rescue operations, ensuring that help is dispatched immediately and efficiently. Additionally, the proactive health monitoring system continuously evaluates workers' well-being, offering early detection of potential health issues and preventing medical emergencies from escalating.

Incorporating both location tracking and health monitoring into a single system, this approach enhances overall occupational safety, improves emergency response times, and enables more effective worker management. The end result is a comprehensive, intelligent solution that fosters a safer, more responsive construction site environment.

III. HARDWARE AND SOFTWARE REQUIREMENTS

A. *Hardware Requirements*

- ESP32
- Body Temperature sensor – DS18B20
- Heart beat sensor
- Environment Temperature sensor – DHT11
- Humidity sensor – DHT11
- Air Quality Sensor – MQ135
- Transceiver
- LCD
- Buzzer
- Power Supply

B. *Software Requirements*

- Arduino IDE 1.8
- ➢ Embedded Programming
- ThinkSpeak Website

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

- Proteus

IV. BLOCK DIAGRAM

The proposed block diagram is structured into two key modules.

1) Worker Module
2) Control Centre Module

A. Block Diagram Of Worker Module

The Block diagram of Worker module is shown in Figure 1.1.

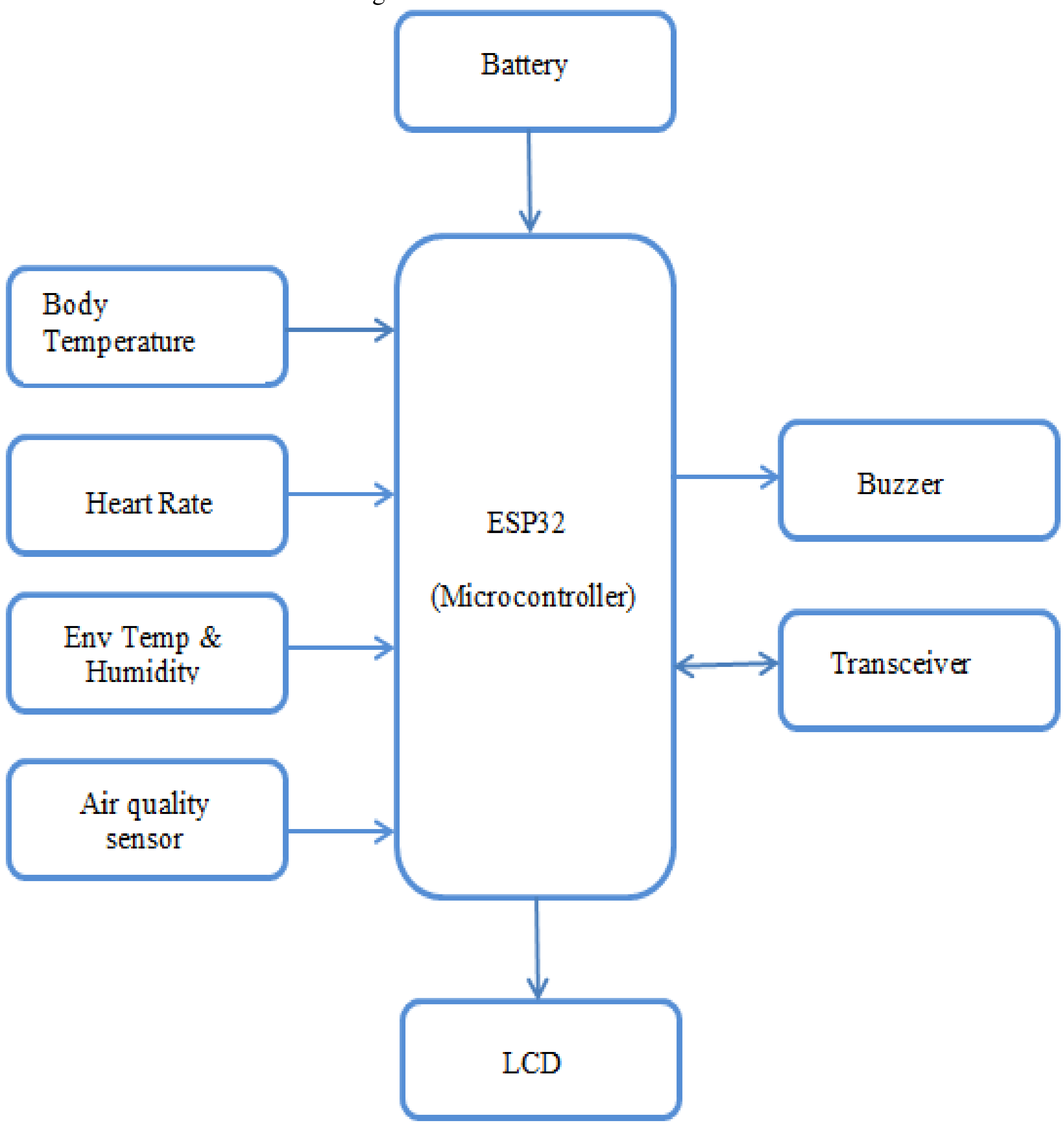

Fig 1-Block Diagram of Worker Module

The worker module is equipped with a range of environmental and biometric sensors connected to a microcontroller. These include sensors for measuring body temperature, heart rate, environmental temperature and humidity, and air quality. All these inputs are processed in real-time and displayed on a local screen. A transceiver is used to wirelessly communicate data to the central control unit, while a buzzer is triggered during emergencies to alert nearby personnel. The entire system is powered by a compact battery, allowing it to operate efficiently on-site. This setup ensures constant monitoring of each worker's health and surroundings, making it suitable for dynamic and hazardous construction environments.

B. Block Diagram Of Control Center Module

The block diagram of Control center module is shown in Figure 1.2.

