

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY,  
BELGAUM 59014**



Internet of Things Project Report on  
**“HEALTH SYSTEM MONITOR”**

By

**Sohan R Kumar (1BM19CS159)**

**Sampreeth P(1BM19CS142)**

**Praveen Kumar S (1BM20CS413)**

**Sushmitha Y V(1BM19CS165)**

Under the Guidance of

**Mrs. Antara Roy Choudhury**

Assistant Professor, Department of CSE  
BMS College of Engineering

IoT Application Development carried out at



Department of Computer Science and Engineering  
BMS College of Engineering  
(Autonomous college under VTU)  
P.O. Box No.: 1908, Bull Temple Road, Bangalore-560 019  
2019-2020

**BMS COLLEGE OF ENGINEERING**  
**DEPARTMENT OF COMPUTER SCIENCE AND**  
**ENGINEERING**



***CERTIFICATE***

This is to certify that the Internet of Things project titled “**HEALTH MONITOR SYSTEM**” has been carried out by **Sohan R Kumar(1BM19CS159), Sampreeth P(1BM19CS142), Praveen Kumar S(1BM20CS413) and Sushmitha Y V(1BM19C165)** during the academic year 2019-2020.

Signature of the guide

**Mrs. Antara Roy Choudhury**

Assistant Professor

Department of Computer Science and Engineering

BMS College of Engineering, Bangalore

**Examiners**

**Name**

**Signature**

1.

2.

**BMS COLLEGE OF ENGINEERING**  
**DEPARTMENT OF COMPUTER SCIENCE AND**  
**ENGINEERING**



***DECLARATION***

We, **Sohan R Kumar(1BM19CS159), Sampreeth P(1BM19CS142), Praveen Kumar S (1BM20CS413) and Sushmitha Y V (1BM19C165)** students of 5<sup>th</sup> Semester, B.E, Department of Computer Science and Engineering, BMS College of Engineering, Bangalore, hereby declare that, this IoT Application development work entitled "**HEALTH SYSTEM MONITOR**" has been carried out by us under the guidance of **Mrs. Antara Roy Choudhury**, Assistant Professor, Department of CSE, BMS College of Engineering, Bangalore during the academic semester Aug-Dec 2019

We also declare that to the best of our knowledge and belief, the development reported here is not from part of any other report by any other students.

Signature

**Sohan R Kumar(1BM19CS159)**

**Sampreeth P(1BM19CS142)**

**Praveen Kumar S(1BM20CS413)**

**Sushmitha Y V(1BM17C165)**

# **Introduction**

## **Objective of the project**

- As the name suggests, health system monitor is used to monitor the wellbeing of an individual. This system checks and assures the person's health
- This system is used to check the heart rate, spo2, temperature. In case there is any vary in the heart rate or spo2 or temperature, the same will be notified to the concerned person who then can address the issue,

## **Abstract description of the project**

The project consists of GY-MAX30100 which is a plug and play Pulse Oximeter Heart Rate Sensor it is used to incorporate live heart rate data, we must clip the sensor to earlobe or finger tip to read the pulse rate, MLX90614 IR Temperature Sensor is infrared thermometer for non-contact temperature measurement which will be used to measure the body temperature of an individual and SIM900A GSM GPRS Module is used to send alerts like call or messages in emergency situation for a specific person.

Our project consists of two modes Pulse Read Mode and Temperature Read Mode.

In Pulse Read Mode the puloxi() function is triggered and the user must place his finger tip on the Heart rate sensor , the sensor starts capturing data and gives the heart rate in beats per minute. If the beats per minute is less than 50 the system calls the emergency number using GSM Module to notify that user is in critical state. If the condition is normal, it displays heart rate of the user.

In Temperature Read Mode the temp() function is triggered and the user must bring his finger tip near the temperature sensor, the sensor captures the IR radiation form the user fingertip, makes internal conversion and displays the body temperature of the user if the temperature is greater than 100 the system sends alert SMS using GSM Module to the given number saying BODY TEMPERATURE IS HIGH.

## **Literature Survey**

Sl.No	Name of the Project or Product (Existing)	Commercial or Non-Commercial	Features
1.	Pulse-oximeter	Non-Commercial	Calculates pulse and oxygen levels in the body
2.	Non-Contact temperature sensor	Non-Commercial	Calculates body temperature.

