SMART HEALTH MONITORING SYSTEM

by

Keerthivasan J (19BLC1045)

Motish Raj P (19BLC1088)

Sai Saran S (19BLC1183)

A project report submitted to

Dr. BERLIN HENCY

SCHOOL OF ELECTRONICS ENGINEERING

in partial fulfillment of the requirements for the course of

ECE3502 – IoT Domain Analyst

in

B.Tech. ELECTRONICS AND COMPUTER ENGINEERING

VIT

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Vandalur – Kelambakkam Road

Chennai – 600127

JUNE 2022

BONAFIDE CERTIFICATE

This is to certify that the Project work titled “Smart Health Monitoring System" is being submitted by **Keerthivasan J (19BLC1045), Motish Raj P (19BLC1088) and Sai Saran S (19BLC1183)** for the course **ECE 3502-IoT Domain Analyst**, is a record of bonafide work done under my guidance. The contents of this project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University.

Dr. Berlin Hency
Associate Professor
School of Electronics Engineering (SENSE),
VIT University, Chennai
Chennai – 600 127.

ABSTRACT

In the current scenario, the workload of the healthcare industry is tremendously increasing especially, after the advent of COVID-19. The ratio of the number of patients to the number of doctors is on the rise. A doctor cannot constantly monitor all the patients at a time. The other healthcare workers like the nurses or the medical assistants cannot always monitor them properly. The patient responders like relatives and friends also cannot look after them to the maximum extent. So, there is a need for automating the process of monitoring patient health which not only reduces the workload of medical staff but also enables them to view the patient health parameters from anywhere and gives alert to them whenever required. The paper introduces a Smart health monitoring system based on IoT. The proposed system measures the body health parameters and displays it in a webpage and in the Blynk app through which the doctors/patient responders can check the status of the patient. It also gives alerts to the doctors/patient responders when the health parameters cross a certain threshold and sends the patient's data to the cloud platform and performs data analytics.

ACKNOWLEDGEMENT

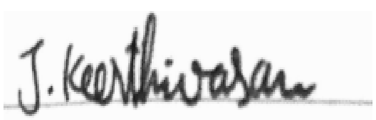
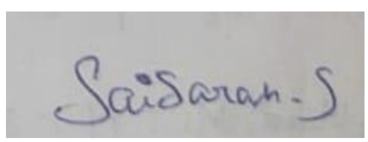
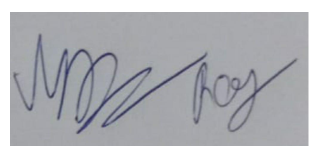
We wish to express our sincere thanks and deep sense of gratitude to our project guide, **Dr.Berlin Hency**, Associate Professor, School of Electronics Engineering, for his consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work.

We are extremely grateful to **Dr. Sivasubramanian**. A, Dean of the School of Electronics Engineering, VIT Chennai, for extending the facilities of the School towards our project and for his support.

We express our thanks to our Head of the Department **Dr. Vetrivelan.P** for his support throughout the course of this project.

We also take this opportunity to thank all the faculty of the School for their support and their wisdom imparted to us throughout the course.

We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

KEERTHIVASAN J **SAI SARAN S** **MOTISH RAJ P**

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

In India, the population is growing massively. The huge population in India is not supported with adequate health care facilities and infrastructure. In our country, the ratio of number of doctors to the total population is very less. The world is witnessing a constant increase in the patient count in hospitals due to many new diseases and also due to the modern lifestyle and food habits followed by people. As a result, the government hospitals are always fully accommodated. People/Patients who are financially strong opt for private hospitals over government hospitals due to better facilities. Poor people can't afford private hospitals because of its expenses and the government hospitals are fully occupied. So, people with a strong economical background, get better healthcare than the poor people. Doctors and other health care workers have a huge workload in order to meet the health demands of the massive population. The proposed Smart health monitoring system solves the problems stated above to great extent. It enables the doctor to monitor multiple patients in a short span of time from anywhere in the world and also gives alerts to them whenever required. The patients who are out of the serious condition and just need proper monitoring by the doctors can be monitored from their homes, thus creating more accommodation for the patients who need intensive care and treatment.

1.1 OBJECTIVE AND GOALS

The objective of our project is To design an IoT based smart health monitoring system which

- Measures the body health parameters such as blood oxygen level(SpO2), body temperature and heart rate.
- Displays the measured parameters in a webpage and in the Blynk app from which the doctors/patient responders can check the status of the patient.
- Gives alert to the doctors/patient responders when the health parameters cross certain threshold.
- Sends the patients data to the cloud platform which shows visual graph representation.
- Performs data analytics which helps the doctor to a great extent in understanding the patient's health conditions.
- Additionally, it also senses the temperature and humidity of the patient's surroundings.

1.2 APPLICATIONS

The proposed system has many features that makes it suitable for a post-operation/post-surgery monitoring of the patient's health which greatly reduces the workload of a doctor and makes them focus more on patients who are in a critical situation. The patients who are in need of a home-nurse or a medical assistant can make use of the system which not only cuts down the cost of giving salaries to them but also provides a 24x7 monitoring of their health condition and alerts the doctor or their responders in critical situations. The system can also be used for the patients in coma stage for monitoring and analyzing their health from their homes, so that the hospital can accommodate more patients who are in serious condition and need treatment/medication.