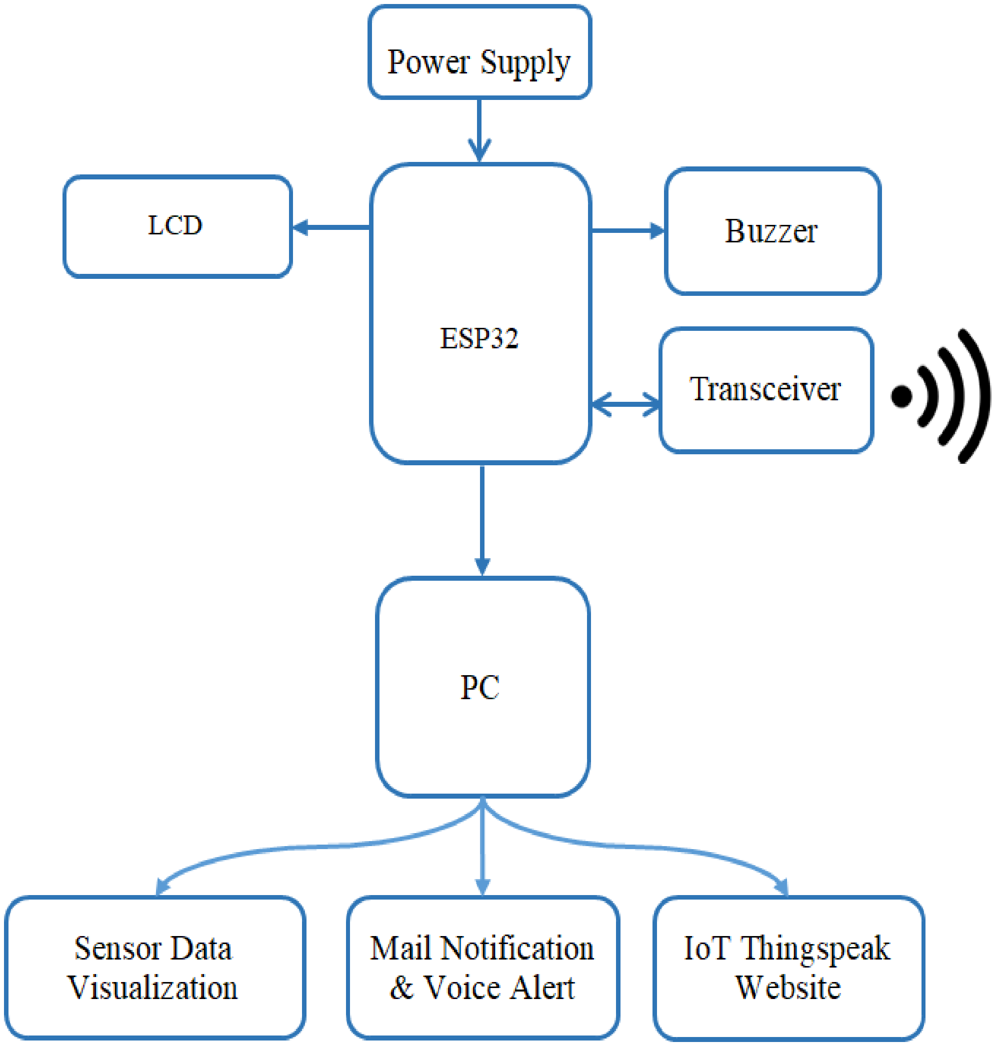

Fig 2-Block diagram of Control Centre Module

The control center module acts as the centralized monitoring and alert management hub. It receives data from the transceiver via wireless communication and processes it using another microcontroller. The system interfaces with a computer to visualize sensor data, generate voice alerts, and send automated emails in the event of abnormal readings. It uploads data in real-time to a remote access platform where it can be analyzed. By combining health monitoring and precise location tracking, the system significantly improves emergency response and contributes to a safer work environment on construction sites.

C. Simulation Diagram Of Worker Safety Monitoring System

The simulation diagram represents a worker safety monitoring system using a ESP32 microcontroller The device monitors the worker's signs and exposure to the gas with a heartbeat sensor integrated with an MQ-3 gas sensor. The collected data is displayed on an LCD screen for real-time feedback. A Zigbee module is used for wireless communication to transmit data to a central system. The virtual terminal in the simulation shows continuous live data output, validating the system's functionality.

The Simulation diagram of Worker safety monitoring system is shown in Figure 3

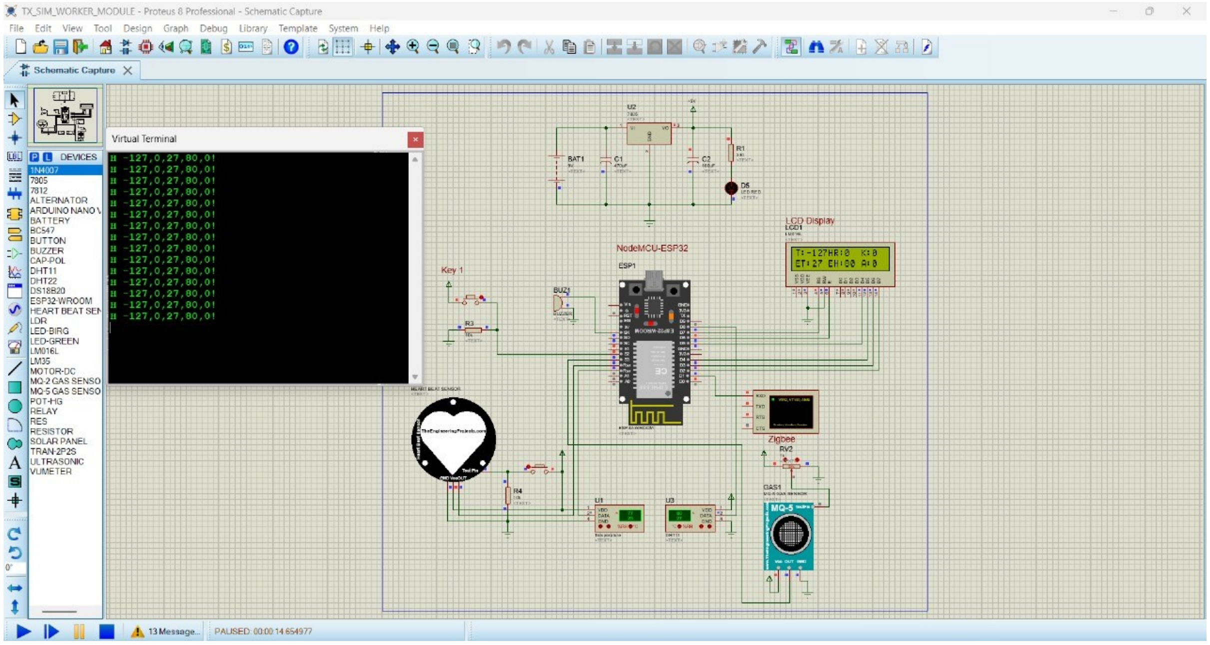

Fig 3-Simulation output of Worker safety monitoring system

D. Simulation Diagram Of Control Centre Module

The simulation diagram of control center module is shown in figure 4. The diagram represents the control center of a worker safety monitoring system using a ESP32 microcontroller. Using a Zigbee module, it collects data coming from the worker module and displays on an LCD critical parameters such as RSSI and the worker ID. A regulated power supply circuit ensures stable voltage for reliable operation. The virtual terminal outputs live communication data, helping in system verification and debugging.

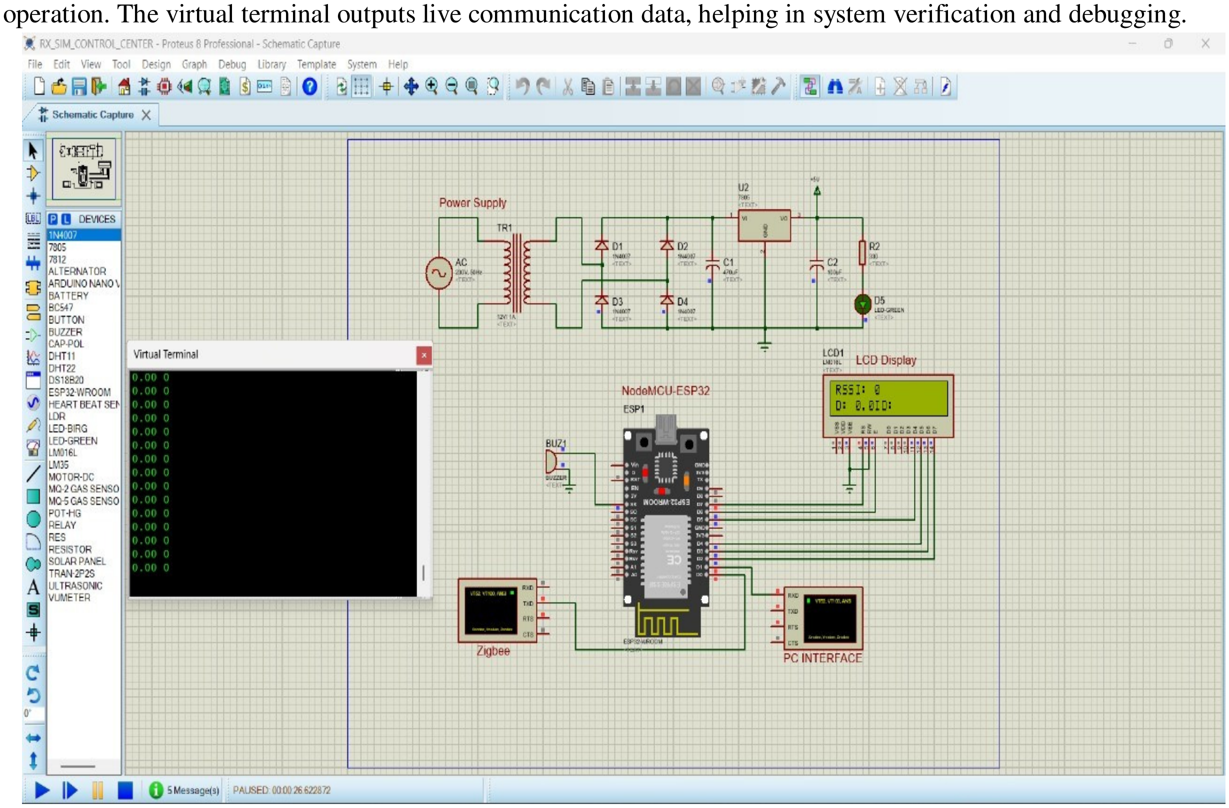

Fig 4-Simulation diagram of Control center module

V. IMPLEMENTATION

This project focused on integrating multiple sensors and communication modules into a single wearable device to monitor the health and safety condition of a worker in real-time. The worker module based on the ESP32 microcontroller gathers data from heartbeat sensor, body temperature sensor (DS18B20), environmental sensor (DHT11), and gas sensor (MQ-5). These readings are shown on a 16x2 LCD, and then sent to a control center module through periodic wireless transmission using NRF24L01 transceiver. The control center gathers this information, estimates the distance based on received signal strength indicator (RSSI) calculation, and also stores it. A python script executed on a PC captures the serial output, and actively monitors for defined thresholds such as elevated body temperature or high CO2 levels, triggering email alerts via the yagmail library as conditions dictate. Moreover, the system sends the collected data to ThingSpeak for cloud-based visual data representation. The complete system was first designed in Proteus for simulation purposes and later put on real hardware to test the alert functionality and communication checks. With these steps, a dynamically scalable IoT safety monitoring system aimed at hazardous environments like mines and construction sites can be achieved.

