HEALTH MONITORING SYSTEM

**A project report submitted for the partial fulfillment of the Bachelor of Technology Degree in Computer science & Engineering under Maulana Abul Kalam Azad University of Technology**

BY

**SHUBHAM SANGANERIA**

(ROLL NO : 1040016084 , REGISTRATION NO: 161040110503 )

**&**

Under the Guidance of:

**Prof. Dr. Sumita Ghosh**

Department of **Electronics and Communication**

For the Academic Year 2016-2020



Institute of Engineering & Management

Y-12, Salt Lake, Sector-V, Kolkata-700091

Affiliated To:



Maulana Abul Kalam Azad University of Technology

BF-142, Salt Lake, Sector I, Kolkata-700064

**CERTIFICATE**

**TO WHOM IT MAY CONCERN**

This is to certify that the project report entitled “**[PROJECT TITLE]**”, submitted by

**1. Shubham Sanganeria**

**(*Registration No: 161040110503 of 2016-2020 Roll no. 1040016084)***,

Students of **INSTITUTE OF ENGINEERING &MANAGEMENT,** in partial fulfilment of requirements for the award of the degree of **Bachelor of Technology in ……,** is a bona fide work carried out under the supervision and guidance of **Prof. Dr. ……. [Name of your project mentor]** during the final year of the academic session of 2014-2018.The content of this report has not been submitted to any other University or Institute for the award of any other degree.

It is further certified that work is entirely original and its performance has been found to be quite satisfactory.

Prof .Dr. ………

[Name of your Project Guide] Prof. Dr. ………… Project Guide H.O.D

Dept. of …….. Dept. of …..

Institute of Engineering & Management Institute of Engineering &Management

Prof. Dr.A.K.Nayak

Principal

Institute of Engineering & Management

Sector-V, Salt Lake Electronics Complex, Kolkata-700091

**ACKNOWLEDGEMENT**

We should like to take this opportunity to extend our gratitude to the following revered persons without whose immense support, completion of this project wouldn’t have been possible.

We are sincerely grateful to our advisor and mentor **[Name of your project Guide]** of the **(Department Name)**, IEM Kolkata, for his/her constant support , significant insights and for generating in us a profound interest for this subject that kept us motivated during the entire duration of this project.

We would also like to express our sincere gratitude to **Prof. Dr. Satyajit Chakrabarti** (Director**,**IEM)**, Prof. Dr. Amlan Kusum Nayak (**Principal,IEM) and **Prof. Dr Malay Ganguly** HOD of **(ECE)** and other faculties of Institute of Engineering & Management, for their assistance and encouragement.

Last but not the least, we would like to extend our warm regards to our families and peers who have kept supporting us and always had faith in our work.

**SHUBHAM SANGANERIA**

Reg. No: 10400316084

Dept. of Electronics & Communication Engineering

Institute of Engineering & Management, Kolkata

**ABSTRACT**

With an improvement in technology and miniaturization of sensors, there have been attempts to utilize the new technology in various areas to improve the quality of human life. One main area of research that has seen an adoption of the technology is the healthcare sector. The people in need of healthcare services find it very expensive this is particularly true in developing countries.

As a result, this project is an attempt to solve a healthcare problem currently society is facing. The main objective of the project was to design a remote healthcare system. It’s comprised of three main parts. The first part being, detection of patient’s vitals using sensors, second for sending data to cloud storage and the last part was providing the detected data for remote viewing. Remote viewing of the data enables a doctor or guardian to monitor a patient’s health progress away from hospital premises.

The Internet of Things (IoT) concepts have been widely used to interconnect the available medical resources and offer smart, reliable, and effective healthcare service to the patients. Health monitoring for active and assisted living is one of the paradigms that can use the IoT advantages to improve the patient’s lifestyle. In this project, I have presented an IoT architecture customized for healthcare applications. The aim of the project was to come up with a Remote Health Monitoring System that can be made with locally available sensors with a view to making it affordable if it were to be mass produced.

Hence the proposed architecture collects the sensor data through Arduino microcontroller and relays it to the cloud where it is processed and analyzed for remote viewing. Feedback actions based on the analyzed data can be sent back to the doctor or guardian through Email and/or SMS alerts in case of any emergencies.

**TABLE OF CONTENTS**

**TOPIC PAGE NUMBER CHAPTER 1. INTRODUCTION**

1.1. Motivation..................................................................................................................

1.2. Objective....................................................................................................................

1.3. Organization of the report..........................................................................................

**CHAPTER 2. LITERATURE REVIEW**

2.1. ......................................................................................

2.2. ...................................................................................................................................

**CHAPTER 3. PROPOSED STRATEGY**

3.1

3.2

**CHAPTER 4. EXPERIMENTAL SETUP AND PRACTICAL IMPLEMENTATION**

4.1

4.2

**CHAPTER 5. [IF REQUIRED] CHAPTER 6. [IF REQUIRED]**

**CHAPTER7. RESULTS................................................................................................ CHAPTER 8. CONCLUSION ...................................................................................... REFERENCES .................................................................................................................**

**CHAPTER 1**

**INTRODUCTION**

**INTRODUCTION**

A Remote health monitoring system is an extension of a hospital medical system where a patient’s vital body state can be monitored remotely. Traditionally the detection systems were only found in hospitals and were characterized by huge and complex circuitry which required high power consumption. Continuous advances in the semiconductor technology industry have led to sensors and microcontrollers that are smaller in size, faster in operation, low in power consumption and affordable in cost. The use of sensors detects the conditions of the patient and the data is collected and transferred using a microcontroller. Doctors and nurses need to visit the patient frequently to examine his/her current condition. In addition to this, use of multiple microcontroller based intelligent system provides high-level applicability in hospitals where many patients must be frequently monitored. For this, here we use the idea of network technology with wireless applicability, providing each patient a unique ID by which the doctor can easily identify the patient and his/her status of health parameters. Using the proposed system, data can be sent wirelessly to the Patient Monitoring System, allowing continuous monitoring of the patient. . Contributing accuracy in measurements and providing security in proper alert mechanism give this system a higher level of customer satisfaction and low-cost implementation in hospitals. Thus, the patient can engage in his daily activities in a comfortable atmosphere where distractions of hardwired sensors are not present. Physiological monitoring hardware can be easily implemented using simple interfaces of the sensors with a Microcontroller and can effectively be used for healthcare monitoring. This will allow development of such low-cost devices based on natural human-computer interfaces.

**1.1 MOTIVATION**

Giving care and health assistance to the bedridden patients at critical stages with advanced medical facilities have become one of the major problems in the modern hectic world. In hospitals where many patients whose physical conditions must be monitored frequently as a part of a diagnostic procedure, the need for a cost-effective and fast responding alert mechanism is inevitable. Thus our effort to design our very own system.

**Why Internet Of Things(IoT)?**

The Internet of Things (IoT) platform offers a promising technology to achieve the healthcare services, and can further improve the medical service systems. IoT wearable platforms can be used to collect the needed information of the user and its ambient environment and communicate such information wirelessly, where it is processed or stored for tracking the history of the user. Such a connectivity with external devices and services will allow for taking preventive measure (e.g., upon foreseeing an upcoming heart stroke) or provide immediate care (e.g., when a user falls and needs help).

**1.2 OBJECTIVE**

Here the main objective is to design a Remote Patient Health Monitoring System to diagnose the health condition of the patient. Proper implementation of such systems can provide timely warnings to the medical staffs and doctors and their service can be activated in case of medical emergencies. Present-day systems use sensors that are hardwired to a PC next to the bed.