1.3 FEATURES

- The data is collected in real-time and the live data is displayed in the webpage and in the Blynk app.
- The abnormal values of body health parameters are detected and are alerted to the doctors by fixing a threshold value.
- The room temperature and humidity is also measured and displayed which helps the doctor to know about the patient's surrounding environment.
- The data is stored and analyzed in the cloud platform which thus maintains the patient's history and helps the doctor in understanding their health conditions quickly and easily.
- The patient history is secured in the cloud platform by password, only authorized persons can view it.

CHAPTER 2

LITERATURE REVIEW

S.NO	SOURCE	METHODOLOGY	IMPLICATIONS
1.	"E-health monitoring system". Authors: *Aleksandar Kotevski, Natasa Koceska and Saso Koceski..*	• Open m-Health platform for data collection and visualization. • Mobile app done via Eclipse using Java language. • Samsung Smart SDK have been used to get notification from doctor in TV.	• Using web or mobile application, patients' data can be collected easily and efficiently. • Smart TV application shows reminders and notifications. • Data visualized in the form of graph and bars
2.	"Iot Patient Health Monitoring System". Authors: Shola Usha Rani, Antony Ignatious, Bhava Vyasa Hari, Balavishnu V J. .	• Uses MQTT client to send data from sensors to cloud platform. • AWS Iot cloud helps the data to be stored in the DynamoDB database. • Uses Android app to see the data collected from the sensors messages from doctors	• The reading of various important indications of the patient and after that evaluate at cloud. • web interface is used to give a pictorial representation of information. • The android application shows parameters heart rate, SPO2 , last fall detected, and current detected sleep state

3.	"A Review on IOT Based Health Care Monitoring System". Authors :*Pravin Kshirsagar, Akshay Pote, K.K.Paliwal,VaibhavHendre,Pranav Chippalkatti and Nikhil Dhabekar*.	● Signals send to the Raspberry Pi through speaker circuit and flag molding unit (scu). ● MIT application innovator programming is utilized for exchange these parameters (Patient's body temperature, Blood Pressure, and Respiration rate and body developments) from Raspberry Pi to Android App	● patient checking framework is screen patient's body temperature, Blood weight, and Respiration rate and body developments utilizing Raspberry Pi. ● Utilizing IP address anyone can screen the patient's wellbeing status anyplace on the planet utilizing PCs, tablets and advanced cell phones
4.	"A Review on Health Monitoring Systems using IoT". Authors: *Harshitha Bhat Nishmitha Shetty and Ankitha Shetty.*	● Attempt is made to collect the parameters such as body temperature, heart beat rate, sugar level ● Using microcontroller, can send the send the data from the sensors to the cloud.	● The system determines the pulse rate and body temperature per minute or as per the time specified and then sends it to an android application. ● Doctors can obtain the results in their cell phone so that immediately the doctor can attend the patient for further treatment if necessary.

5.	"A SURVEY ON HEALTH CARE MONITORING SYSTEM USING IOT ". Authors : *Sasi Priya Saminathan, K.Geetha.*	- The proposed system collects real time data from the patients and delivers an updated patients status to the medical professionals and to the caretakers using WSN. - MQTT is light weight protocol used for transfer the messages.	- It monitors the subject's vital parameters such as temperature, pressure, fall detection, breath activity and ECG through PHD prototype model - The system sends an alert message to the caretakers and doctors in case of any abnormality through WBAN
6.	"Personalized Health Monitoring System of Elderly Wellness at the Community Level in Hong Kong." AUTHOR: *LISHA YU, WAI MAN CHAN2, YANG ZHAO , AND KWOK-LEUNG TSUI.* IEEE journal	- They can easily monitor and analyze an elderly's overall activity and vital signs using a wearable wellness tracker and an all-in-one satiation-based monitoring device. - A data-preparing scheme for acquiring data and processing data from multiple monitoring devices, and propose a personalized scheme for forecasting the elderly'sone-day-	- A smart personalized health monitoring system of elderly general wellness and demonstrated its implementation. - An efficient data preparing scheme to extract raw data from the electronic wearable tracker and an all-in-one station-based health monitoring device.

		ahead wellness condition via data integration and statistical learning.	
7.	“Three Tier Architecture for IoT Driven Health Monitoring System Using Raspberry Pi ” . IEEE journal Author: *Neel Kamal, Prasun Ghosal*	● The heartbeat, and body position movements constantly, and sends this information to site pages and crisis centers/services from the remote location. ● The proposed and implemented system prototype uses raspberry pi that is driven by Internet of Things (IoT) connected through different sensors DS18B20, ADXL345, ADC1015 and heartbeat sensor.	● The system can inform the wellness status of the inhabitant to the caregiver in advance. ● The proposed system generate real time graph of data that are body temperature, heart beat and body position that are collected through GPIO for better visualization of patient data
8.	“Wireless, Multipurpose In-Home Health MonitoringPlatform: Two Case Trials”. IEEE journal AUTHOR: *Sakari Junnila, Harri Kailanto, Juho*	● A general purpose home area sensor network and monitoring platform that is intended for e-Health applications, ranging from elderly monitoring to early	● We propose a health monitoring platform consisting of a chosen set of sensors, a Zigbee network, a home client, and a remote server.

