A Report on Microcontroller Project

CSE316

Health Monitoring System

Submitted by:
Ayesha Binte Mostofa

1805062

Zarin Tasnim

1805071

Nusrat Nur Labiba

1805081

Sep 2, 2022

Contents

1 Introduction

This report briefly illustrates the infrastructure of our microcontroller term project of the CSE-316 course.

The main aim of the project is to monitor heart rate, body temperature and blood oxygen saturation level of a patient body. It also monitors room humidity and room temperature of the patient room. This system designed by using Arduino Uno. Pulse oximeter is being used here to measure temperature, blood oxygen level and pulse rate of body. Temperature and humidity sensor are being used to measure room temperature and room humidity. Corona Patient can be benefitted from this technology the most. A message will be sent to the doctor in case of any critical situation by passing signal to GSM module. A buzzer will be activated so that doctor can pay more attention to the situation.

This report will reflect on all the components that we used in our project. We will be including all the difficulties that we faced during executing our project. We are going to try to explain every details in this document.

2 Motivation

In 2020, thousands of people were affected by corona. Still in this year, at any time, hospitals can and are getting overcrowded with patients. That is why a cheap, automated monitoring system for patients is an urgent necessity. Patients with stable condition need not be at the hospital if they have this device. And doctors needn't to be here to the patient to check so closely. So, the 6 feet distance protocol can be maintained.

The pulse-oximeter is a tool which is very simple to use and DHT11 (room humidity and room temperature sensor) takes data read from the room continuously if connected with power. If the saturation of oxygen gets down due to corona, it can become deadly in a moment. Patient's heart rate and room condition is a health affecting factor too. So, a message is sent to doctor in critical condition.

In order to keep oneself updated on his/her body temperature, oxygen saturation level and heart rate (bpm) in areas with

overcrowded hospitals, oximeter and heart rate monitoring machines can save thousands of lives. Room condition affects a patient health condition. So, we have considered that condition too. Our vision of such a device includes the following features:

- Easily portable
- Constantly measures blood oxygen level, temperature and heart rate of a patient body and monitors room humidity and room temperature.
- Sends alert to doctor when oxygen level is too low or heart rate is low or high or room condition is not good using GSM and Buzzer.

3 Components

We bought all the components from TechShopBD and electrical device are from various shops. The list of components is given below:

Component name and model	Quantity	Price Per Unit (Tk)
MAX30102	1	870
DHT11	1	156
16x2 LCD Display	1	186
I2C Adapter	1	100
Arduino Uno	1	1370
GSM Module (SIM-900A)	1	1050
Buzzer	1	19
Breadboard	2	70
Mini Breadboard	1	40
Jumper Wires	2 packs	20
Power Bank	1	1500
Resistors	30	2
12V-2A Adapter	1	180