## **Hardware and Software Requirements**

### **Hardware requirement**

- 1.Arduino Uno (Quantity-1).
2. Cable (USB 2.0 Cable Type A/B for Arduino Uno) (Quantity-1).
- 3.GY-MAX30100 Pulse Oximeter Heart Rate Sensor (Quantity1)
- 4.MLX90614 Non-Contact IR Temperature Sensor (Quantity1)
5. SIM900A GSM GPRS Module (Quantity-1).
6. 12 Volt, 2 Ampere AC adaptor (Quantity-1)
- 7.Jumper wires Male to male (Quantity-25 wires).
8. Breadboard (Quantity-1).

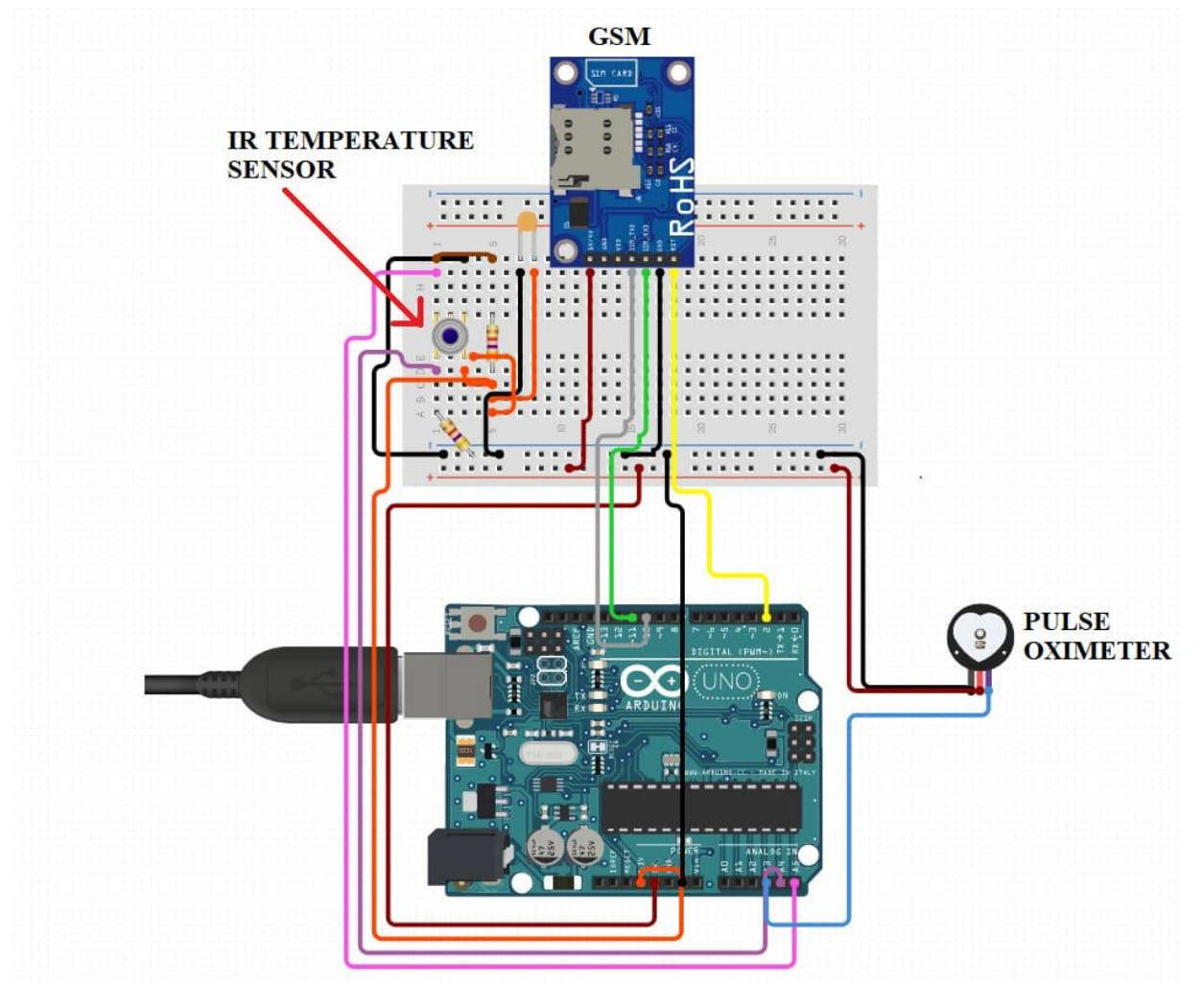
### **Software Requirements**

OS: Windows/Linux

IDE: Arduino IDE

## Design

Architectural diagram or Circuit diagram



**Explanation:**

In Pulse Read Mode the `puloxi()` function is triggered and the user must place his finger tip on the Heart rate sensor, the sensor starts capturing data and gives the heart rate in beats per minute. If the beats per minute is less than 50 the system calls the emergency number using GSM Module to notify that user is in critical state. If the condition is normal, it displays heart rate of the user.