	Merilahti, Antti-Matti Vainio, Antti Vehkaoja, Mari Zakrzewski, and Jari Hyttinen	homecoming after a hospitalization period. ● Consists of a chosen set of sensors, a wireless sensor network, a home client, and a distant server.	● The platform is easily adaptable for various user needs with the aid of a common sensor interface.
9.	"Smart Health Monitoring System through IOT" AUTHOR: *Ananth S, Sathya P and Madhan Mohan P* IEEE journal	● Patients with abnormal health conditions can be quickly monitored through a smart health care system and provide a rapid solution for the patients. ● The main aim of this work is to provide extensive research in capturing the sensor data, analyzing the data and providing a feedback to patients based on different health parameters.	● We acquire ECG & PPG signals from that we are extracting heart rate. ● Through these parameters, we are predicting the heart condition based on their heart rate and age factor. ● If the patient is having abnormal heart rate or any kind of heart disease trigger had been sent to particular person's relation or locality doctor as a precaution.
10.	"COVID-SAFE: An IoT-Based System for	● They present a potential application of the Internet of Things (IoT) in	● The proposed system integrates a wearable IoT node with a smartphone app,

	Automated Health Monitoring and Surveillance in Post-Pandemic Life ” AUTHOR: *SEYED SHAHIM VEDAEI , AMIR FOTOVVAT , MOHAMMAD REZA MOHEBBIAN , GAZI M. E. RAHMAN* , *KHAN A.WAHID, PAUL BABYN , HAMID REZA MARATEB , MARJAN MANSOURIAN , AND RAMIN SAMI.* IEEE journal	healthcare and physical distance monitoring for pandemic situations. • The proposed framework consists of three parts: a lightweight and low cost IoT node, a smartphone application (app), and fog-based Machine Learning (ML) tools for data analysis and diagnosis.	by which the IoT sensor node can collect a user's health parameters, such as temperature and blood oxygen saturation, and the smartphone connects to the network to send the data to the server. • A voice coughing-detector continually monitors the user's voice and records the number and severity of coughing. • The system can assist participants in monitoring their daily activities and minimize the risk of exposure to the Coronavirus.
11.	“A Smart Health System: Monitoring Comatose Patient's Physiological Conditions Remotely ”. AUTHOR: *Tati Erlina, Marreza*	• A smart health monitoring system for family members of comatose patients in the intensive care units is built. • The sensors detect the patient's heartbeat rate,	• The system which have been developed in this research can monitor physiological conditions of patients, proven by high

	Rifa Saputra & Rahmi Eka Putri. IEEE journal	respiratory rate and the opening of eyelids that indicate the patient's consciousness. • The data that have been collected from the sensors then is sent to server by the internet.	percentage of accuracy in detecting heartbeat rate; 98%, respiratory rate; 94% and eyelids status of patients; 100%. • The system is also capable of producing notification when abnormal condition is detected. • The android application developed monitors data, physiological condition and receives notification when anomaly data is detected.
12.	"Smart Health Care Monitoring System based on Internet of Things (IOT)". AUTHOR: *V. Baby Shalini* IEEE journal.	• In this paper, an IoT based Health Monitoring System is built using heartbeat and blood pressure sensor, Arduino UNO, GSM and GPS Module, ThingSpeak. • Heartbeat Sensor measures the speed	• An IoT based health monitoring system was developed which checks the blood pressure and pulse rate of a specific person, which are shown on LCD. • The readings of sensor's are directed to a

		of a patient's heartbeat in beat per minutes and BP sensor measures the blood pressure of patient. • The Arduino UNO constantly peruse the analog or digital input signals from heartbeat as well as blood pressure sensors and forward it to the ThingSpeak. • Using API keys in ThingSpeak, the doctor and patient family can see the patient health data over a period of time. • GSM is used for sending messages(SMS) to the authorized persons like doctors. GPS is used for collecting the geographical situation of the patient.	server wirelessly and received by authorized personals. • With the values received, the doctor diagnoses the sickness of the patient and recommend medication.
13.	"IoT Augmented Health Monitoring System" .IEEE journal	• In this project, raspberry pi is used which acts as a control unit, DS18B20 is used	• This system provides continuous monitoring of patient after

	AUTHOR: *A. Nishitha Reddy , Achsah Mary Marks , S.R.S Prabaharan, S.Muthulakshmi.*	for sensing body temperature and pulse rate sensor SEN-11574 is interfaced to raspberry pi which senses the pulse. • The measured values are sent to cloud to continuously monitor. • A doctor or caretaker can view the data that is being sent and if there is any variation they can take appropriate measures like call an ambulance or sent a message so that the patient is saved. • Android app is developed and data from cloud can be monitored on app. • In case of any emergency, call for help in the app calls to emergency number. There is also an option to send text message to the doctor.	discharge from hospital. • In the system implemented mainly two parameters body temperature and pulse rate are measured and data from sensor is sent to cloud and an android app is developed for monitoring data from cloud.