VI. HARDWARE DESCRIPTION

A. ESP32

The ESP32 is a powerful microcontroller with integrated Wi-Fi and Bluetooth capabilities, making it highly suitable for IoT applications. In this project, the ESP32 acts as the central processing unit, interfacing with multiple sensors to gather real-time data. Its dual-core processor and high processing speed allow it to efficiently handle data from environmental sensors, human detection modules, and communication units. One core can be dedicated to reading sensor inputs, while the other manages wireless data transmission or decision-making processes. The ESP32's built-in Bluetooth Low Energy (BLE) capabilities are particularly crucial in this project, enabling it to detect signals from BLE beacons worn by individuals. These signals provide Received Signal Strength Indicator (RSSI) values, which are analyzed to estimate the distance between the victim and the sensor, facilitating triangulation for accurate location detection.

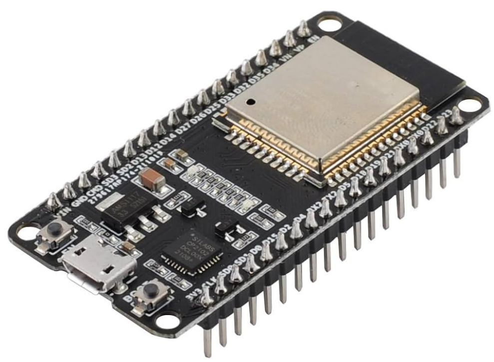

Fig 5-ESP32

The ESP32 also manages data communication with a PC or cloud server via a serial interface. It transmits sensor data to visualization dashboards, sends notifications, and uploads readings to platforms like ThingSpeak. Furthermore, the ESP32 controls local output devices like buzzers and LCDs based on sensor inputs, acting instantly when life signs are detected. Its low power consumption, cost-effectiveness, and versatility make it an ideal choice for building a responsive, scalable, and robust disaster management system. The ESP32's GPIO pins support various analog and digital sensors, enabling a modular system design where additional sensors or modules can be easily integrated depending on deployment needs. The ESP32 is a highly versatile microcontroller, and its working process begins as soon as it is powered on. Once the device is supplied with power, it starts the boot process, where the system checks the boot mode and either runs the application stored in memory or enters a bootloader mode for flashing firmware. After booting up, the system configures its pins for various tasks, such as input, output, communication protocols, and analog signal processing. The microcontroller's cores are responsible for executing the main program, and it can perform tasks such as sensor data acquisition, motor control, and communication with other devices.

The ESP32 is particularly strong in connectivity and can establish Wi-Fi or Bluetooth connections to communicate with external devices or networks. The microcontroller can send and receive data, either via its built-in communication interfaces or by acting as a server for IoT applications. It can also manage real-time data processing, store data in flash memory, and trigger actions based on sensor readings.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

The chip supports low power operation, allowing it to be used in battery-powered devices. Additionally, the ESP32 is capable of handling firmware updates remotely, making it suitable for applications where physical access to the device is limited.

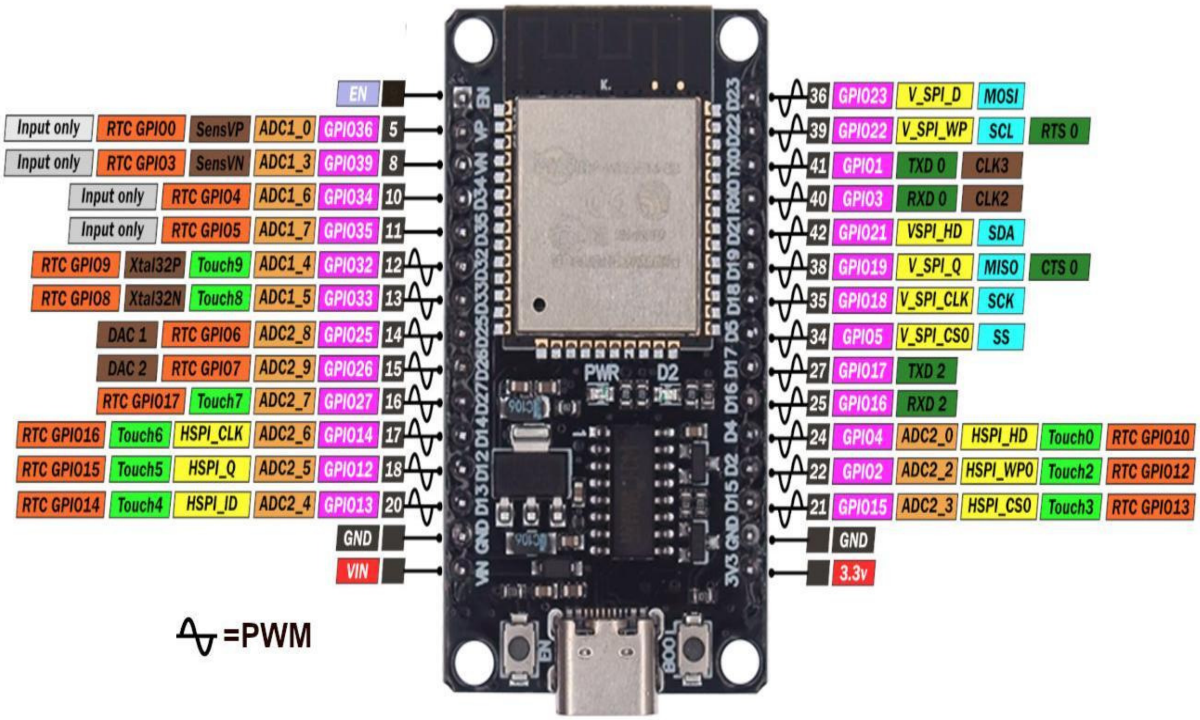

Fig 6-ESP32 Pin Configuration

B. Body Temperature Sensor

A body temperature sensor is a vital component in health monitoring systems, especially in environments where rapid detection of abnormal physiological changes is critical, such as construction sites. These sensors are designed to detect and measure the surface temperature of the human body with precision. They typically use thermistors or infrared technology to capture thermal readings from the skin, which are then processed and interpreted by a microcontroller like the ESP32. Some advanced sensors even use non-contact methods, which minimize discomfort and are suitable for continuous monitoring applications.

In occupational safety systems, body temperature readings serve as an early warning signal for conditions such as heat exhaustion, fever, or infection. For example, construction workers operating in hot environments are at high risk of heatstroke. A continuous monitoring system with a temperature sensor can detect rising body temperature early and trigger an alert, helping supervisors take preventive action before the situation becomes critical.

Similarly, in pandemic or health-sensitive settings, temperature monitoring helps identify workers who may have a fever, thereby reducing the risk of contagious disease spread at the workplace.