The system designed comprised :-

1.Sensors which would collect the temperature, heart rate of the patient and display the same in the Liquid Crystal Display(LCD) attached to the same.

2. A cloud server hosted on ThingSpeak which would help in real time monitoring of the data with graphs and charts which would help the doctors and health experts in better analysis of the patient’s health condition.

**CHAPTER 2**

**Literature Review**

## 2.1: Intelligent wireless mobile patient monitoring system

Nowadays, Heart-related diseases are on the rise. Cardiac arrest is quoted as the major contributor to the sudden and unexpected death rate in the modern stress filled lifestyle around the globe. A system that warns the person about the onset of the disease earlier automatically will be a boon to the society. This is achievable by deploying advances in wireless technology to the existing patient monitoring system. This paper proposes the development of a module that provides mobility to the doctor and the patient, by adopting a simple and popular technique, detecting the abnormalities in the bio signal of the patient in advance and sending an SMS alert to the doctor through Global System for Mobile(GSM) thereby taking suitable precautionary measures thus reducing the critical level of the patient. Worldwide surveys conducted by World Health Organization (WHO) have confirmed that the heart-related diseases are on the rise. Many of the cardiac-related problems are attributed to the modern lifestyles, food habits, obesity, smoking, tobacco chewing and lack of physical exercises etc. The post-operative patients can develop complications once they are discharged from the hospital. In some patients, the cardiac problems may reoccur, when they start doing their routine work. Hence the ECG of such patients needs to be monitored for some time after their treatment. This helps in diagnosing the improper functioning of the heart and take precautions. Some of these lives can often be saved if acute care and cardiac surgery is provided within the so-called golden hour. So, the need for advice on first-hand medical attention and promotion of good health by patient monitoring and follow-up becomes inevitable. Hence, patients who are at risk require that their cardiac health to be monitored frequently whether they are indoors or outdoors so that emergency treatment is possible. Telemedicine is widely considered to be part of the inevitable future of the modern practice of medicine

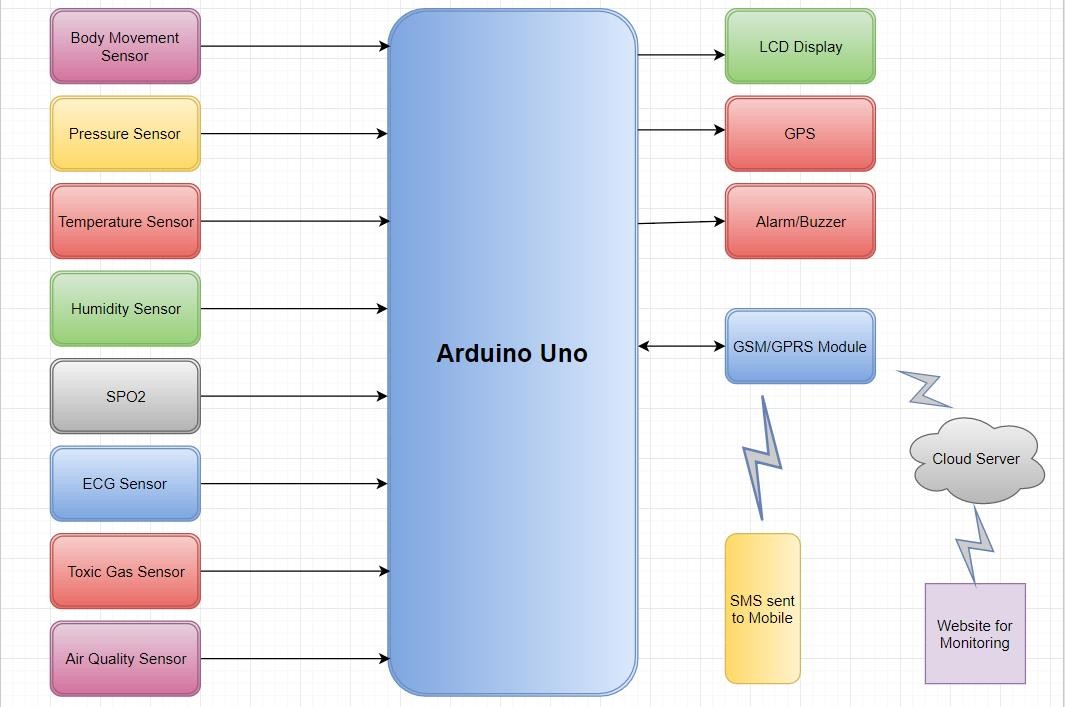
## 2.2 The real-time monitoring system for in-patient based on ThingSpeak

The system is made up of two sub-systems: patient physical states data acquisition and communication system based on ThingSpeak technology, and hospital monitoring and control center. The patient physical states data acquisition and communication system monitors the main physical parameters and movement status continuously. The information from data acquisition system is sent to hospital monitoring center by ThingSpeak wireless communication module. The monitoring center receives the information from each patient and save them to the database, and then judges the states of the patient by fuzzy reasoning. The data from the patient can be displayed as a graph or numeric on the monitor if it is necessary, and then the doctor can diagnose the patient according to the recorded continuous data. Wireless sensor network is made up of a lot of wireless sensors based on ThingSpeak technology. The ThingSpeak technology provides a resolution for transmitting sensors’ data by wireless communication. ThingSpeak technology can transmit data with a rate of 250kbps, and then it is enough for the physical parameters of the patient. The communication distance of ThingSpeak node can be over 200 meters and can be spread by add route node, and then ThingSpeak technology is suited to a short distance wireless sensors network. ThingSpeak technology owns many virtues, such as low power consumption, low cost, small size, free frequency, etc. To know the physical states of in-patient, the physical parameters need to be monitored real-time. The traditional medical test instrument is a large size and connected by wire often, and the patient is required to be quiet during the test. In most of the hospital, the medical instruments need to be read by doctor or nurse, and the physical parameters are tested and recorded one or two times each day, the real-time monitoring is expensive for most of the patients, and can be only acquirable for ICU by a nurse. For this reason, the worsening of patient can’t be found in time, and then the patient can’t be helped in time. For most of the patients can be monitored real-time in hospital, we should find a new method.Consider that the movement of the patient is limited in hospital, we adopted the ThingSpeak and wireless sensors network to acquire the physical parameters of the patient.

**CHAPTER 3**

PROPOSED SYSTEM

**3.1 BLOCK DIAGRAM OF COMPLETE SYSTEM**



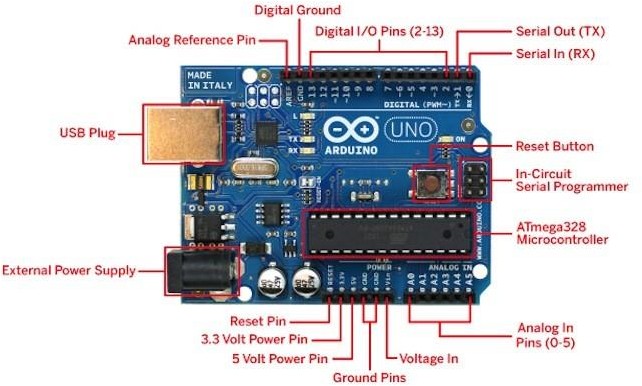
**3.2 Sensors and Modules**

Proposed system consists of following sensors and modules:-

* + 1. Arduino Micro Controller
    2. WIFI Module
    3. Temperature sensor
    4. Humidity Sensor
    5. Heart Rate Sensor
    6. Liquid Crystal Display(LCD)
    7. I2C Module
  1. **Detailed Explanation**

## 3.3.1 Arduino Micro controller

Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.