14.	"Health Monitoring System Using Internet of Things". AUTHOR: *Nurassyl Zholdas, Octavian Postolache, Madina Mansurova* IEEE journal	• This paper is aimed at building a monitoring system for patients with diabetes mellitus. It contains ESP32, Heart rate (pulse) sensor, Temperature Sensor, Red LED. • The DS18B20 sensor is used to measure body temperature and PPG sensor is used to measure heart rate. • The red LED connected to the ESP32, blinks according to the heart rate (pulse) sensor. The ESP32 sends data from these sensors to the ThingSpeak platform. • The ThingSpeak platform and ThingView mobile application are used to collect and display the data obtained from the sensors. ThingSpeak enables computers to store data in the cloud.	• By integrating sensor devices with the monitoring system, a patient's complete history of vital signs (heart rate, steps and blood glucose) is collected and analyzed. • The monitoring system is scalable to accommodate the growing volume of sensory data from sensor devices and the number of patients. • The monitoring system can be used as a personal tool for monitoring patients with diabetes and for self-monitoring.

15.	Internet of things (IoT) based smart health care system AUTHOR *Vikas Vippalapalli & Snigdha Ananthula.* IEEE journal	• In this paper an IoT based smart health care is built using Arduino Fio, LM35 temperature sensor, pulse rate sensor, BP sensor, LabVIEW software. • The different sensors are placed at the respective locations on the human body and are connected to the Arduino board. • The values of temperature, pulse rate and blood pressure are calculated by Arduino from sensors. • The processing of ECG signal is done by using the biomedical tool kit present in LabView. • These values are wirelessly transmitted and seen on the Arduino serial monitor.	• In this paper, tele-monitoring application is presented which allows the doctor to view the patient's vital parameters remotely and dynamically in a Web page doesn't need to have any special requirement on the PC; an Internet access is enough. • For the patient side, a home-based LabVIEW application which is embedded in home PC is required.

CHAPTER 3

DESIGN

3.1 COMPONENTS USED

3.1.1 ESP 32 Development Board

To handle all the processing related to both monitoring the patient's health and the weather,this system uses an ESP32. ESP32 has many applications when it comes to the Internet of Things (IoT).This provides Wi-Fi (and in some models) dual-mode Bluetooth connectivity to embedded devices. The ESP32 chip has a Tensilica Xtensa LX6 microprocessor in both dual-core and single-core variations, with a clock rate of over 240 MHz. The ESP32 enables connectivity to integrated Wi-Fi through the 802.11 b/g/n/e/i/. Moreover, dual-mode Bluetooth is made possible with the v4.2 BR/EDR and features Bluetooth Low Energy (BLE). All the sensors are connected to this ESP32.

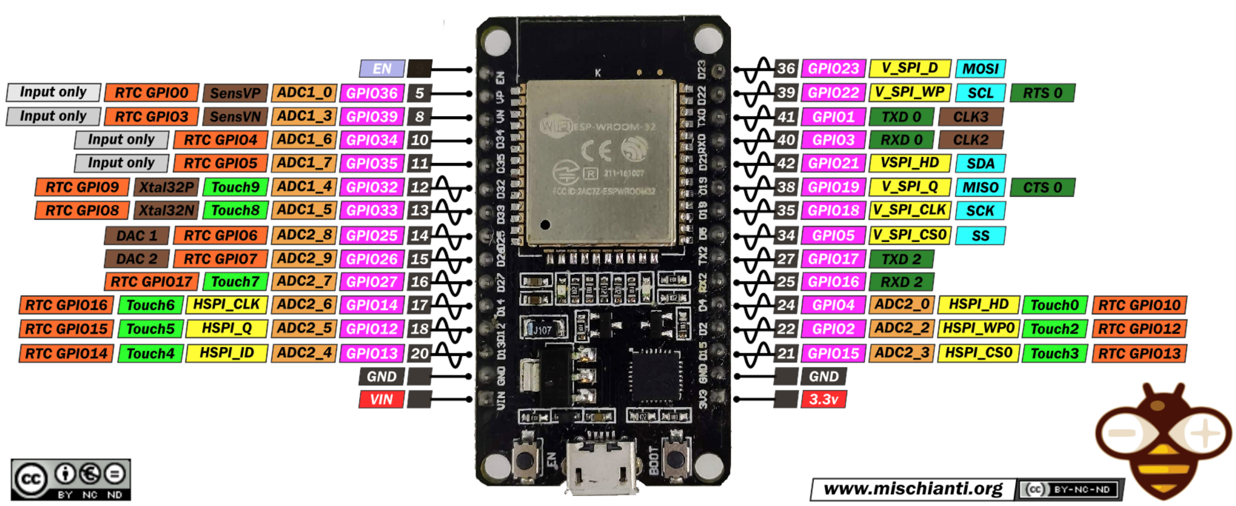


Fig 1: ESP32

Figure 1 shows the detailed explanation of ESP32 pins

3.1.2 DHT 11 Temperature and Humidity Sensor

The DHT11 is a basic, ultra low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data pin (no analog input pins needed). It's fairly simple to use, but requires careful timing to grab data.