From a technical standpoint, the integration of body temperature sensors into wearable devices or helmets makes the system non-intrusive and efficient. These sensors provide analog or digital output signals which can be easily interpreted by a microcontroller. Coupled with a wireless transceiver, the data can be transmitted to a central control unit for further analysis and decision-making. The ability to store and review historical temperature data also enables long-term health trend analysis, supporting proactive healthcare interventions.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

Fig 7-Body Temperature Sensor

C. .Heart Beat Sensor

The heartbeat sensor, or heart rate monitor, is another crucial component in real-time health monitoring systems. It is used to measure the pulse rate of a person, which is an important indicator of cardiovascular health and stress levels. These sensors work by detecting the changes in blood volume through optical methods. Typically, they use photoplethysmography (PPG) technology, which involves a light source (usually infrared or green LED) and a photodetector to measure blood flow. As the heart beats, the volume of blood in the tissue changes, and this variation is picked up by the sensor.

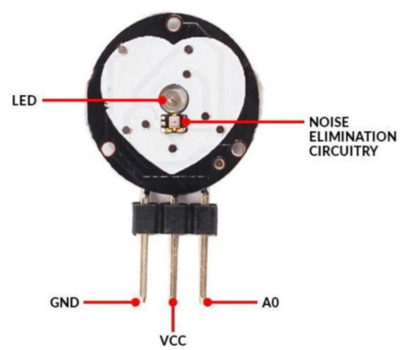

Fig 8-Heart Beat Sensor

In the context of construction site safety, heartbeat sensors help in detecting conditions such as arrhythmias, high stress, fatigue, or even early signs of heart attacks. Workers engaged in physically demanding tasks are prone to cardiovascular strain, and continuous heart rate monitoring can flag unusual readings in real-time. When paired with alert systems such as buzzers or transceivers, these sensors can notify medical personnel or supervisors instantly, allowing timely medical intervention and potentially saving lives.

The data collected from heartbeat sensors is typically fed into a microcontroller like the ESP32, where it can be filtered and analyzed using algorithms to distinguish between normal and abnormal readings. These sensors are compact and can be embedded into wearables like smart bands, chest straps, or helmets, offering a seamless user experience. Their integration into an IoT-based framework allows remote monitoring and data logging, which is beneficial for long-term health analysis and workplace safety evaluations.

D. Environment Temperature Sensor

Sensors for temperature detection in the environment have a critical function to monitor the working conditions of employees.These sensors are designed to detect the temperature of the environment rather than the body and are critical for assessing thermal comfort, preventing heat stress, and ensuring compliance with occupational safety standards. Common sensors used include thermistors, thermocouples, or digital sensors like the DHT11 or DHT22, which offer reliable and accurate readings across a wide range of temperatures.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

For construction workers, particularly those working outdoors or in enclosed, high-heat environments such as tunnels or boiler rooms, knowing the surrounding temperature is essential. Excessive ambient temperatures can lead to dehydration, fatigue, and reduced work efficiency, while extremely cold conditions can result in hypothermia or frostbite. By continuously monitoring environmental temperature, site managers can make informed decisions about work schedules, hydration breaks, and safety protocols.

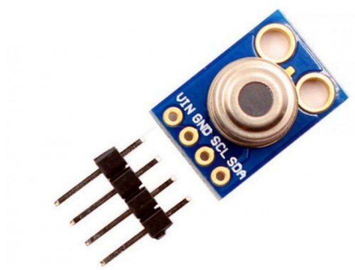

Fig 9-Environmental Temperature Sensor

These sensors are easily interfaced with microcontrollers, which process the data and either display it locally or transmit it to a control center via a wireless transceiver. When integrated into a larger system, the environmental data can be cross-referenced with body temperature and heart rate to gain a more holistic view of the worker’s health and safety conditions. This data can also be uploaded to cloud platforms for remote access, trend analysis, and regulatory compliance documentation.

E. Humidity Sensor

Humidity sensors are electronic instruments used to determine how much moisture is present in the atmosphere.These sensors are especially important in health and safety monitoring systems, as high humidity levels can intensify heat stress, affect respiratory conditions, and degrade air quality. There are two main types of humidity sensors: capacitive and resistive. Capacitive humidity sensors measure changes in electrical capacitance caused by moisture in the air, while resistive sensors detect changes in electrical resistance due to humidity levels.

On construction sites, where physical exertion is high and workers often wear protective gear, high humidity can contribute to discomfort and exacerbate heat-related illnesses. Excessive moisture in the air also affects the drying time of building materials and can contribute to growth, posing both health and structural risks. By monitoring humidity in real-time, proactive steps can be taken to ensure workers operate in safe and comfortable conditions. It also helps in scheduling tasks that are sensitive to environmental conditions, like painting or plastering. Humidity data, when paired with temperature data, gives a better understanding of the heat index—a combined metric that reflects how hot it feels. Microcontrollers can use this combined information to trigger automated alerts, display warnings on LCDs, or log data for compliance. Combined with other health metrics, humidity readings contribute significantly to creating an intelligent, responsive safety monitoring system.

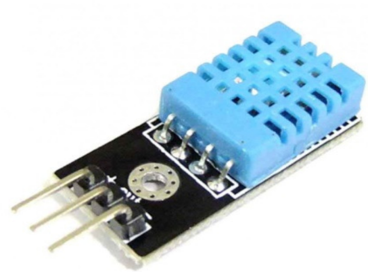

Fig 10-Humidity Sensor

F. Air Quality Sensor

Air quality sensors are devices that measure pollutants in the air, such as particulate matter (PM2.5, PM10), carbon monoxide (CO), volatile organic compounds (VOCs), and other harmful gases. These sensors are particularly crucial on construction sites where dust, chemical fumes, and machinery emissions can pose serious respiratory risks. Sensors like the MQ-series or SDS011 are commonly used for detecting gas concentrations and airborne particles in real-time.

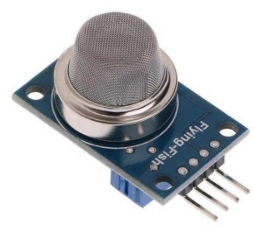

Fig 11-Air Quality Sensor

Poor air quality can lead to chronic respiratory issues, eye irritation, fatigue, and in severe cases, occupational asthma or long-term lung damage. For workers exposed to cement dust, paint fumes, and other hazardous substances, monitoring air quality is essential. Continuous real-time monitoring enables early detection of unsafe air conditions and helps enforce protective measures like mask usage or ventilation system activation. This not only safeguards the health of workers but also ensures compliance with environmental and workplace safety regulations. The air quality data is typically processed by a microcontroller and can be displayed locally or transmitted to a control center. Advanced systems can even activate alarms or shutdown procedures automatically if pollutant levels exceed thresholds. Integration with cloud platforms enables remote monitoring and data logging, allowing supervisors and health officers to analyze long-term exposure risks and plan better workplace safety strategies.

G. Transceiver

A transceiver is a device that both transmits and receives data, and it plays a central role in wireless communication systems. In the context of health and safety monitoring on construction sites, transceivers are responsible for sending sensor data from the worker module to a central control unit or cloud-based server. Technologies like Zigbee, LoRa, or Bluetooth Low Energy are often used due to their low power consumption and reliable data transmission over short to medium ranges.

Transceivers enable real-time monitoring of multiple workers simultaneously without requiring physical connectivity. This wireless approach is ideal for construction environments, where movement is constant and wiring can be impractical. Each worker’s wearable system collects health and environmental data and transmits it via the transceiver to the control center. If abnormalities are detected, the system can immediately trigger alarms, send alerts, or log the incident for review.