***Figure 4:*** *Arduino Board*

Starting clockwise from the top center:

* + 1. Analog Reference pin
    2. Digital Ground
    3. Digital Pins 2-13
    4. Digital Pins 0-1/Serial In/Out - TX/RX (dark green) - These pins cannot be used for digital i/o (digitalRead and digitalWrite) if you are also using serial communication (e.g. Serial.begin).
    5. Reset Button - S1
    6. In-circuit Serial Programmer
    7. Analog In Pins 0-5
    8. Power and Ground Pins
    9. External Power Supply In (9-12VDC) - X1
    10. Toggles External Power and USB Power (place jumper on two pins closest to desired supply) - SV1
    11. USB (used for uploading sketches to the board and for serial communication between the board and the computer; can be used to power the board)

**3.3.2. NodeMCU Wifi Module**

NodeMCU is an open source development board and firmware based in the widely used ESP8266 -12E WiFi module. It allows you to program  the ESP8266 WiFi module with the simple and powerful LUA programming language or Arduino IDE.

With just a few lines of code you can establish a WiFi connection and define input/output pins according to your needs exactly like arduino, turning your ESP8266 into a web server and a lot more. It is the WiFi equivalent of ethernet module. Now you have internet of things (iot) real tool.

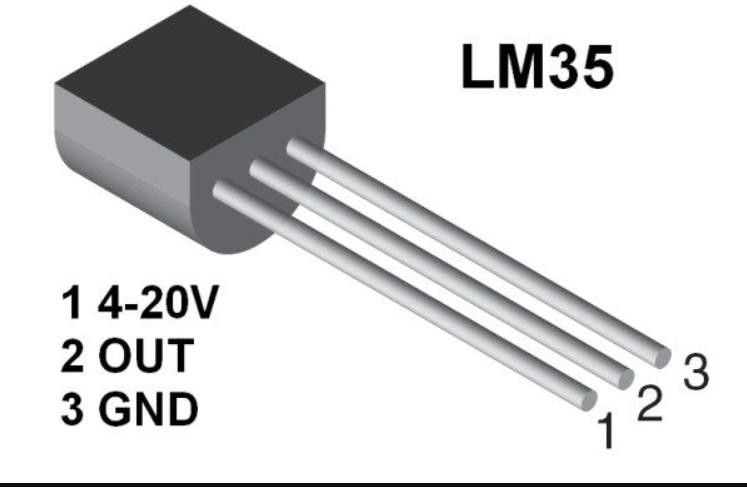
With its USB-TTL , the nodeMCU Dev board supports directly flashing from USB port. It combines features of WIFI accesspoint and station + microcontroller. These features   make the NodeMCU extremly powerful tool for Wifi networking. It can be used as access point and/or station, host a webserver or connect to internet to fetch or upload data.

**Features**

* Finally, programable WiFi module.
* Arduino-like (software defined) hardware IO.
* Can be programmed with the simple and powerful Lua programming language or Arduino IDE.
* USB-TTL included, plug & play.
* 10 GPIOs D0-D10, PWM functionality, IIC and SPI communication, 1-Wire and ADC A0 etc. all in one board.
* Wifi networking (can be used as access point and/or station, host a web server), connect to internet to fetch or upload data.
* Event-driven API for network applications.
* PCB antenna.

**3.3.3. Temperature Sensor**

Temperature sensor is a device which is designed specifically to measure the hotness or coldness of an object. LM35 is a precision IC temperature sensor with its output proportional to the temperature (in °C). With LM35, the temperature can be measured more accurately than with a thermistor. It also possesses low self-heating and does not cause more than 0.1 °C temperature rise in still air. The operating temperature range is from -55°C to 150°C.The LM35’s low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy.

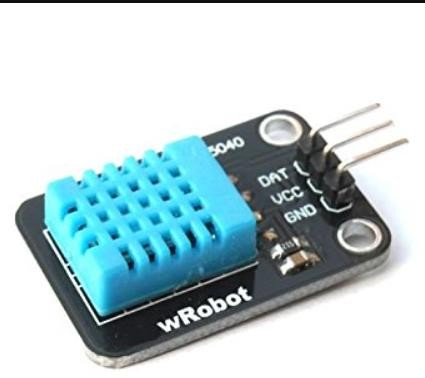


(Ref: <http://www.instructables.com/id/LM35-Temperature-Sensor/>)

***Figure 7:*** *Temperature Sensor*

**3.3.4 HUMIDITY SENSOR**

A humidity sensor (or hygrometer) senses, measures and reports the relative humidity in the air. It therefore measures both moisture and air temperature. Relative humidity is the ratio of actual moisture in the air to the highest amount of moisture that can be held at that air temperature. The warmer the air temperature is, the more moisture it can hold. Humidity / dew sensors use capacitive measurement, which relies on electrical capacitance. Electrical capacity is the ability of two nearby electrical conductors to create an electrical field between them. The sensor is composed of two metal plates and contains a non-conductive polymer film between them. This film collects moisture from the air, which causes the voltage between the two plates to change. These voltage changes are converted into digital readings showing the level of moisture in the air.



(Ref: <http://erltech.com/product/shop/dht11-temperature-and-humidity-sensor-module/>)

***Figure 10:*** *Humidity Sensor*

**3.3.5. HEART RATE SENSOR**

An alternate name of this sensor is heartbeat sensor or heart rate sensor. The working of this sensor can be done by connecting it from the fingertip or human ear to Arduino board. So that heart rate can be easily calculated.

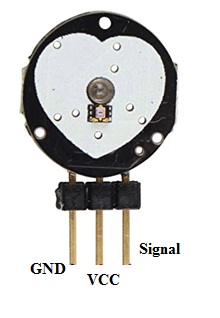
### Pulse Sensor Specifications

The main specifications of this sensor mainly include the following.

* This is a hear beat detecting and biometric pulse rate sensor
* Its diameter is 0.625
* Its thickness is 0.125
* The operating voltage is ranges +5V otherwise +3.3V
* This is a plug and play type sensor
* The current utilization is 4mA
* Includes the circuits like Amplification & Noise cancellation
* This pulse sensor is not approved by the FDA or medical. So it is used in student-level projects, not for the commercial purpose in health issues applications.

### Pin Configuration

The heartbeat sensor includes three pins which discussed below.

pulse-sensor-pin-configuration

* Pin-1 (GND): Black Color Wire – It is connected to the GND terminal of the system.
* Pin-2 (VCC): Red Color Wire – It is connected to the supply voltage ( +5V otherwise +3.3V) of the system.
* Pin-3 (Signal): Purple Color Wire – It is connected to the pulsating o/p signal

**3.3.6. LIQUID CRYSTAL DISPLAY(LCD)**

The term [LCD stands for liquid crystal display](https://www.elprocus.com/difference-alphanumeric-display-and-customized-lcd/). It is one kind of electronic display module used in an extensive range of applications like various circuits & devices like mobile phones, calculators, computers, TV sets, etc. These displays are mainly preferred for multi-segment [light-emitting diodes](https://www.elprocus.com/light-emitting-diode-led-working-application/) and seven segments. The main benefits of using this module are inexpensive; simply programmable, animations, and there are no limitations for displaying custom characters, special and even animations, etc.



### LCD 16×2 Pin Diagram

The 16×2 LCD pinout is shown below.