Fig 2: DHT11 Sensor

Figure 2 shows the image of the DHT11 sensor apparatus

3.1.3 MAX 30100 Pulse Oximeter Sensor

The MAX30100 is an integrated pulse oximeter and heart rate monitor sensor solution. It combines two LEDs, a photodetector, optimized optics, and low-noise analog signal processing to detect pulse oximetry and heart-rate signals. The MAX30100 operates from 1.8V and 3.3V power supplies and can be powered down through software with negligible standby current, permitting the power supply to remain connected at all times.

Fig 3: MAX 30100 Pulse Oximeter Sensor

Figure 3 shows the image of the MAX30100 pulse oximeter sensor apparatus

3.1.4 DS18B20 One-Wire Waterproof Temperature Sensor

The DS18B20 digital thermometer provides 9-bit to 12-bit Celsius temperature measurements and has an alarm function with nonvolatile user-programmable upper and lower trigger points. The DS18B20 communicates over a 1-Wire bus that by definition requires only one data line (and ground) for communication with a central microprocessor. In addition, the DS18B20 can derive power directly from the data line ("parasite power"), eliminating the need for an external power supply. Each DS18B20 has a unique 64-bit serial code, which allows multiple DS18B20s to function on the same 1-Wire bus. Thus, it is simple to use one microprocessor to control many DS18B20s distributed over a large area.

Fig 4: DS18B20 One-Wire Waterproof Temperature Sensor

Figure 4 shows the image of DS18B20 One-Wire Waterproof Temperature Sensor apparatus.