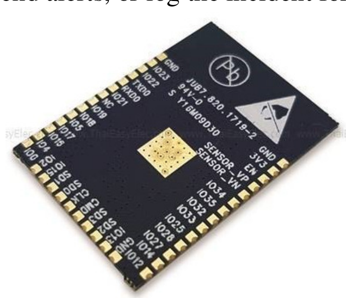

Fig 12-Transceiver

Besides basic data transmission, transceivers can support features like mesh networking, where devices relay information between one another, ensuring coverage even in complex or obstructed workspaces like multi-level buildings or tunnels. This makes the communication network more robust and fault-tolerant. The use of a reliable transceiver module ensures that no critical data is lost, maintaining the integrity and responsiveness of the health monitoring and emergency response system.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

VII. APPENDICES CODE LISTINGS

A. Worker Module

```
include <OneWire.h>
#include <DallasTemperature.h>
#include <LiquidCrystal.h>
LiquidCrystal lcd(15, 2, 0, 4, 16, 17);
#include <WiFi.h>
// Set your AP credentials
const char* ssid = "ESP32-AP";
const char* password = "12345678";
//Buzzer
int buzzer = 23;
//Temperature Sensor Define I/O
//Temperature Sensor
#define ONE_WIRE_BUS 26
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
//Humidity Sensor
#include "DHT.h"
#define DHTPIN 14    // Digital pin connected to the DHT sensor
#define DHTTYPE DHT11   // DHT 11
// Initialize DHT sensor.
DHT dht(DHTPIN, DHTTYPE);
// Airquality
int aq = 13;
int aqv= 0;
//Key
int ke = 12;
int key= 0;

// Heart Beat Sensor Variables
#include <PulseSensorPlayground.h>
//  Variables
const int PulseWire = 33;       // PulseSensor PURPLE WIRE connected to ANALOG PIN 0
int Threshold = 550;           // Determine which Signal to "count as a beat" and which to ignore.
PulseSensorPlayground pulseSensor;  // Creates an instance of the PulseSensorPlayground object called "pulseSensor"
int myBPM;
void setup() {
  // put your setup code here, to run once:
  pinMode(aq, INPUT);
  pinMode(ke, INPUT);
  dht.begin();
  pinMode(buzzer, OUTPUT);
  Serial.begin(9600);         // Setting the baudrate at 9600
    pulseSensor.analogInput(PulseWire);
  pulseSensor.setThreshold(Threshold);
  lcd.begin(16, 2);
  lcd.setCursor(0,0);
  lcd.print("HEALTH & ENVIRONM");
  lcd.setCursor(0,1);
```

```
  lcd.print(" WORKERS MODULE  ");
  delay(1500);
  // Double-check the "pulseSensor" object was created and "began" seeing a signal.
   if (pulseSensor.begin()) {
    //Serial.println("We created a pulseSensor Object !");  //This prints one time at Arduino power-up,  or on Arduino reset.
    lcd.setCursor(0,1);
    lcd.print("Sensor Begin");
   }
  lcd.clear();
  // Set up the ESP32 as an access point
  WiFi.softAP(ssid, password);
  lcd.clear();
  lcd.setCursor(0,1);
  lcd.print(WiFi.softAPIP());
  delay(500);
  digitalWrite(buzzer,HIGH);
  delay(100);
  digitalWrite(buzzer,LOW);
  lcd.clear();
}
void loop() {
  sensors.requestTemperatures();
  // Calculate the average value from all "j1" readings.
  int tempvv = sensors.getTempCByIndex(0);

  //Humidity Sensor
  delay(100);
  int ehum = dht.readHumidity();
  int etemp = dht.readTemperature();
//Airquality Sensor
  aqv = digitalRead(aq);
  //Key
  key = digitalRead(ke);
  if (pulseSensor.sawStartOfBeat()) {            // Constantly test to see if "a beat happened".
  myBPM = pulseSensor.getBeatsPerMinute();  // Calls function on our pulseSensor object that returns BPM as an "int".
    //Serial.println("♥  A HeartBeat Happened ! "); // If test is "true", print a message "a heartbeat happened".
    //Serial.print("BPM: ");                        // Print phrase "BPM: "
    //Serial.println(myBPM);                        // Print the value inside of myBPM.
    lcd.setCursor(6,0);
    lcd.print("HR:");
    lcd.print(myBPM);
    lcd.print("  ");
   }
  lcd.setCursor(0,0);
  lcd.print("T:");
  lcd.print(tempvv);
  lcd.print(" ");
  lcd.setCursor(6,0);
  lcd.print("HR:");
  lcd.print(myBPM);
```

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

```
  lcd.print("  ");
  lcd.setCursor(0,1);
  lcd.print("ET:");
  lcd.print(etemp);
  lcd.print("  ");
  lcd.setCursor(6,1);
  lcd.print("EH:");
  lcd.print(ehum);
  lcd.print("  ");
  lcd.setCursor(12,1);
  lcd.print("A:");
  lcd.print(aqv);
  lcd.print("  ");
  if(tempvv  >= 39)
  {
   digitalWrite(buzzer,HIGH);
   delay(100);
   digitalWrite(buzzer,LOW);
   lcd.setCursor(12,0);
   lcd.print("BT H");
  }
  else if (aqv == 0 )
  {
   digitalWrite(buzzer,HIGH);
   lcd.setCursor(12,0);
   lcd.print("AQ L");
   delay(200);
   digitalWrite(buzzer,LOW);
  }
  else
  {
   lcd.setCursor(12,0);
   lcd.print("NOR ");
   digitalWrite(buzzer,LOW);
  }
  Serial.print("H ");
  Serial.print(tempvv);
  Serial.print(",");
  Serial.print(myBPM);
  Serial.print(",");
  Serial.print(etemp);
  Serial.print(",");
  Serial.print(ehum);
  Serial.print(",");
  Serial.print(aqv);
  Serial.print("!");
  Serial.println();
  delay(1000);
}
```

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

B. Control Centre Module

```
#include <WiFi.h>
#include <LiquidCrystal.h>
LiquidCrystal lcd(7,6,5,4,3,2);
const char* ssid = "IoTData";
const char* password = "123456789";
int buz = 13;
//Serial Data Clear
int incomingByte = 0;
float estimatedDistance = 0;
int rssi = 0;
String ssidr;
void setup() {
 pinMode(buz, OUTPUT);
  Serial.begin(9600);
  lcd.begin(16, 2);
  lcd.setCursor(0, 0);
  lcd.print("  SMART RESCUE");
  lcd.setCursor(0, 1);
  lcd.print("RSSI BASED FINDER");
      }
//Serial.println("Connected to WiFi!");
  delay(100);
  digitalWrite(buz, HIGH);
  delay(50);
  digitalWrite(buz, LOW);
  lcd.clear();
}
void loop() {
  lcd.setCursor(0, 1);
  lcd.print("D: ");
  lcd.print(estimatedDistance);
  lcd.print(" ");
  lcd.setCursor(0, 0);
  lcd.print("RSSI: ");
  lcd.print(rssi);
  lcd.setCursor(6, 1);
  lcd.print("ID: ");
  lcd.print(ssidr);
  Serial.print(estimatedDistance);
  Serial.print(" ");
  Serial.print(rssi);
  Serial.print(" ");
  Serial.print(ssidr);
  Serial.println();
  delay(100); // Scan every 1 seconds
}

float calculateDistance(int rssi) {
  float A = -50;  // RSSI value at 1 meter (this is a calibration constant)
```

```
  float n = 2.5;  // Path loss exponent (this may need calibration for your environment)
  return pow(10, (A - rssi) / (10 * n));
}
```