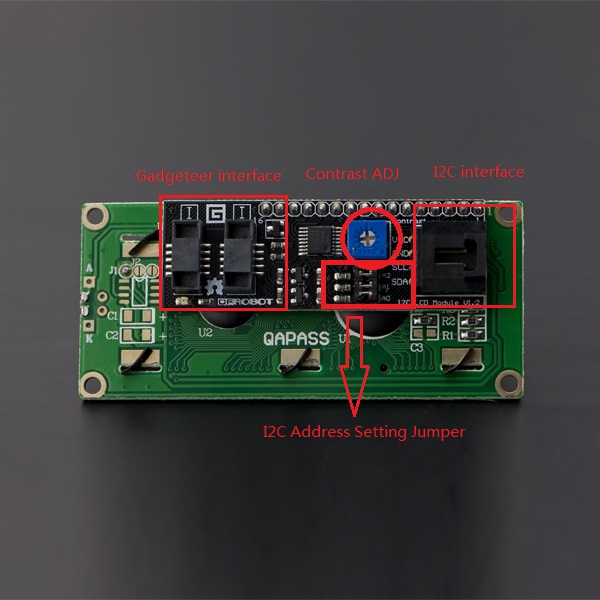
* Pin1 (Ground/Source Pin): This is a GND pin of display, used to connect the GND terminal of the microcontroller unit or power source.
* Pin2 (VCC/Source Pin): This is the voltage supply pin of the display, used to connect the supply pin of the power source.
* Pin3 (V0/VEE/Control Pin): This pin regulates the difference of the display, used to connect a changeable POT that can supply 0 to 5V.
* Pin4 (Register Select/Control Pin): This pin toggles among command or data register, used to connect a microcontroller unit pin and obtains either 0 or 1(0 = data mode, and 1 = command mode).
* Pin5 (Read/Write/Control Pin): This pin toggles the display among the read or writes operation, and it is connected to a microcontroller unit pin to get either 0 or 1 (0 = Write Operation, and 1 = Read Operation).
* Pin 6 (Enable/Control Pin): This pin should be held high to execute Read/Write process, and it is connected to the microcontroller unit & constantly held high.
* Pins 7-14 (Data Pins): These pins are used to send data to the display. These pins are connected in two-wire modes like 4-wire mode and 8-wire mode. In 4-wire mode, only four pins are connected to the microcontroller unit like 0 to 3, whereas in 8-wire mode, 8-pins are connected to microcontroller unit like 0 to 7.
* Pin15 (+ve pin of the LED): This pin is connected to +5V
* Pin 16 (-ve pin of the LED): This pin is connected to GND.

**3.3.7. I2C MODULE**

This is an 16x2 LCD display screen with I2C interface. It is able to display 16x2 characters on 2 lines, white characters on blue background.

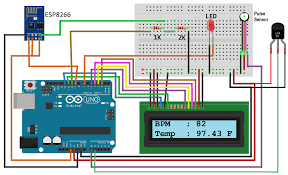
Usually, Arduino LCD display projects will run out of pin resources easily, especially with [Arduino Uno](https://www.dfrobot.com/product-610.html). And it is also very complicated with the wire soldering and connection.

This I2C 16x2 Arduino LCD Screen is using an I2C communication interface. It means it only needs 4 pins for the [LCD](https://www.dfrobot.com/category-130.html) display: VCC, GND, SDA, SCL. It will saves at least 4 digital / analog pins on [Arduino](https://www.dfrobot.com/category-35.html). All connector are standard XH2.54 (Breadboard type). You can connect with [jumper wire](https://www.dfrobot.com/product-356.html) directly.



**3.3.8. CIRCUIT DIAGRAM**

Attached below is the circuit diagram of the system designed by me in this project.

****

**CHAPTER 4**

CODES AND RESULTS

**4.1.1 HARDWARE CODE**

We had designed our software in Arduino IDE which is based on C programming language. My code is as follows:

// **importing the modules required**

#define USE\_ARDUINO\_INTERRUPTS true

#include <LiquidCrystal\_I2C.h>

#include<Wire.h>

#include <PulseSensorPlayground.h>

//**declaring the global variables and objects**

LiquidCrystal\_I2C lcd(0x3f,2,1,0,4,5,6,7,3,POSITIVE);

PulseSensorPlayground pulseSensor;

const int PulseWire=0;

int Threshold=550;

int str=0;

// **inbuilt function setup which runs once to initialize global** //**variables and objects.**

void setup() {

// put your setup code here, to run once:

//**lcd setup code.**

lcd.begin(16,2);

lcd.clear();

//**declaring serial monitor.**

Serial.begin(9600);

//**setting our GPIO pins.**

pinMode(A1,INPUT);

pulseSensor.analogInput(PulseWire);

pulseSensor.setThreshold(Threshold);

// **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(0,0);

lcd.print(" Heart Rate Monitor");

}

}

// **inbuilt function where our main code runs infinitely.**

void loop() {

int myBPM = pulseSensor.getBeatsPerMinute(); // **Calls function on our pulseSensor object that returns BPM as an "int".**

//**getting the temperature reading in our program.**

float reading=analogRead(A1);

//**processing the reading into proper format.**

reading= (reading/1024)\*5000;

float fhr = reading/10;

fhr=(fhr\*9)/5 + 32;

//**starting our code to detect heartbeats.**

if (pulseSensor.sawStartOfBeat()) {

if(str!=myBPM){

// **Constantly test to see if "a beat happened".**

str=myBPM;

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.**

Serial.println("Temperature: ");

Serial.println(fhr);

//**printing our data in the lcd.**

lcd.clear();

lcd.setCursor(0,0);

lcd.print("Temp: ");

lcd.print(fhr);

// **If test is "true", print a message "a heartbeat happened".**

lcd.setCursor(5,3);

lcd.print("BPM: "); // **Print phrase "BPM: "**

lcd.print(myBPM);

delay(10000);

}

}

delay(20);

}//**end of function and thus code.**

**4.1.2. SOFTWARE CODE**

Our software was coded in python language wherein we plotted the data received in a plot so that the expert could analyze and thereby make proper deductions about the condition of the patient and thus continue with his/her treatment.

We used the modules pandas and matplotlib to achieve our desired results.

**CODE:**

**//importing the desired modules.**

import pandas as pd

import matplotlib.pyplot as plt

//**reading the data from the excel sheet created.**

data=pd.read\_csv('C:\\Users\\PC\\Desktop\\FYP\\result.csv')

//**defining the axes to plot the data in an organized way.**

ax1=plt.subplot2grid((6,1),(0,0),rowspan=2,colspan=1,label='Humidity')

ax2=plt.subplot2grid((6,1),(2,0),rowspan=2,colspan=1,label='Heart Rate')

ax3=plt.subplot2grid((6,1),(4,0),rowspan=2,colspan=1,label='Body Temperature')

//**plotting the humidity data in the form of bar charts**

x=[i for i in range(0,100)]

ax1.bar(x,data['Humidity'],label="Humidity")

//**giving the legends to the plot.**

leg=ax1.legend(loc=1,prop={'size':11})

leg.get\_frame().set\_alpha(0.3)

//**plotting the heart rate data in the form of bar charts and giving the required legend.**

ax2.bar(x,data['BPM'],color='green',label="Heart Rate")

leg=ax2.legend(loc=1,prop={'size':11})

leg.get\_frame().set\_alpha(0.3)

//**plotting the heart rate data in the form of bar charts and giving the required legend.**

ax3.bar(x,data['Body Temperature'],color='red',label="Body Temperature")

leg=ax3.legend(loc=1,prop={'size':9})

leg.get\_frame().set\_alpha(0.3)

//**finally plotting the data.**

plt.xlabel("Days")

plt.show()

**4.1.3. FURTHER ANALYSIS CODE**

We decided to use our data to perform analysis of the code via specifically designed graphs with proper legends and conditional fillings.