In Temperature Read Mode the temp() function is triggered and the user must bring his finger tip near the temperature sensor, the sensor captures the IR radiation form the user fingertip, makes internal conversion and displays the body temperature of the user if the temperature is greater than 100 the system sends alert SMS using GSM Module to the given number saying **BODY TEMPERATURE IS HIGH**.

### Cost Analysis:

COMPONENT	QUANTITY	PRICE (RS.)	TOTAL PRICE (Rs.)
Aurdino UNO R3	1	708	708
GY-MAX30100 Heart Rate Sensor	1	550	550
MLX90614 IR Temperatrure Sensor	1	839	839
SIM900A GSM/GPRS	1	820	820
12V DC Apdapter	1	150	150
Jumper Wires	1	49	49
Bread Board	1	60	60
TOTAL			3176

### Source code:

```
#include <Wire.h>

#include "MAX30100_PulseOximeter.h"

#include <Adafruit_MLX90614.h>

#include <SoftwareSerial.h>


#define REPORTING_PERIOD_MS 1000

Adafruit_MLX90614 mlx = Adafruit_MLX90614();

PulseOximeter pox;

uint32_t tsLastReport = 0;

SoftwareSerial mySerial(2,3);
```

```
void setup()
{

    mySerial.begin(9600);

    delay(500);

    Serial.begin(9600);

    Serial.begin(9600);

    Serial.println("Adafruit MLX90614 test");

    mlx.begin();

    Serial.begin(115200);

    Serial.print("Initializing pulse oximeter..");

    // Initialize the PulseOximeter instance

    // Failures are generally due to an improper I2C wiring, missing power supply

    // or wrong target chip

    if (!pox.begin()) {

        Serial.println("FAILED");

        for(;;);

    } else {

        Serial.println("SUCCESS");

    }

    pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);

    // Register a callback for the beat detection

    pox.setOnBeatDetectedCallback(onBeatDetected);
```



```
}
```

```
void loop()
```

```
{
```

```
  Serial.println("Enter 'p' to check pulse/oxygen level and 't' to check temperature:")
```

```
  if (Serial.available()>0)
```

```
    switch(Serial.read())
```

```
    {
```

```
      case 'p':
```

```
        puloxi();
```

```
        break;
```

```
      case 't':
```

```
        temp();
```

```
        break;
```

```
    }
```

```
  if (mySerial.available()>0)
```

```
    Serial.write(mySerial.read());
```

```
}
```

```
void puloxi()
```

```
{
```

```
  // Make sure to call update as fast as possible
```

```
  pox.update();
```

```
  if (millis() - tsLastReport > REPORTING_PERIOD_MS) {
```

```
    Serial.print("Heart rate:");
```

```

Serial.print(pox.getHeartRate());

Serial.print("bpm / SpO2:");

Serial.print(pox.getSpO2());

Serial.println("%");

if(pox.getHeartRate() < 80 && pox.getHeartRate() > 100){

    if(pox.getSpO2() < 50){

        Serial.println("Alert!!!");

        Serial.println("CALLING.....");

        cell.println("ATD+919742980606;");

        delay(10000);

        cell.println("ATH"); // Attention Hook Control

    }

}

tsLastReport = millis();

}

}

```

```

void temp() {

    Serial.print("Ambient = "); Serial.print(mlx.readAmbientTempC());

    Serial.print("°C\tObject = "); Serial.print(mlx.readObjectTempC()); Serial.println("°C");

    Serial.print("Ambient = "); Serial.print(mlx.readAmbientTempF());

    Serial.print("°F\tObject = "); Serial.print(mlx.readObjectTempF()); Serial.println("°F");

    if(mlx.readObjectTempF() > 100){

        mySerial.println("AT+CMGF=1"); //Sets the GSM Module in Text Mode //AT+CMGF,
        SMS Format

        delay(1000); // Delay of 1000 milli seconds or 1 second
    }
}

```

```
    mySerial.println("AT+CMGS=\"+919742980606\"\\r"); // AT+CMGS, Send Message
    delay(1000);

    mySerial.println("I am SMS. Your body temperature is high!"); // The SMS text you
want to send

    delay(100);

    mySerial.println((char)26); // ASCII code of CTRL+Z , to terminate the message
    delay(1000);

}

Serial.println();

delay(500);

}

void onBeatDetected()

{

    Serial.println("Beat!");

}
```

## Device Set-up Images & Results