Table : Component List

4 Block Diagram

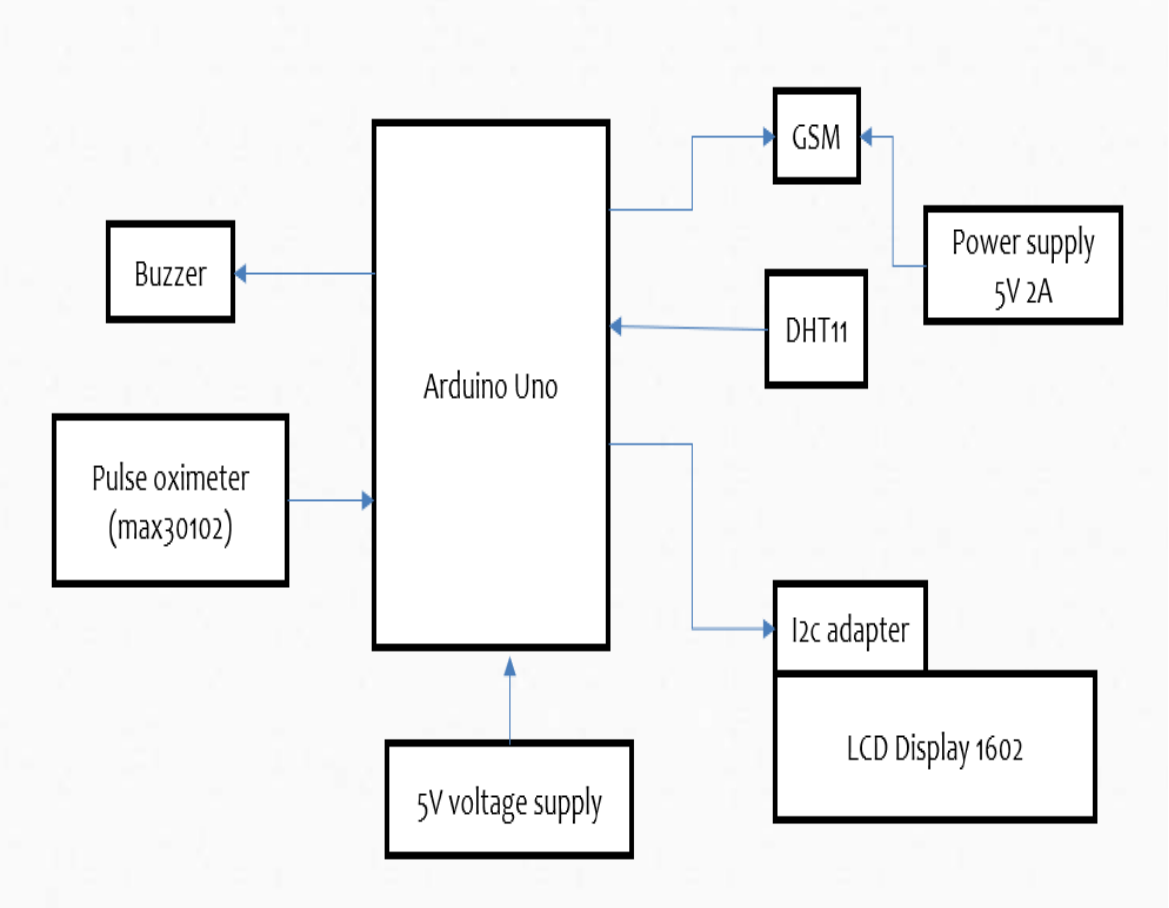

Fig: Block Diagram

5 Description

In this section, we are going to describe our entire circuit.

5.1 Complete Circuit

In our module we connected a Max30102 Pulse Oximeter and Heart Rate Sensor which measures any person's blood oxygen level and it sends the data to Arduino Uno. Then Arduino sends the data to LCD display and shows the body temperature, heart rate and blood oxygen level (SPo2) value on the display. If the data is out of range (SP02 <90 or HR<60 or HR>125), then a signal is sent to the GSM module from the Arduino and a buzzer will be activated. Again, if the room temperature > 36 degree centigrade and room humidity < 40%, alert (smg & alarm) will be sent to doctor using GSM module and Buzzer.
We tried to minimize our project with the help of a mini breadboard and fewer wire connections.