We used the standard body temperature, heart rate and humidity as 99F,90 bpm, 70% respectively.

We plotted separate graphs of each parameter together to analyze the possible relationship between each parameter.

**CODE:**

**//importing required modules.**

import pandas as pd

import matplotlib.pyplot as plt

//**importing the data from the excel sheet.**

data=pd.read\_csv('C:\\Users\\PC\\Desktop\\FYP\\result.csv')

x=[i for i in range(0,100)]

//**declaring subplots to plot the graphs in a logical format.**

ax1=plt.subplot2grid((6,1),(0,0),rowspan=3,colspan=1,label='Humidity')

ax2=plt.subplot2grid((6,1),(3,0),rowspan=3,colspan=1,label='Heart Rate')

ax3=plt.subplot2grid((6,1),(3,0),rowspan=3,colspan=1,label='Body Temperature')

//**plotting the data and filling them with the required conditions.**

ax1.plot(data['Humidity'])

ax1.fill\_between(x,data['Humidity'],70,alpha=0.3,facecolor='blue')

ax1.grid(True)

ax2.plot(data['BPM'])

ax2.fill\_between(x,data['BPM'],90,alpha=0.3,facecolor='green')

ax2.grid(True)

ax3.plot(data['Body Temperature'])

ax3.fill\_between(x,data['Body Temperature'],100,alpha=0.3,facecolor='red')

ax3.grid(True)

//**giving the required legends to each graph.**

leg1=ax1.legend(loc=1,prop={'size':11})

leg1.get\_frame().set\_alpha(0.2)

leg2=ax2.legend(loc=1,prop={'size':11})

leg2.get\_frame().set\_alpha(0.2)

leg3=ax3.legend(loc=1,prop={'size':9})

leg3.get\_frame().set\_alpha(0.1)

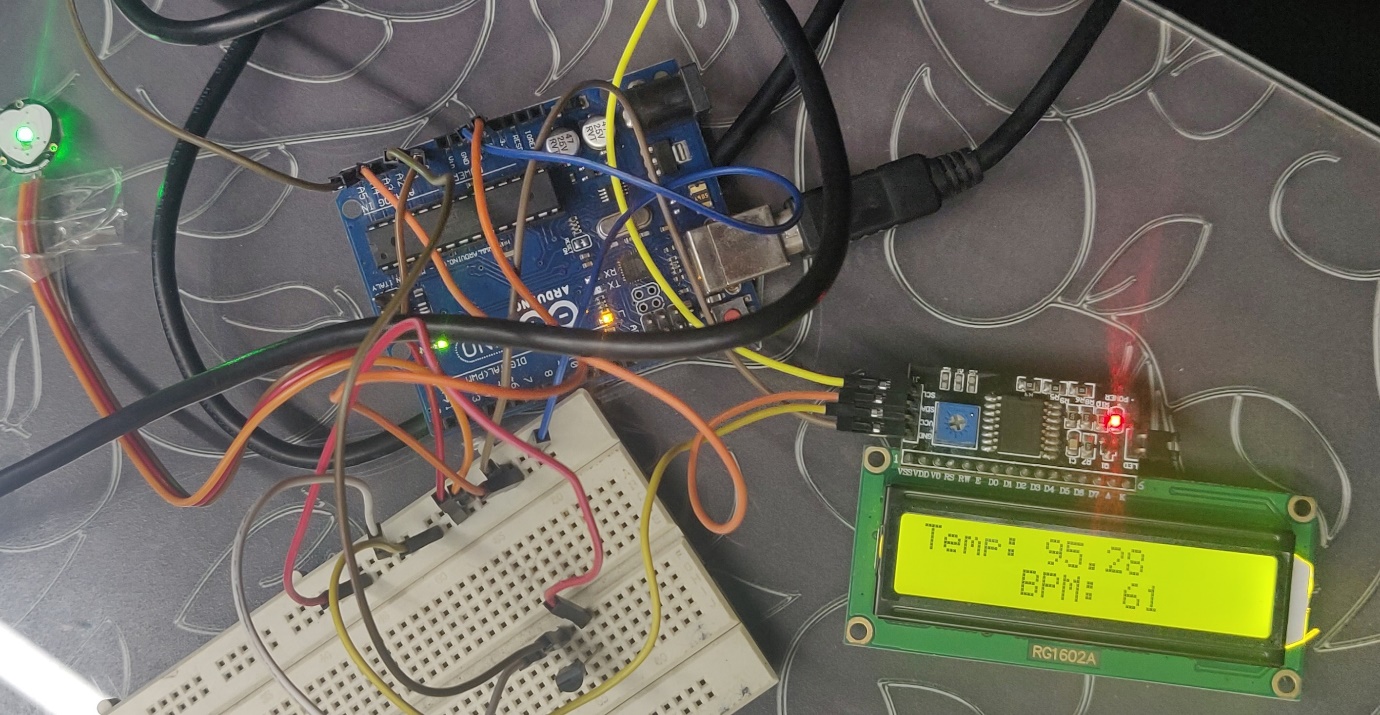
//**plotting the graphs.**

plt.xlabel("Days")

plt.show()