3.2 PROPOSED SYSTEM

3.2.1 BLOCK DIAGRAM

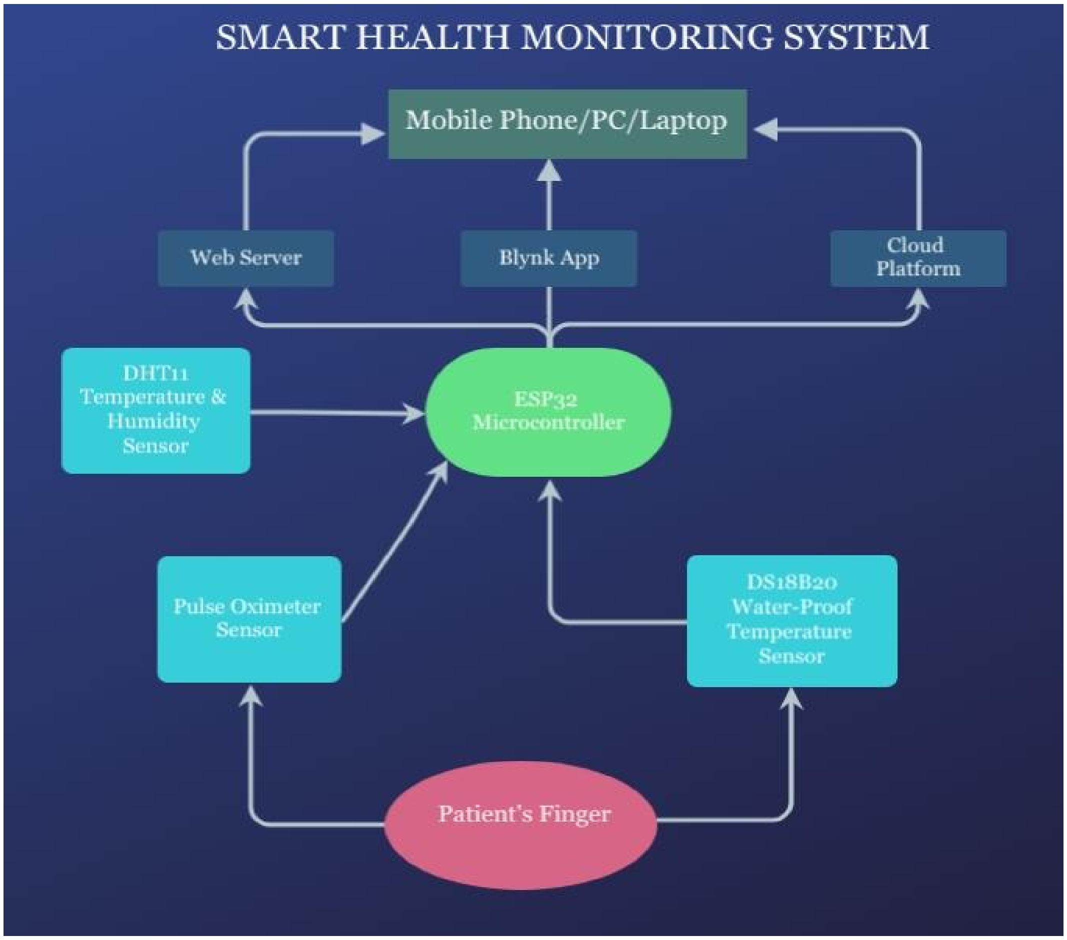


Fig 5: Block Diagram

3.2.2 DESIGN APPROACH

The proposed system is designed with three sensors(MAX 30100 pulse oximeter sensor, DHT 11 Temperature & Humidity sensor, DS18B20 one-wire waterproof temperature sensor) interfaced with the ESP32 microcontroller. The patient should place his/her finger on the sensor to sense and measure the body health parameters. The data from the sensors are sent to the webpage hosted by ESP32 and is developed/coded along with the ESP 32 code in Arduino IDE. Once the ESP32 is powered and connected to the Wi-Fi network it hosts the webpage in the local network. The data is also sent to the Blynk app using the authentication token generated by the creation of a new project in the application. The data is visualized using different widgets and it also gives alert notifications when the body health parameters cross a certain threshold. The data is uploaded to ThingSpeak cloud platform where the authorized users like doctors can view the patient history in the form of graphs. Using the MATLAB analysis feature in ThingSpeak, we can analyze the patient's health data. The patient history can also be downloaded in CSV, JSON or XML format.

3.2.2 HARDWARE ANALYSIS

The pulse oximeter sensor module gets the values of a patient's heart rate(in BPM) and blood oxygen levels. The ESP32 receives the values sent by all the sensors and sends them to the BLYNK app and ThingSpeak cloud. It also hosts a webpage where the live data from the sensors can be seen.The received Beats per minute is viewed in Field 4 of ThingSpeak. The blood oxygen levels received are viewed in field 5 of ThingSpeak. The body temperature sensor senses the patient's body temperature values in degree celsius. The DHT sensor measures the room temperature and room humidity.The room temperature is viewed in field1 of ThingSpeak. The humidity of the room is viewed in field 2 of ThingSpeak.

3.2.4 EXPERIMENTAL SETUP

1. **Hardware Setup**

ESP 32 is used as the microcontroller. It has an inbuilt Wi-Fi module and it is best suited for IoT based systems. For the three sensors, the Vcc is connected with 3.3V and ground is connected with the ground of the microcontroller. The data pin of DHT 11 is connected to D18 pin, DS18B20 waterproof temperature sensor is connected to D5 pin, the SCL,SDA and INT pins of MAX 30100 pulse oximeter is connected with D22,D21 & D19 pins of ESP 32 microcontroller.

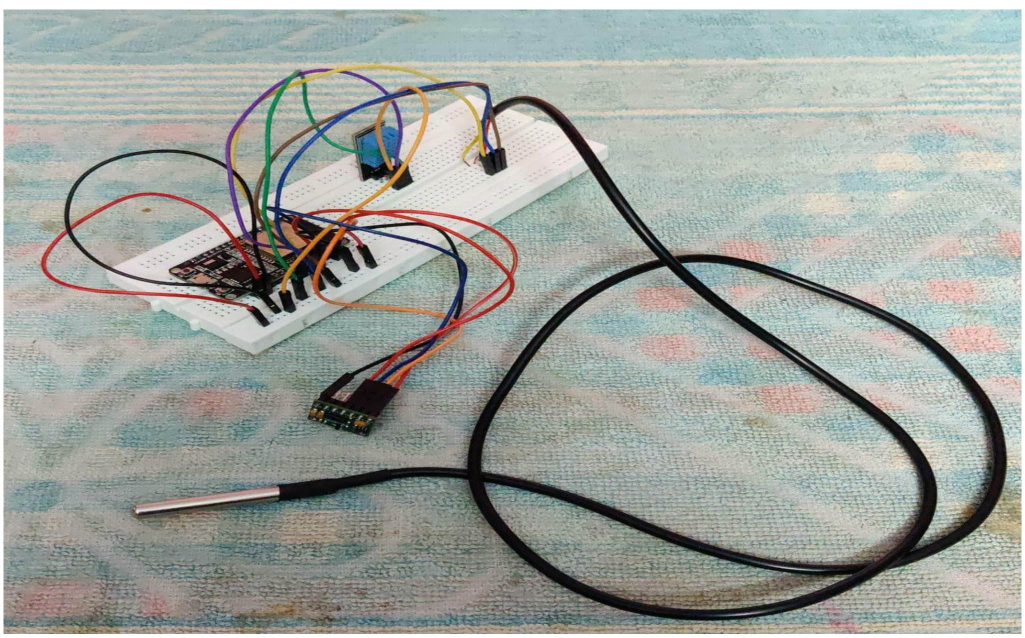


Fig 6: Hardware Setup of the project

2. Cloud interface

A ThingSpeak Channel is created with five fields and the write API key & channel-ID is copied and entered in the ESP 32 code so that the measured values can be uploaded. The channel is private, only authorized persons can access it.