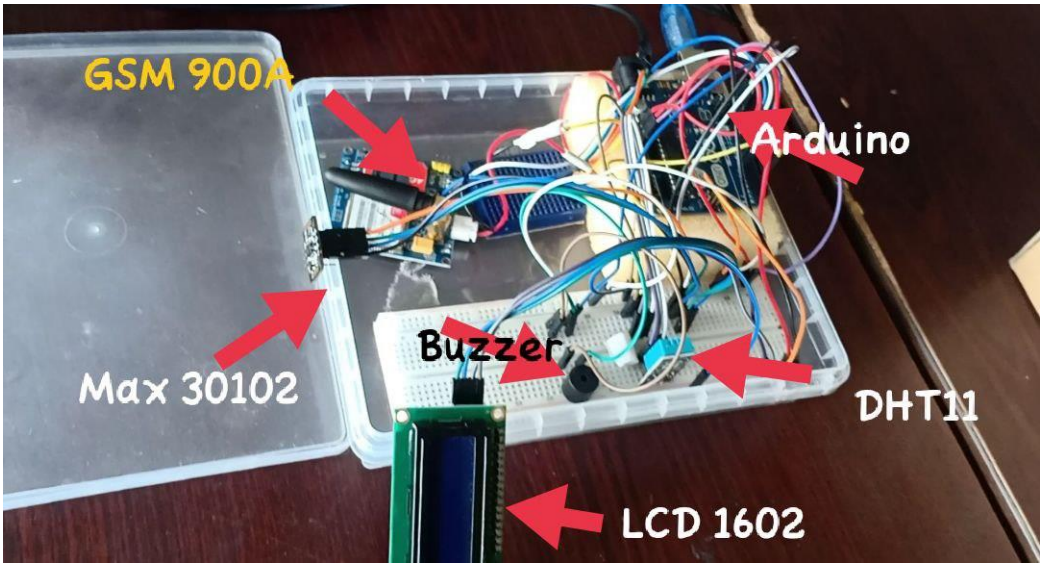

Fig: Complete Circuit

6 Circuit Diagram

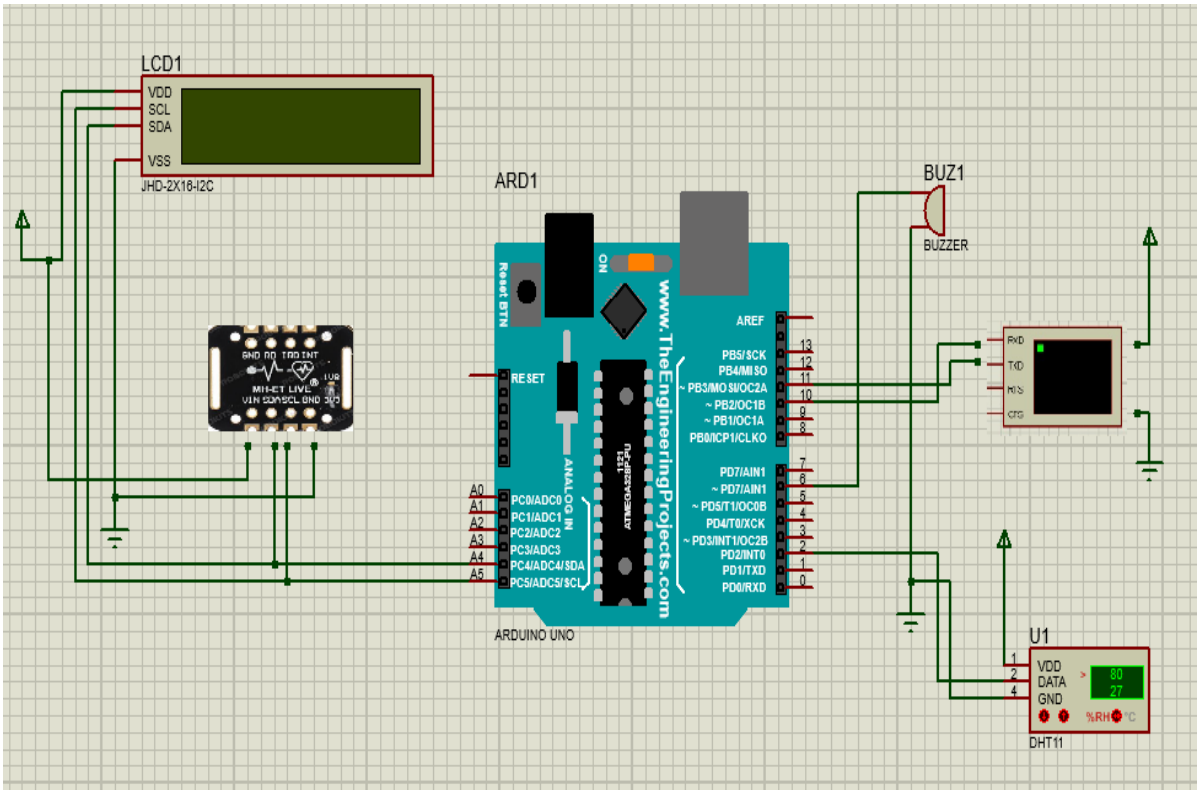

Fig: Circuit Diagram

7 Connection

We will describe each major component that we used in our project here and also the difficulties that we faced while connecting them.

7.1 Arduino Uno

- **Configuration:** The Arduino was powered through USB cable from computer. With the help of serial monitor, the modules connected with it could be configured easily.

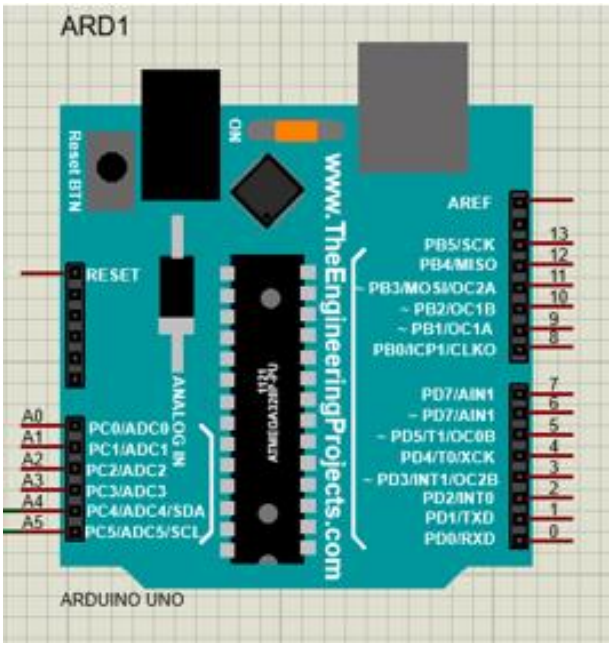

Figure : Arduino Uno

• **Difficulties & Solutions**:: No difficulties were faced for the Arduino.