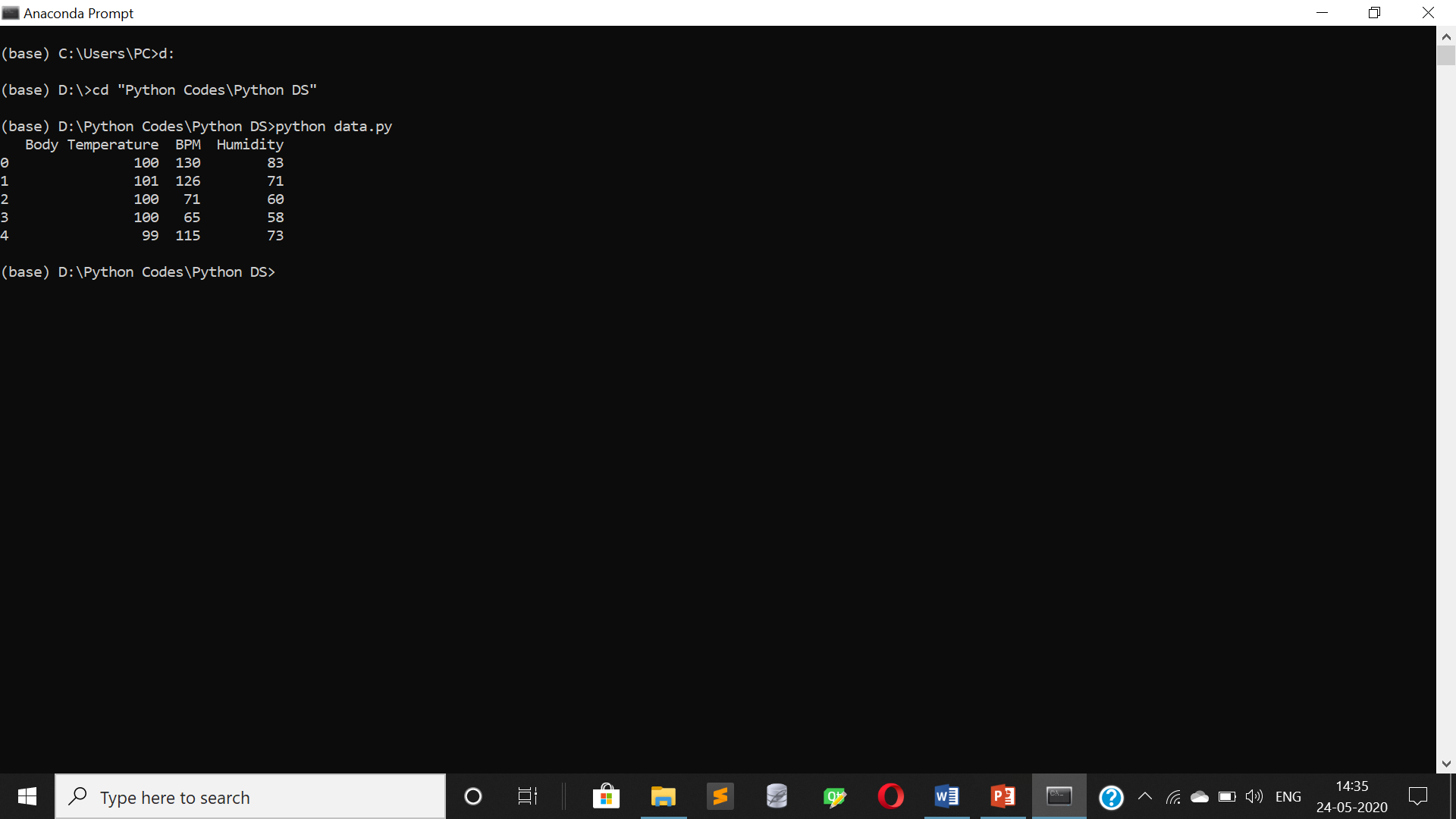
**4.2.1. HARDWARE RESULT SNAPSHOT**

The following is a snapshot of our project in the working condition. In the figure is visible the temperature, humidity, bpm sensors and the lcd module attached with the I2C module.



**4.2.2.SNAPSHOT OF THE FIRST FEW ELEMENTS OF OUR DATASET.**

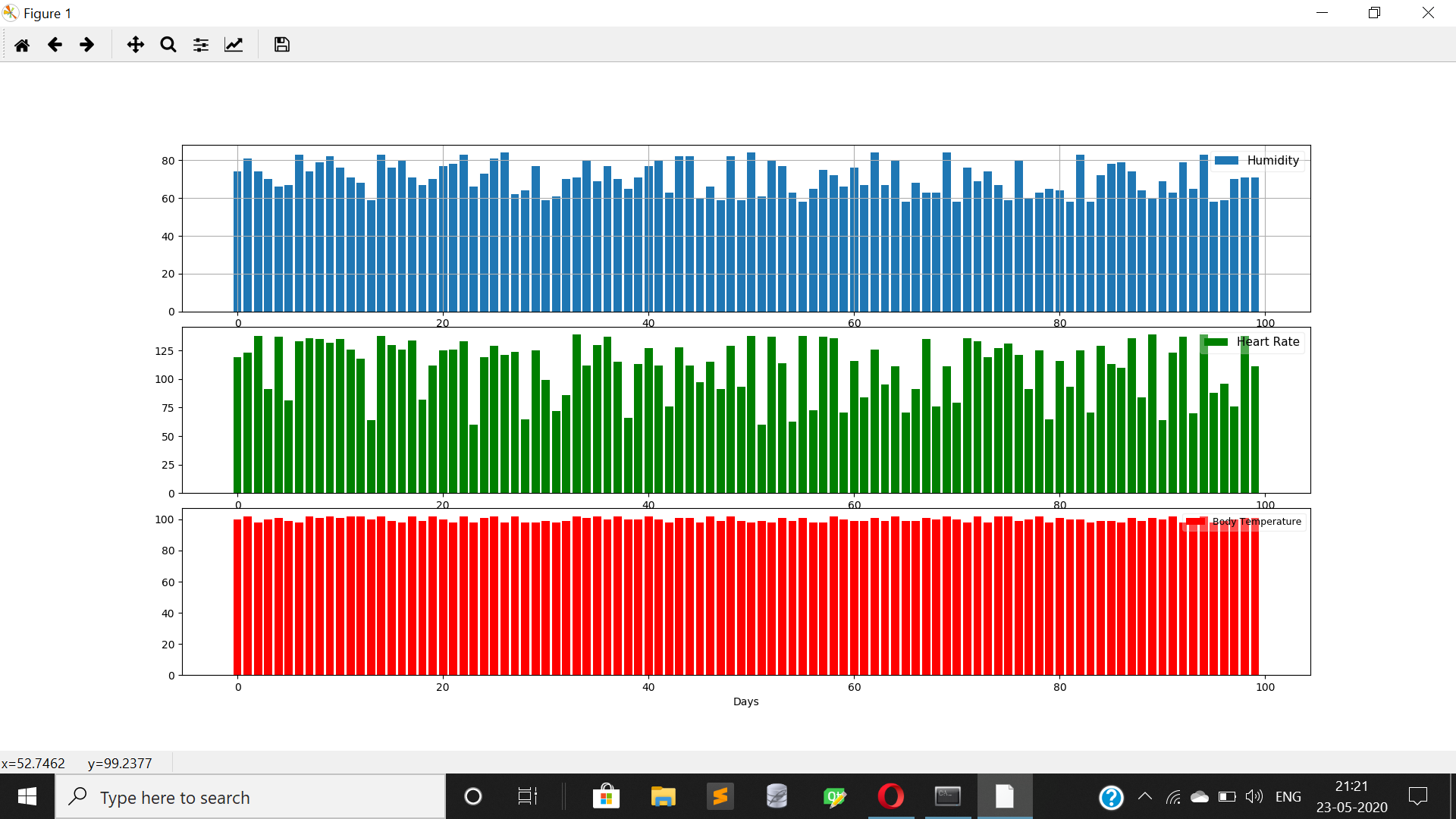
Attached herewith is the screenshot of the python command prompt displaying the first five rows of our dataset.



In the figure is clearly visible the three columns of our dataset.