C. Centre Monitoring And Alert System

```
import os
import time
import serial
import urllib.request
import yagmail
import pyttsx3
try:
    arduino = serial.Serial('COM5', 9600, timeout=1)
    time.sleep(10)
    print(" Arduino connected successfully.")
except serial.SerialException as e:
    print(f" Error connecting to Arduino: {e}")
    exit()
count = 0  # To avoid repeated emails
def mail_send(data):
    try:
        print("📨 Sending mail...")
        mail = 'sampletestmail.786@gmail.com'
password = 'vnzymrxlzmdyurch'
        recipients = ["anushsathyan@gmail.com"]
yag = yagmail.SMTP(mail, password)
        for dest in recipients:
            yag.send(
                to=dest,
                subject="Alert...! Abnormal Detected..",
                contents=data,
            )
        print(" Mail sent to all.")
    except Exception as e:
        print(f" Error sending email: {e}")
while True:
    try:
        raw_data = arduino.readline().decode('utf-8').strip()
if not raw_data:
            continue
data_v = raw_data.split()
        print(f" Received: {data_v}")
if len(data_v) < 8:
            print(f"⚠ Warning: Incomplete data received. Length = {len(data_v)}")
            continue
try:
            # Parse and convert values
            data1i = int(data_v[0])  # Body Temp
            data2i = int(data_v[1])  # Heart Rate
```

```
        data3i = int(data_v[2])  # Envi Temp
        data4i = int(data_v[3])  # Envi Humidity
        data5i = int(data_v[4])  # Airquality
        data6f = float(data_v[5])  # Distance
        data7i = int(data_v[6])  # RSSI
        data8s = str(data_v[7])   # ID
 # Display readings
        print(f" Body Temperature : {data1i}")
        print(f" Heart Rate       : {data2i}")
        print(f"Envi. Temperature: {data3i}")
        print(f" Envi. Humidity   : {data4i}")
        print(f" Airquality       : {data5i}")
        print(f" Distance         : {data6f}")
        print(f" RSSI             : {data7i}")
        print(f" ID               : {data8s}")
# Compose email content
        email_content = (
          f"<h1>Human Rescue System Details...!</h1>"
          f"<h2> Body Temperature: {data1i} "
          f"Heart Rate: {data2i} "
          f"Envi. Temperature: {data3i} "
          f"Envi. Humidity: {data4i} "
          f"Airquality: {data5i} "
          f"Distance: {data6f} "
          f"RSSI: {data7i} "
          f"ID: {data8s} - Abnormal </h2>"
        )
# Send to ThingSpeak
        try:
          link = (
            f"https://api.thingspeak.com/update?api_key=LUWXR4Y499VYJNA8"
            f"&field1={data1i}&field2={data2i}&field3={data3i}&field4={data4i}"
            f"&field5={data5i}&field6={data6f}&field7={data7i}"
          )
          urllib.request.urlopen(link)
        except Exception as e:
          print(f" Error sending to ThingSpeak: {e}")
# Abnormal conditions
        abnormal = False
        engine = pyttsx3.init()
        if data1i >= 39:
          print("⚠ Body Temperature High - Abnormal")
          engine.say("Body Temperature High - Abnormal - Take Care!")
          engine.runAndWait()
          abnormal = True
        if data2i < 50 or data2i > 100:
          print("⚠ Heart Rate Abnormal")
          engine.say("Heart Rate Abnormal - Please Check!")
          engine.runAndWait()
```

```
                abnormal = True
            if data5i == 0:
                print("⚠ Airquality LOW - Abnormal")
                engine.say("Airquality Low - Abnormal - Take Care!")
                engine.runAndWait()
                abnormal = True
            # Trigger email alert once per abnormal cycle
            if abnormal and count == 0:
                mail_send(email_content)
                count = 1
            elif not abnormal:
                print(" Status: NORMAL")
                count = 0
            print('')
        except ValueError as e:
            print(f" ValueError: Unable to convert data. Skipping. Raw data: {data_v} | Error: {e}")
            continue
    except Exception as e:
        print(f" Unexpected error: {e}")
        continue
```

VIII. TEST RESULTS AND LOGS

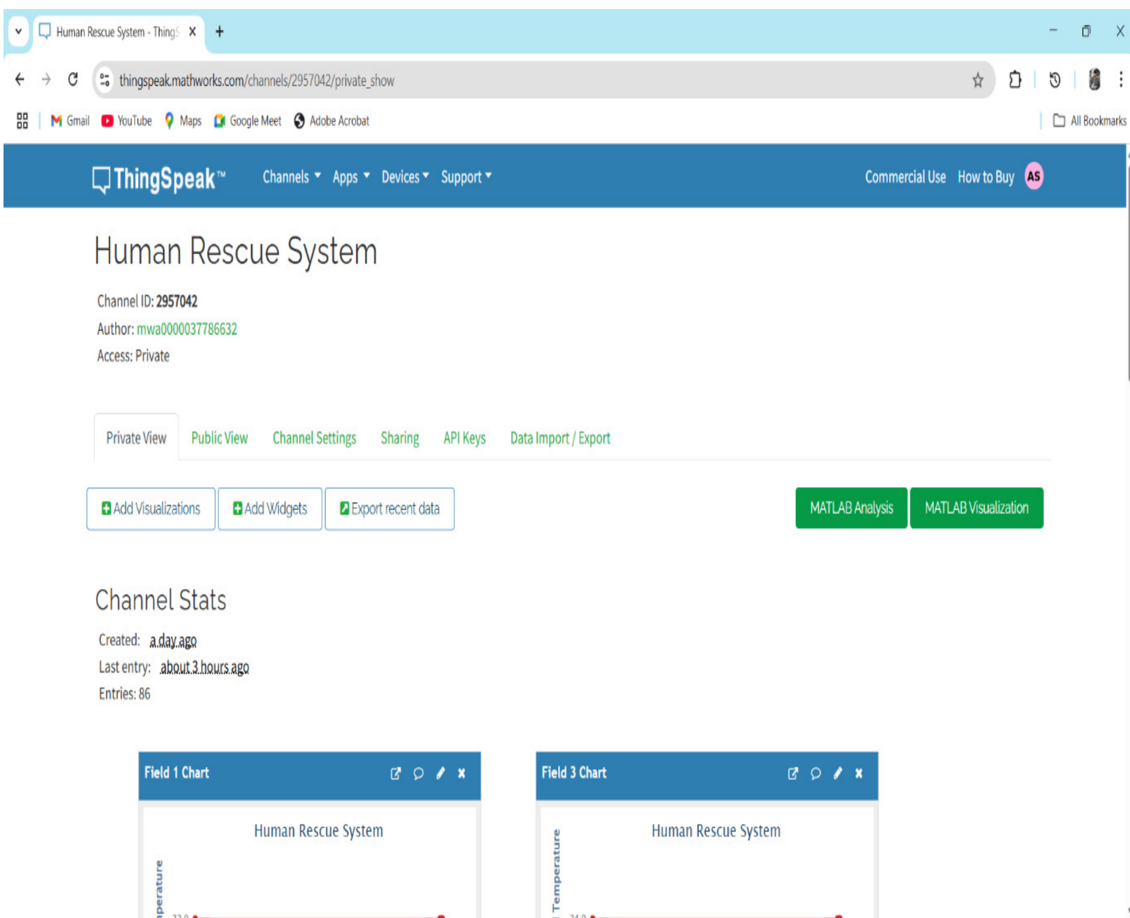

Fig 12-ThinkSpeak Website

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

A. Body Temperature Sensor

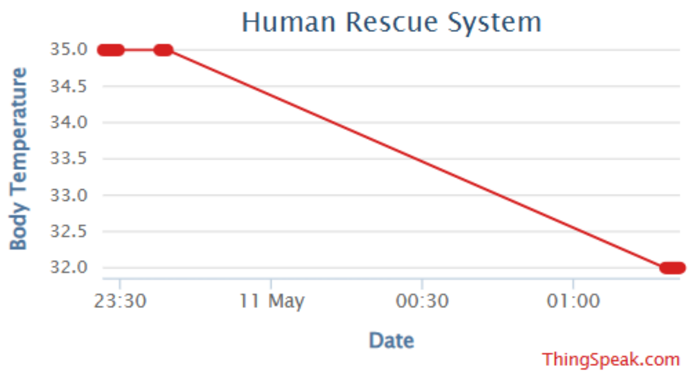

Fig 13-Body Temperature Monitoring using NRF sensor on ThinkSpeak

The chart from ThingSpeak shows body temperature readings collected in real time using the DHT11 sensor. An initial reading of 35°C is recorded, followed by a gradual drop to 32.5°C. This indicates potential distress or a cooling effect on the worker. Such data helps monitor hypothermia risk in harsh environments. Real-time plotting helps supervisors act quickly.

B. Heart Rate Sensor

The graph displays heart rate data from the sensor. A sudden spike to 78 bpm followed by a sharp drop suggests a brief physical exertion or stress response. Continuous tracking of heartbeats can detect irregularities or collapse. Alerts are triggered if values cross safety thresholds. These fluctuations support real-time health assessment.