7.2 Pulse Oximeter and Heart Rate Sensor

- **Configuration**: This sensor has 8 pins. Only 4 pins were used. They are :
  - VCC: Module power supply 3.3 to 5 volt
  - GND: Ground
  - SCL: I2C clock bus
  - SDA: I2C data bus

  We connected the VCC with 3.3V pin of Arduino and Ground with Arduino Gnd. Then we connected the Arduino I2C SDA pin with SDA pin of the sensor and the Arduino I2C SDA pin with SCL pin of the sensor. Also, we used “MAX3010Xs Sensor Library” library for this sensor to work.

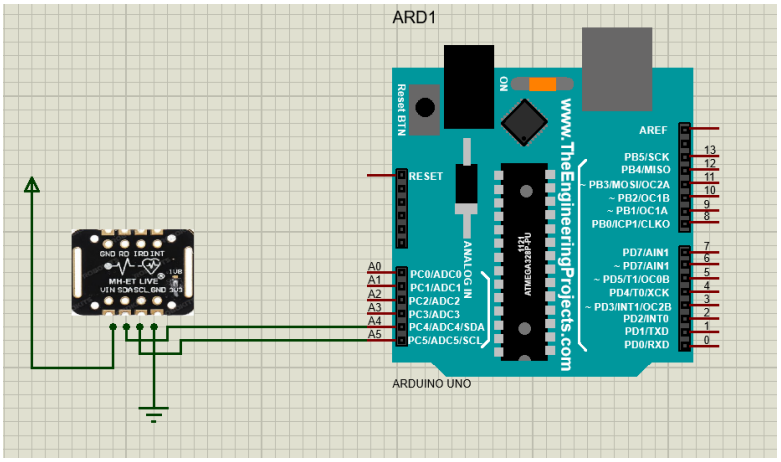

Figure : Max-30102 Connection

- **Difficulties & Solutions**: There were many difficulties we faced connecting this device. They are listed below:
  - Firstly, the sensor was not stable. If we connect the sensor with 3.3 volt instead of 5 volt then it becomes more stable. Soldering the sensor to the board can give it better stability.
  - Secondly, the sensor was not giving us accurate data. If we place the finger strongly on the light then the blood vessels can absorb the light properly. Then the sensor will give the accurate value.

7.3 DHT11(Sensor)

- **Configuration** : This DHT11 was connected with Arduino Uno. This sensor has 3 pins:
  - VDD: Module power supply 5 volt
  - GND: Ground
  - Data: Data pin

We connected the VDD and Ground with the 5V and GND pin of Arduino. Then we connected the Arduino D2 pin with Data pin of the module. We used “AdaFroot unified Sensor” and “DHT sensor library” for this to work.

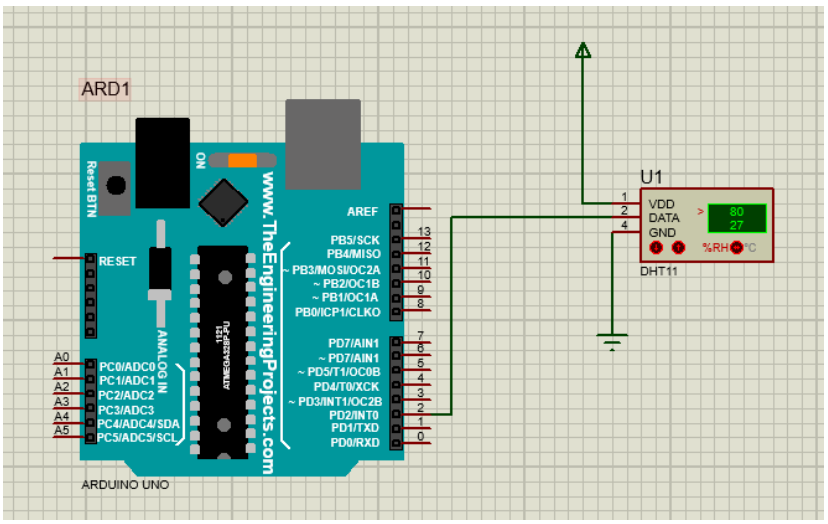

Figure : DHT11 Connection

- **Difficulties & Solutions** : No difficulties were faced for the sensor.