**4.2.3.SOFTWARE RESULT.**

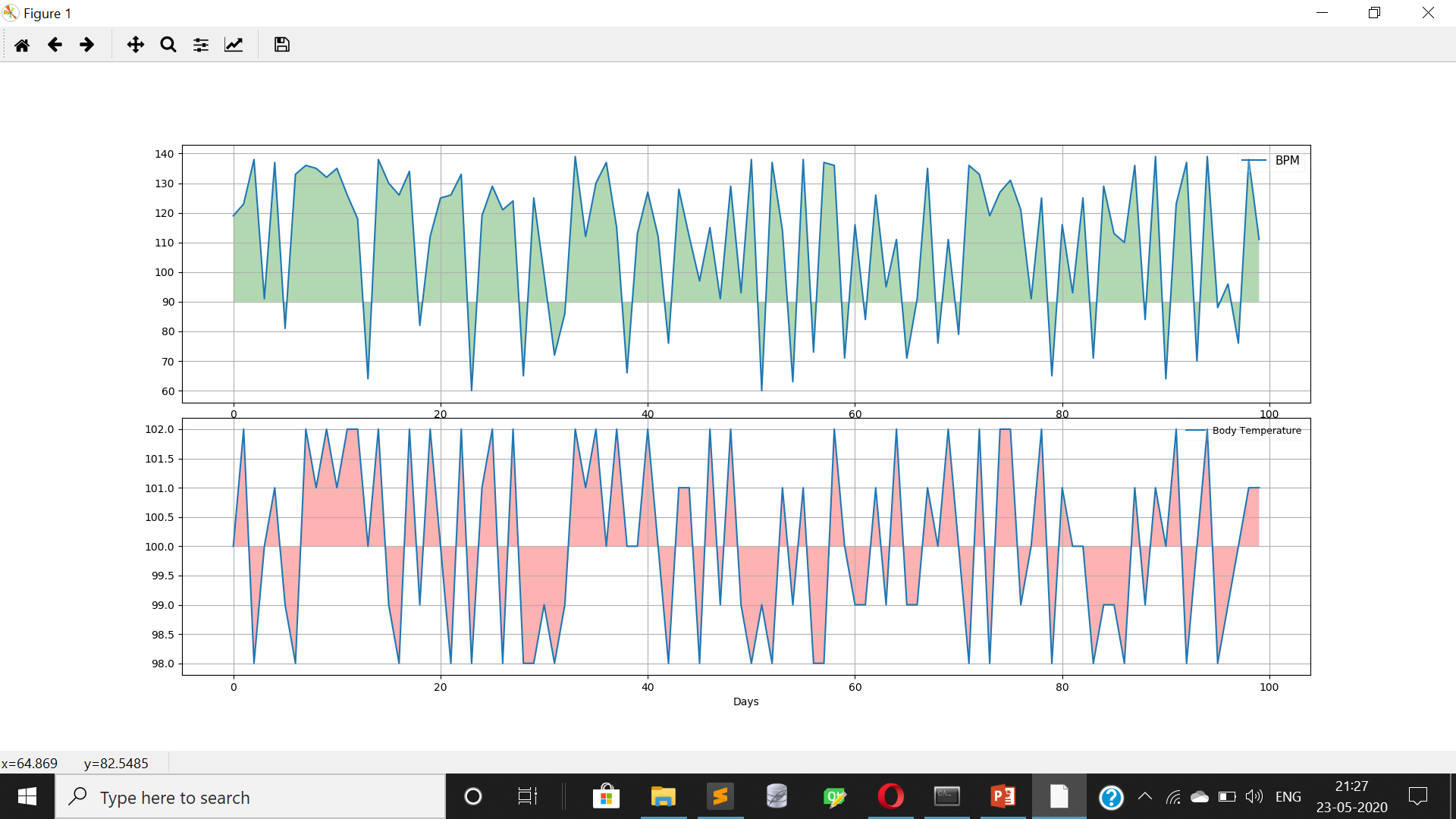
In the figure given below is shown the graphs of the dataset collected from our device.



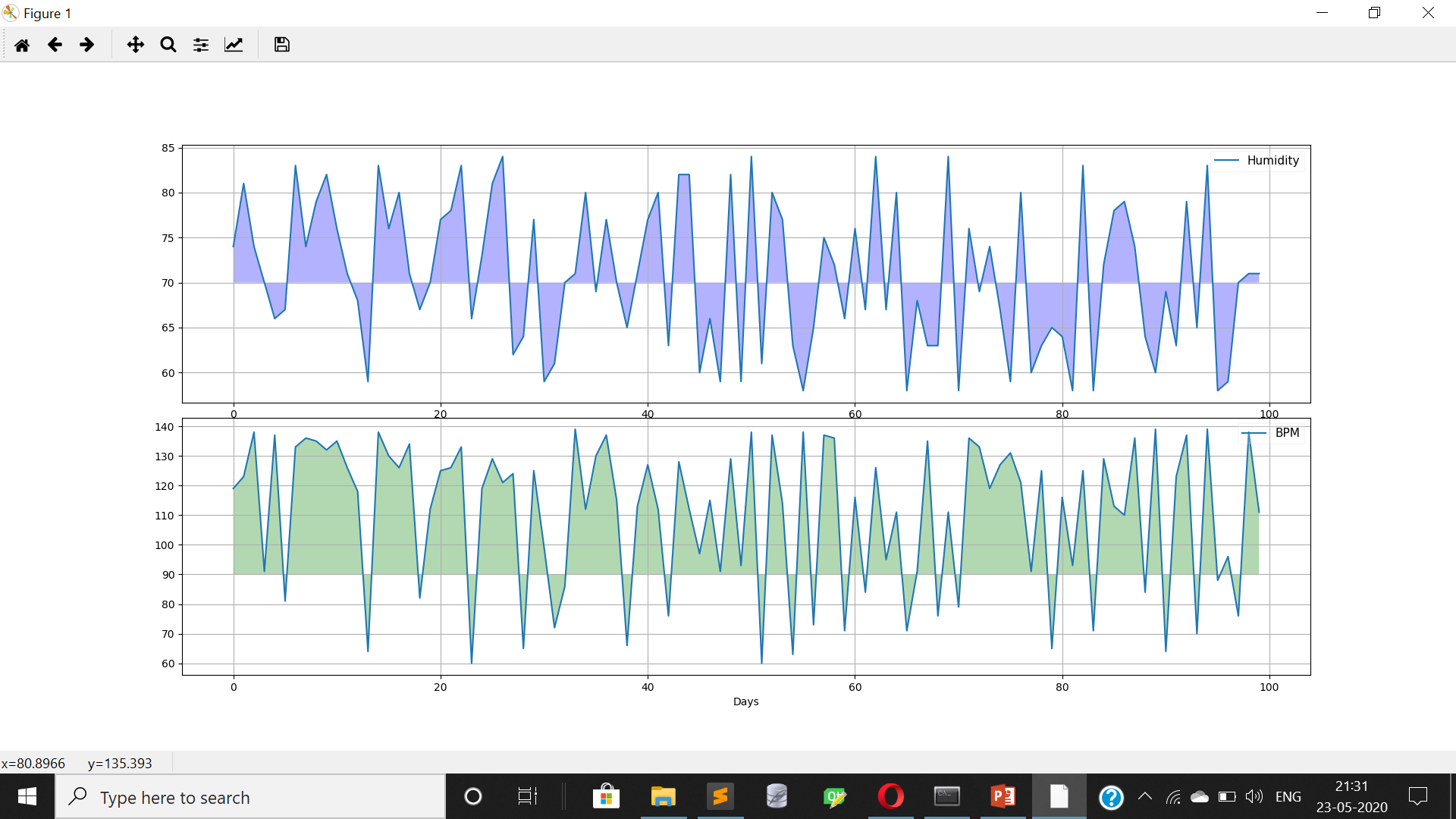
We can see three bar charts. The first graph with blue bars is of the humidity. The second bar graph with green bars is the one representing the heart rate. And the final graph with red bars represents the Temperature data.