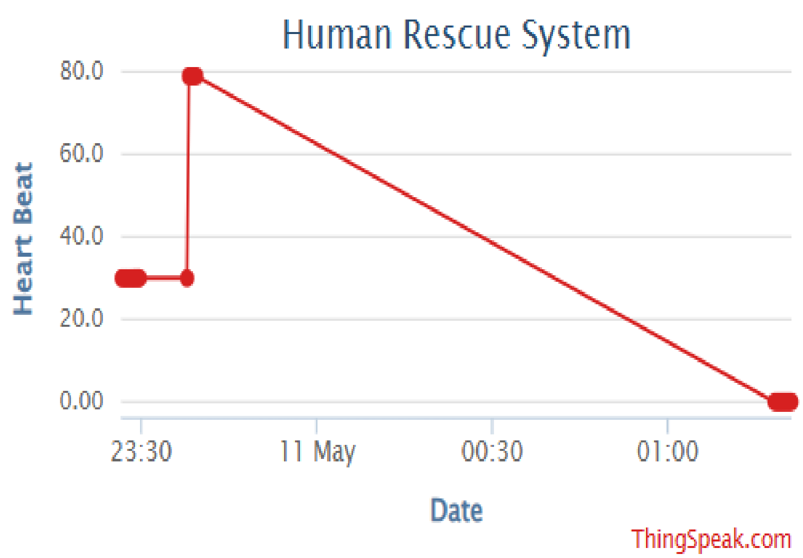

Fig 14-Heart Rate Monitoring using Heartbeat Sensor on ThinkSpeak

C. Environmental Temperature

This plot tracks environmental temperature via the DHT11 sensor. The temperature decreases from 36°C to 34°C, reflecting environmental cooling. Such values help correlate with worker's body response. The data is useful for identifying temperature drops during night shifts or underground conditions. The sensor ensures ambient environment safety.

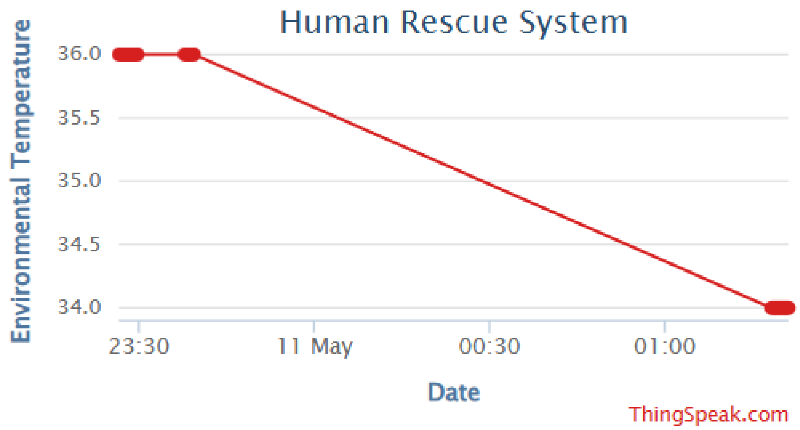

Fig 15-Environmental Temperature Logging via DHT11 sensor

D. Humidity Monitoring

Environmental humidity levels rise from 60% to 64%, possibly due to poor ventilation. The humidity data adds to heat index calculations. High humidity combined with high temperature increases heat stress risk. Monitoring this helps in adjusting ventilation or hydration levels. Real-time monitoring is critical for long shifts.

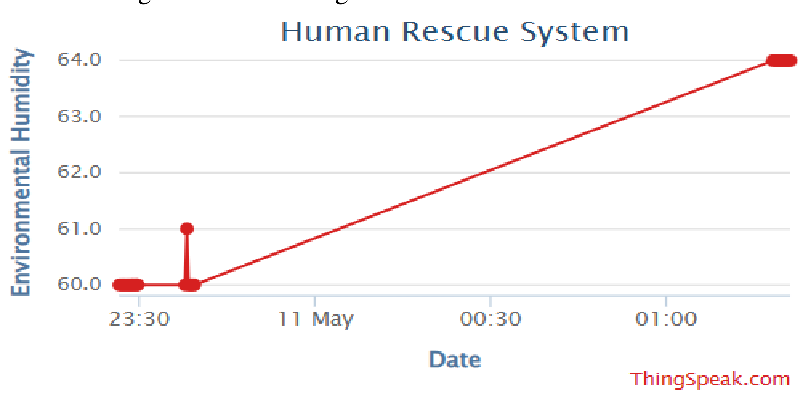

Fig 16-Humidity Variation Monitoring using DHT11 Sensor

E. Air-Quality Monitoring

This graph shows constant air quality levels (value = 1), captured using the MQ-5 gas sensor. No major fluctuations were detected during the logged period. A steady reading indicates no gas leak or pollutant presence. Alerts will activate upon detecting harmful gases like LPG or CO. Ensures safe breathing conditions for workers.

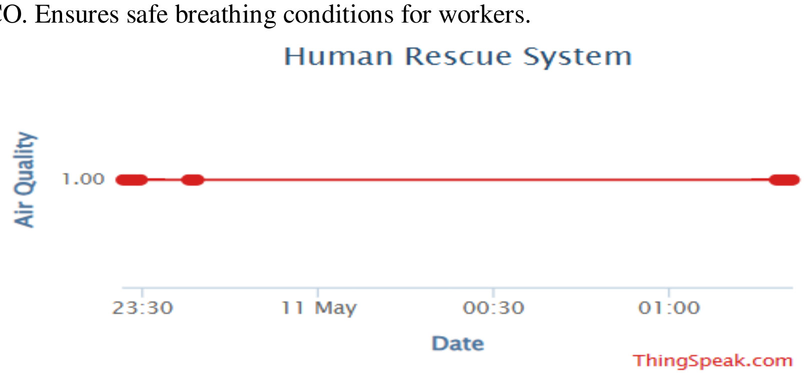

Fig 17- Air Quality Level Reading using MQ-5 Gas Sensor

F. Distance Monitoring

Distance data reflects RSSI signal strength converted to estimated distance from the control center. Fluctuations from 20 cm to 65 cm are observed. These values are used for localization or collapse detection. Constant tracking helps in sending rescue teams to exact locations. Sudden change may indicate a fall or movement.

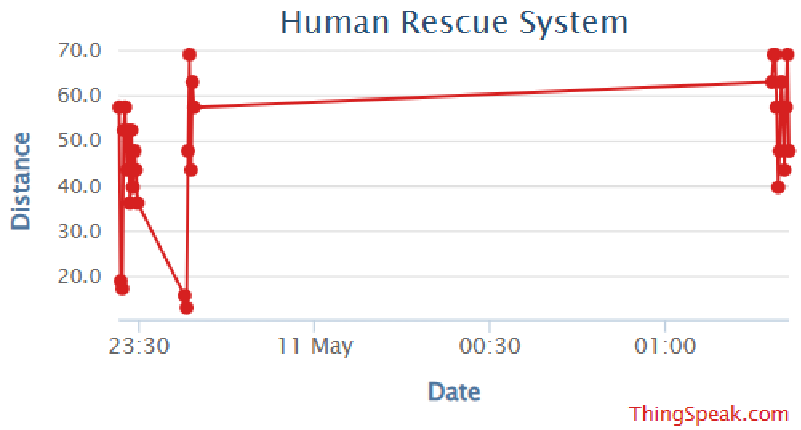

Fig 18- Distance Measurement Based on RSSI Signal Strength

G. Rssi Monitoring

This graph displays RSSI signal values indicating worker proximity to access point. A sudden dip to -94 dBm triggered the alert mail, indicating abnormal status.

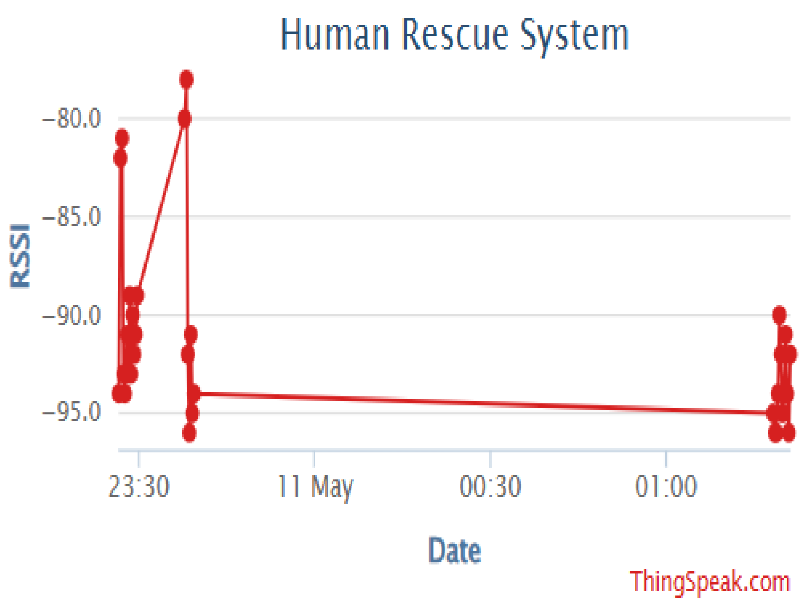

Fig 19-RSSI Monitoring

H. ALERT LOG

The email log confirms an emergency was detected and alert sent to the control center. It includes all sensor data, signal strength, and network details. Enables immediate response via alert system.