7.4 LCD Display(16 × 2)

- **Configuration** : We used an LCD display interfaced with an I2C module. Four pins used by us were :
  - VCC: Module power supply 5 volt
  - GND: Ground
  - SCL: I2C clock bus
  - SDA: I2C data bus

  We connected the VCC with Arduino 5V and Ground with Arduino GND(In the figure, we had to show it differently because Arduino power and ground pins are removed in proteus).Then we connected the Arduino I2C SDA pin with SDA pin of the LCD and the Arduino I2C SCL pin with SCL pin of the LCD. Also, we used "LiquidCrystal-I2C" library for the LCD to work.

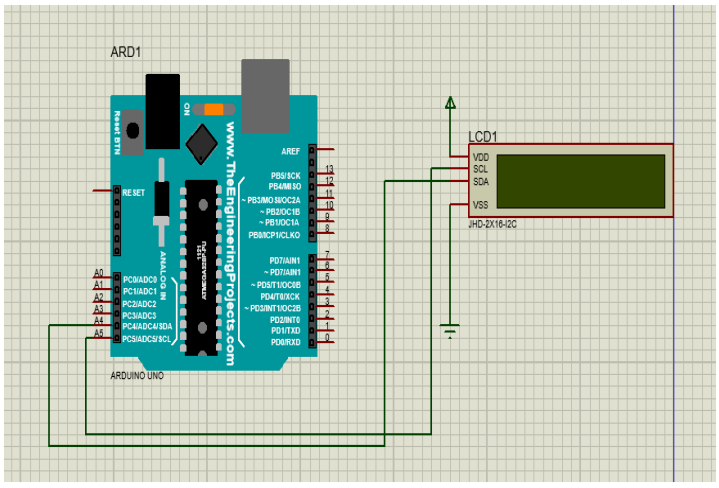

Figure : Connection of LCD(16 × 2) with Arduino using I2C Adapter

- **Difficulties & Solutions**: There were some difficulties we faced connecting this device. They are listed below:
  - Firstly, LCD does not work if it does not get 5 volt all the time. So, 5V is to be ensured.
  - Secondly, the light was too dim to see. So, we had to rotate the potentiometer on the back of the LCD to get a proper view.

•

7.5 Buzzer

- **Configuration**: We used a buzzer. 2 pins used by us were :
  - VCC
  - GND: Ground

  We connected the VCC in Arduino D6 PWM pin and Ground to common ground point in breadboard. We have set tone value to make sound by buzzer. We have tested using a resistor in the middle but it works fine without a resistor.

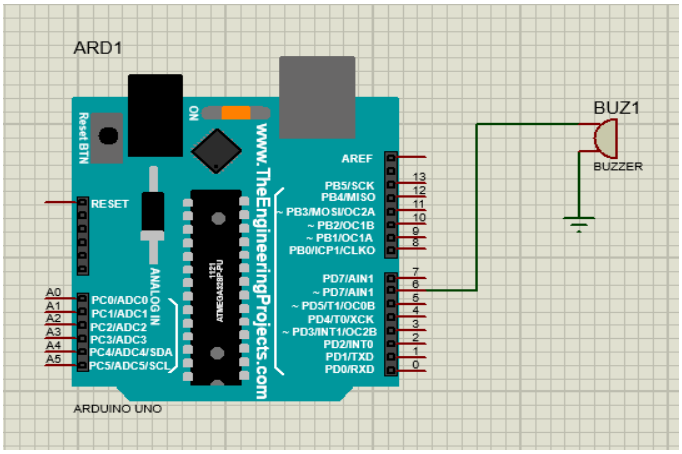

Figure : Connection of Buzzer with Arduino

• **Difficulties & Solutions**: Buzzer tone setting was difficult. We have set a lower volume by trying different tones.

•

7.6 GSM Module(SIM 900A)

Configuration : This module's 4 pins were used:

- VCC: Module power supply 5 volt
- GND: Ground
- TX: Transmits data
- RX: Receives data

We connected the VCC with power bank (5V - 2A) and Ground with the common ground of Arduino. Then we connected Arduino D10 with RXD pin of the module and the Arduino D11 with TXD pin of the module. The power LED must remain stable and the network LED should blink with an interval of 3 seconds, which indicates network stability.