**4.2.4.FURTHER ANALYSIS RESULT.**

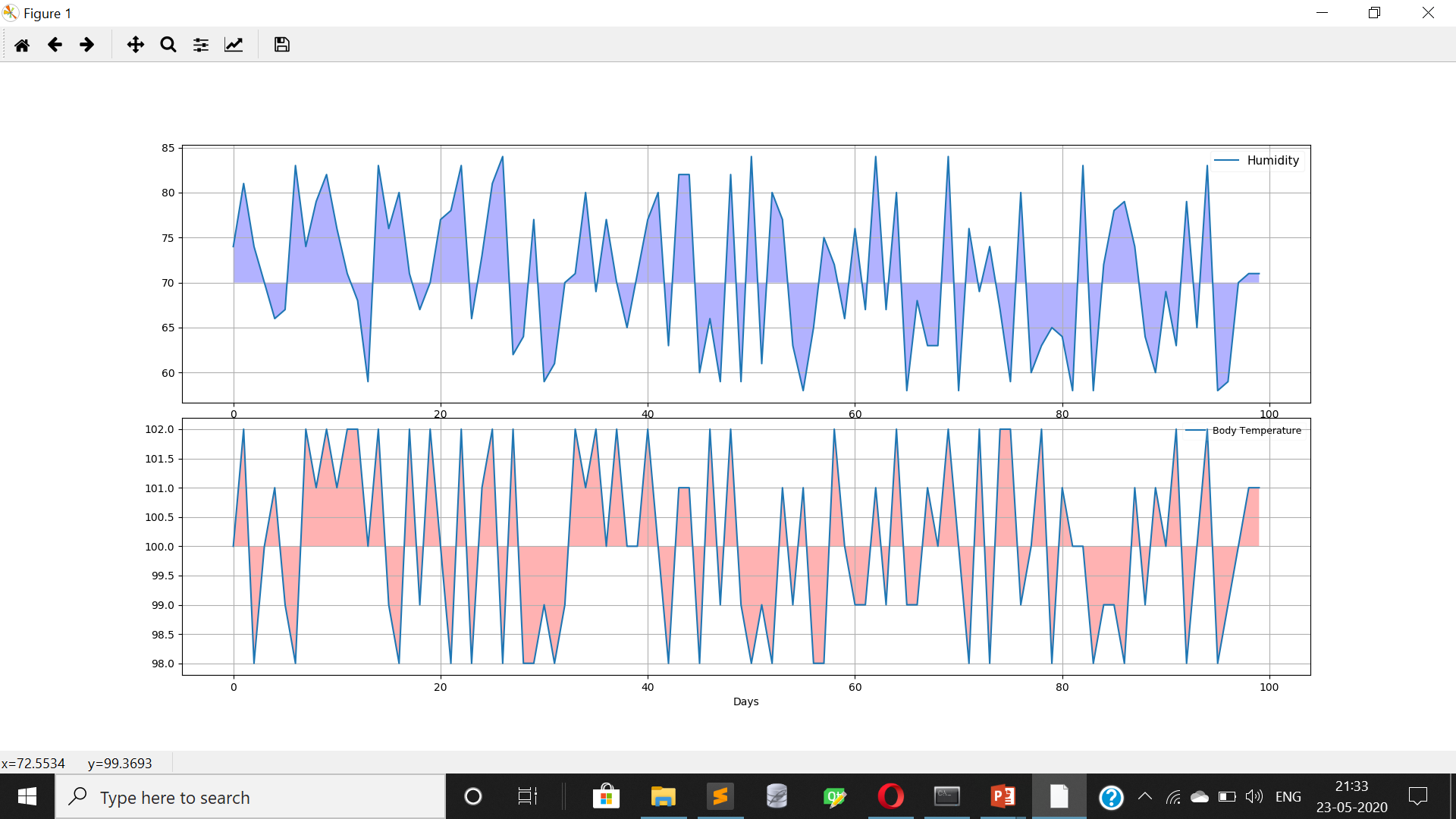
The following graph is the plot between the Body Temperature and BPM. The plots are filled with respect to the normal parameters of a person.



The following graph is the plot between the Humidity and BPM. The plots are filled with respect to the normal parameters of a person.



The following graph is the plot between the Humidity and Body Temperature. The plots are filled with respect to the normal parameters of a person.



**4.2.5.CONCLUSIONS**

* From our graphical analysis we came to the conclusion that:

1. The Body Temperature of a person is directly proportional to his/her Heart beats per minute. Moreover from the data it was observed that if the BPM of a person crosses 100 there are high chances of that person being down with fever.
2. The Heat Beats per minute(BPM) of a person is also directly proportional to the humidity of the environment. Thus the health professionals suggest taking extra care of health during monsoons wherein the humidity is at its peak. It was observed that the humidity going over 80% can lead to very high BPM.
3. Even though one feels warm amidst high humidities there was no definite relation between the body temperature and humidity observed from our analysis.

**4.2.6.FUTURE SCOPE**

Every project or system has a scope of improvement and betterment. A few areas on which this system of ours can be developed further are:

1.Advanced GSM and GPRS technologies can be embedded so as to ensure real time and place evaluation of the patient.

2.Various other data like blood pressure, retinal size , age and weight of the patient could be collected for a better database and a important step towards proactiveness of the system.

3.The system could be given a proactive sense by analysing the data using Machine Learning and Artificial Intelligence and thereby predict the chances of the patient falling ill.

**CHAPTER 5**

REFERENCES

**REFERENCES**

1. “Arduino Architecture”

<https://www.engineersgarage.com/what-is-gsm-gprs-module>[Oct. 1, 2017]

1. “Systems design” [https://en.wikipedia.org/wiki/Systems\_design](https://en.wikipedia.org/wiki/Systems_design%20) [Oct. 15, 2017]
2. “UML - Standard Diagrams” <https://www.tutorialspoint.com/uml/uml_standard_diagrams.htm>[Oct. 18, 2017]
3. “The Internet of Things in healthcare: an overview” <https://scholar.google.com/citations?user=Y4opLB8AAAAJ&hl=en>[Sept. 7, 2017]

1. “Envisioning inclusive futures: technology-based assistive sensory and action substitution” [https://www.infona.pl/resource/bwmeta1.element.elsevier-3d45bfdd-fe55-359f-84e4-](https://www.infona.pl/resource/bwmeta1.element.elsevier-3d45bfdd-fe55-359f-84e4-674a21cae024)

[674a21cae024](https://www.infona.pl/resource/bwmeta1.element.elsevier-3d45bfdd-fe55-359f-84e4-674a21cae024) [Sept 4, 2017]

1. “A multiple communication standards compatible IoT system for medical usage” <http://ieeexplore.ieee.org/document/6577775/?reload=true>[Sept 5, 2017]
2. “Ubiquitous data accessing method in IoT-based information system for emergency medical services”

[https://www.deepdyve.com/lp/institute-of-electrical-and-electronics-engineers/ubiquitous-](https://www.deepdyve.com/lp/institute-of-electrical-and-electronics-engineers/ubiquitous-data-accessing-method-in-iot-based-information-system-for-YCZzyY5W9g) [data-accessing-method-in-iot-based-information-system-for-YCZzyY5W9g](https://www.deepdyve.com/lp/institute-of-electrical-and-electronics-engineers/ubiquitous-data-accessing-method-in-iot-based-information-system-for-YCZzyY5W9g) [Sept 6, 2017]

1. “Implementation of a medical support system considering P2P and IoT technologies” <https://www.computer.org/csdl/proceedings/cisis/2014/4325/00/4325a101-abs.html> [Sept 7,

2017]

1. “Acquisition and management of biomedical data using Internet of Things concepts” <http://ieeexplore.ieee.org/document/7050625/>[Sept 10, 2017]
2. “Real time internet application with distributed flow environment for medical IoT” <https://csdl.computer.org/csdl/proceedings/icgciot/2015/7910/00/07380578-abs.html>[Sept 11,

2017]

1. “Secure end-to-end communication for constrained devices in IoT-enabled ambient assisted living systems”

<https://www.computer.org/csdl/proceedings/wf-iot/2015/0366/00/07389141-abs.html> [Sept 11, 2017]