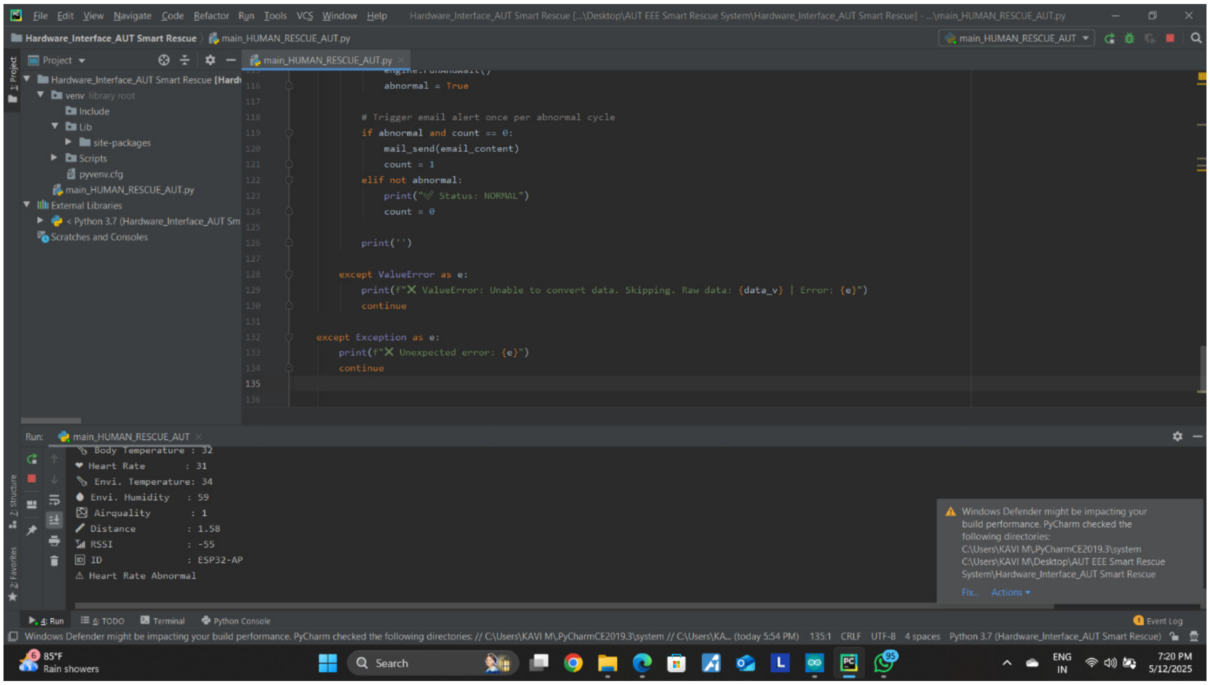

Fig 20-Abnormal Condition Detection Interface-Alert display

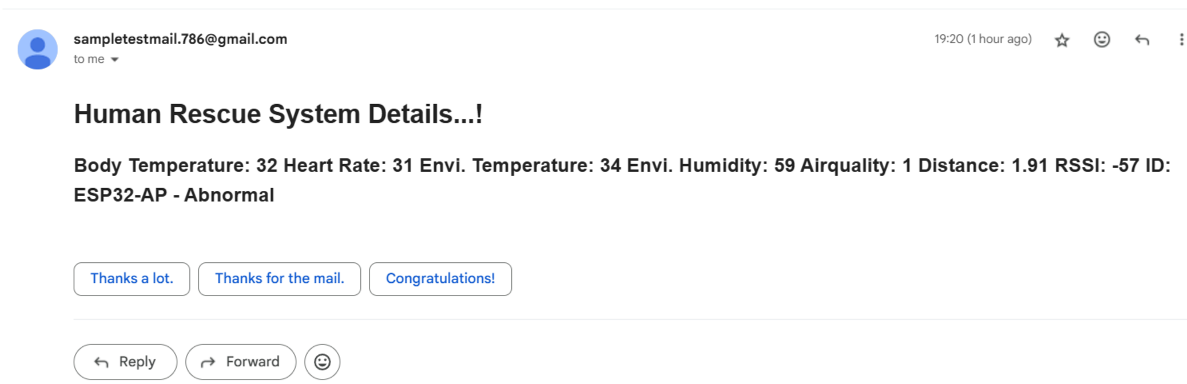

Fig 21-Emergency Email Alert Log

IX. CONCLUSION

The Smart Employee Health Monitoring and Emergency Rescue System has demonstrated its effectiveness in improving worker safety and enhancing emergency response capabilities on construction sites. By leveraging Received Signal Strength Indicator (RSSI) technology for precise indoor positioning and integrating real-time health monitoring through ESP32 and Zigbee-based Wireless Sensor Network (WSN) modules, the system provides reliable, low-power communication and efficient tracking of workers. The system's ability to continuously monitor workers' vital signs and instantly relay location data during emergencies contributes to faster and more accurate rescue operations, ultimately enhancing overall safety in high-risk construction environments. In this system represents a significant advancement in addressing the critical issue of worker safety on construction sites. It combines health monitoring with real-time tracking, ensuring that potential health risks are detected early and that workers can be located and assisted quickly in emergencies. The successful implementation and testing of the system suggest that it can provide substantial benefits in terms of reducing the likelihood of fatalities and severe injuries on construction sites.

REFERENCES

[1] Chen, Haiyan, et al. "The impact of wearable devices on the construction safety of building workers: a systematic review." Sustainability 15.14 (2023): 11165.

[2] Ding, Yifei, Yuxin Zhang, and Xinyan Huang. "Intelligent emergency digital twin system for monitoring building fire evacuation." Journal of Building Engineering 77 (2023): 107416.

[3] Kang, Jian, et al. "Improving the emergency management of energy infrastructure using scenario construction." International Journal of Hydrogen Energy 48.23 (2023): 8731-8742.

[4] Lee, Poizot Li. "Enhancing Construction Worker Safety: Real-time Health Monitoring through Wearable Technology." International Journal of Enterprise Modelling 17.3 (2023): 139-147.

[5] Rane, Nitin, Saurabh Choudhary, and Jayesh Rane. "Leading-edge wearable technologies in enhancing personalized safety on construction sites: a review." Available at SSRN 4641480 (2023).

[6] Schroth, Christian A., et al. "Emergency response person localization and vital sign estimation using a semi-autonomous robot mounted sfcw radar." IEEE Transactions on Biomedical Engineering 71.6 (2024): 1756-1769.

[7] Wang, Mingzhu, Jiayu Chen, and Jun Ma. "Monitoring and evaluating the status and behaviour of construction workers using wearable sensing technologies." Automation in Construction 165 (2024): 105555.

[8] Xu, Wenpei, and Ting-Kwei Wang. "Construction worker safety prediction and active warning based on computer vision and the gray absolute decision analysis method." Journal of Construction Engineering and Management 149.4 (2023): 04023014.

[9] Yang, Hang, and Xunbo Li. "Performance Evaluation of Three-Dimensional UWB Real-Time Locating Auto-Positioning System for Fire Rescue." Intelligent Automation & Soft Computing 37.3 (2023).

[10] Zhu, Aiyu, et al. "Deep reinforcement learning for real-time assembly planning in robot-based prefabricated construction." IEEE Transactions on Automation Science and Engineering 20.3 (2023): 1515-1526.