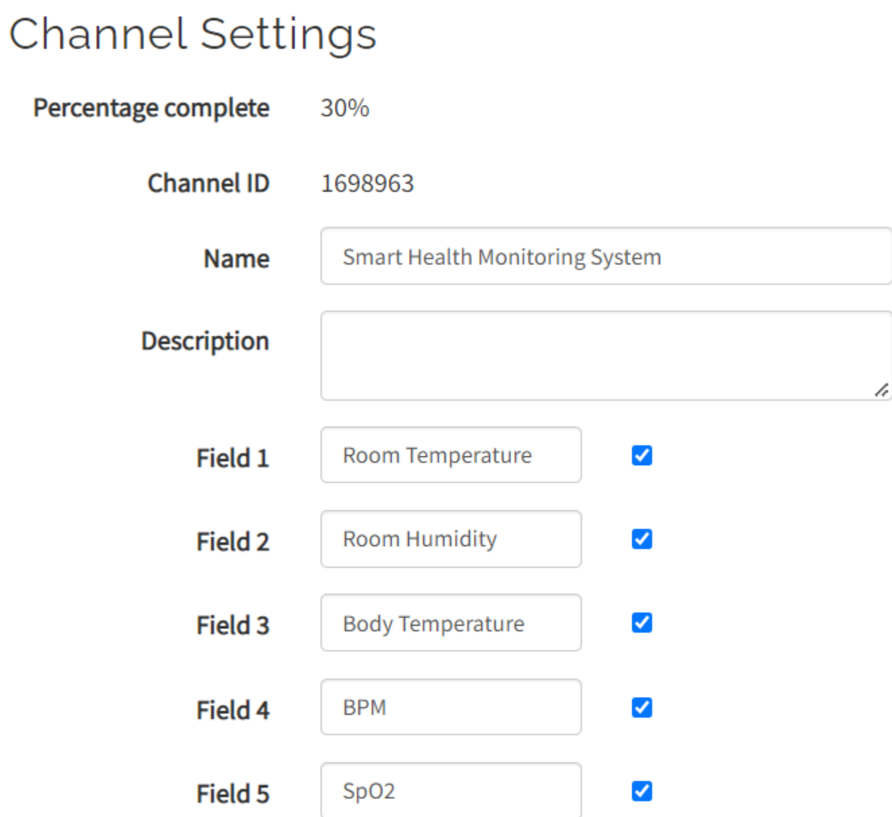


Fig 7: ThingSpeak Channel Settings

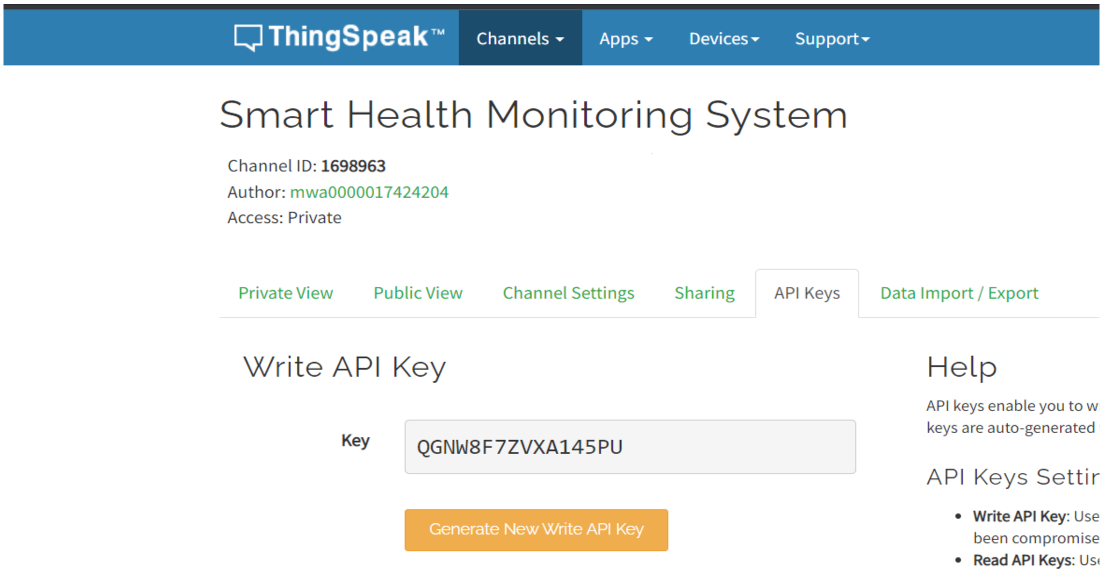


Fig 8: ThingSpeak Channel API keys

3. Blynk Application

A new Blynk project is created with various widgets and an alert notification. The authentication token is copied and entered in the ESP 32 code so that the measured values can be visualized in different widgets.