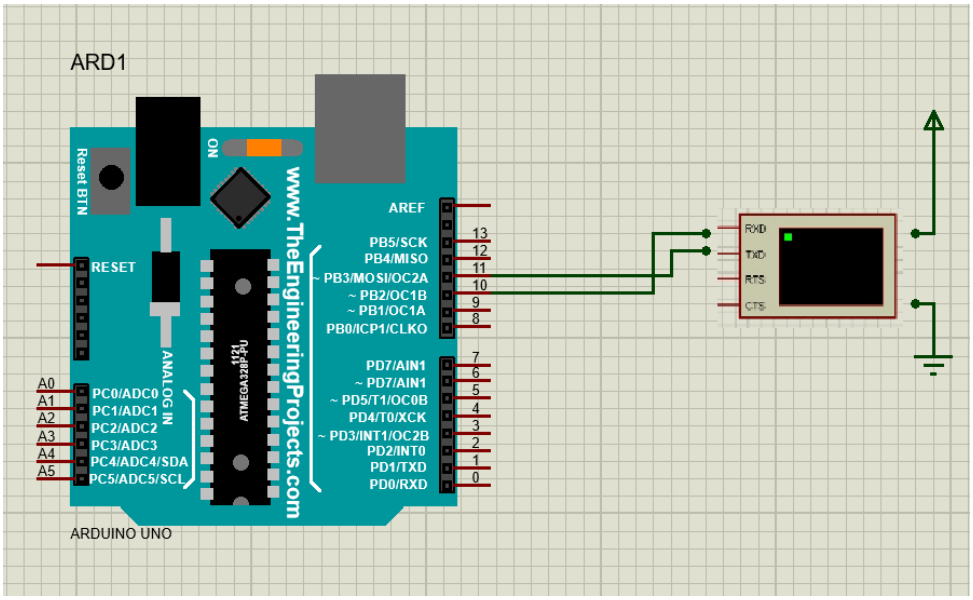

Figure : Connection of GSM with Arduino

- **Difficulties & Solutions**: There are many difficulties we faced connecting this device. They are listed below:
  - The power LED was not stable because it needs 2A current along with 5 volt voltage. So, we connected it with a power bank through a power bank. The power bank gives 2A and 5V (Arduino can give 2A sometimes but not always). We bought a 12V – 2A adapter but it didn't work so we had to power up by a power bank.
  - The sim was not getting any signal. So we tried with 3 sims. But it still was not working. After pressing the sim slot for a minute, the network was stable.
  - We couldn't find network in room, we had to go outside to connect sim with network.
  - We had to make all the grounds common to stabilize the ground of GSM module.
  - With the command AT+CMGF=1, we set the message mode on. Also, we used "SoftwareSerial Library" here. We tried without software serial but it was not working.
  - Most importantly, we had to check if the sim had enough balance to send messages.

8 Overall Difficulties

We had to face major difficulties while buying components for our project because many components were not working soundly. So, we had to go to the shop many times. And, we needed to solder after buying the components. Finding a soldering shop was a little bit difficult in academic week.

9 Conclusion

We were able to successfully implement all the functionalities we planned for our project. We wanted to add body moisture sensor but we found no sensor to do that. One was soil water sensor and another sensor was rain detector. It was not wise to use any of those because it won't give human body moisture. Instead of this, we added room humidity and room temperature sensor. During the process we learnt that working with hardware components are actually much different than software simulations. There are many variables and unexpected events that could occur. However, after successful completion it is very satisfactory to create a tangible product.

We created a product with the hope that it may lessen the load off of the hospitals in the peak times of the pandemic by remotely monitoring the patients. The patient room condition can be known and it's possible to take measure using the data. Doctor can maintain 6 feet distance a little bit more due to this device. It is quite affordable for using at home.

10 References

We received immense help from our supervisor and other course teachers. Moreover, we checked some websites and here are some resources that were used in our project :

- https://electropeak.com/learn/interfacing-max30102-pulse-oximeter-heart-rate-module-with-arduino/
- https://lastminuteengineers.com/i2c-lcd-arduino-tutorial/
- https://www.instructables.com/GSM-SIM900A-With-Arduino/
- https://create.arduino.cc/projecthub/SURYATEJA/use-a-buzzer-module-piezo-speaker-using-arduino-uno-89df45
- https://create.arduino.cc/projecthub/pibots555/how-to-connect-dht11-sensor-with-arduino-uno-f4d239