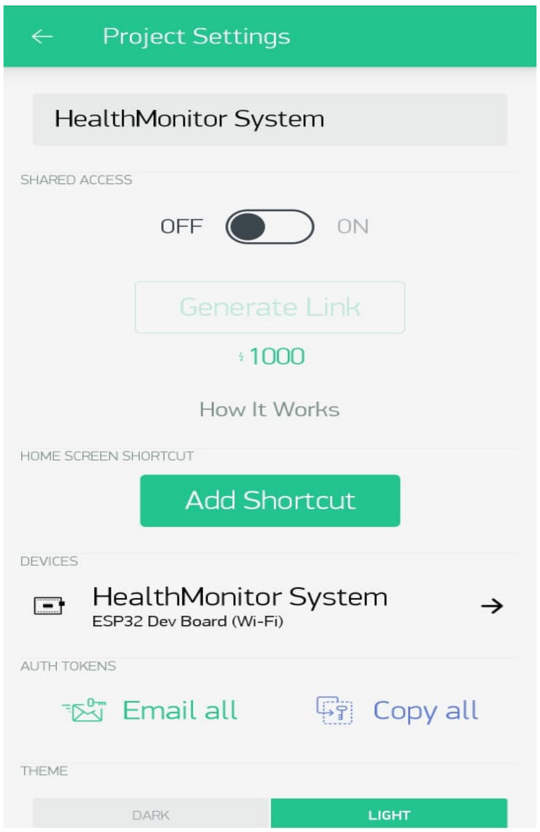


Fig 9: Blynk project settings

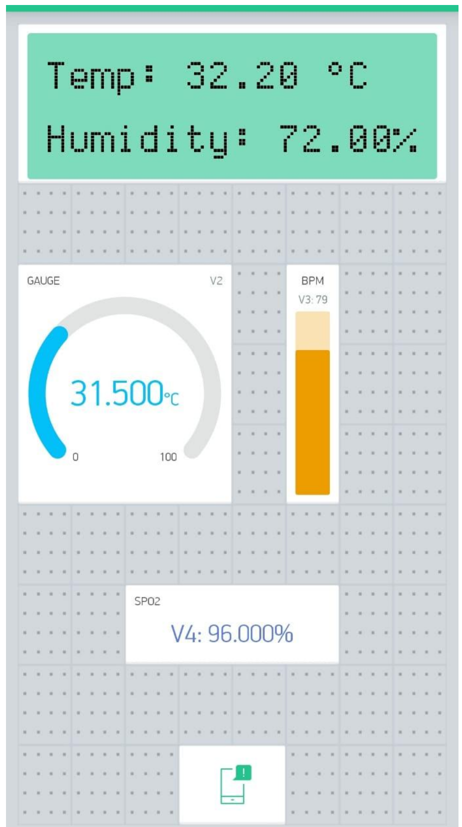


Fig 10: Widget Placement

3.3 SOFTWARE ANALYSIS

3.3.1 CIRCUIT DIAGRAM

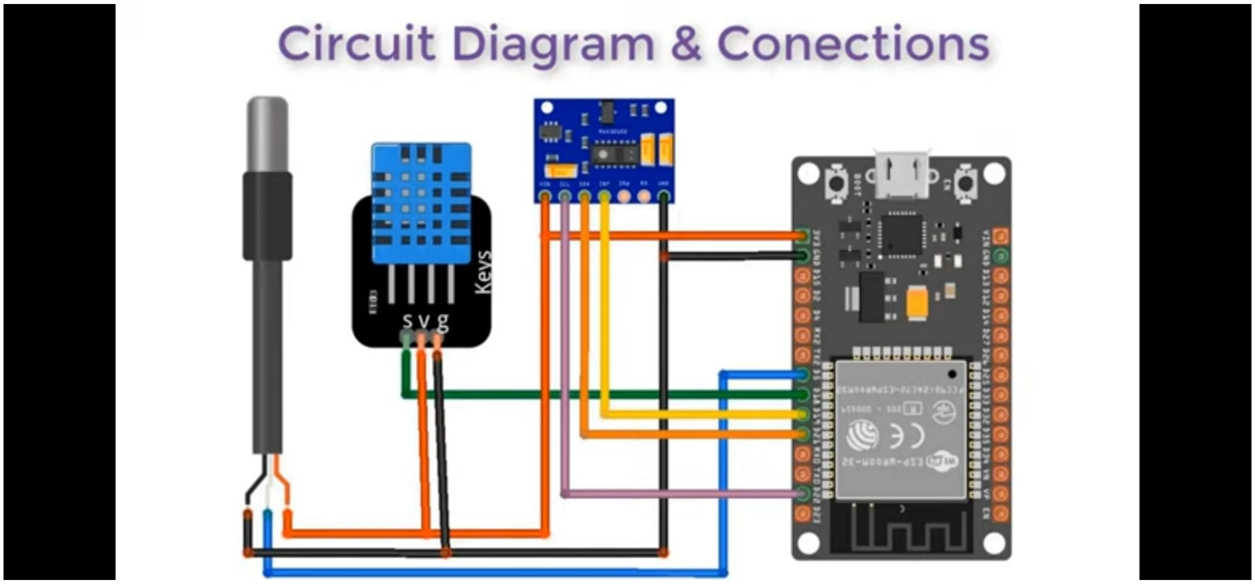


Fig 11: Circuit diagram

3.3.2 ESP32 CODE

```
#include
<C:\Users\KEERTHIVASAN\AppData\Local\Arduino15\packages\esp32\hardware\esp32\1.0.6\l
ibraries\WiFi\src\WiFi.h>
#include <WebServer.h>
#include <Wire.h>
#include "MAX30100_PulseOximeter.h"
#include <OneWire.h>
#include <DallasTemperature.h>
#include <DHTStable.h>
#include<ThingSpeak.h>
#include <BlynkSimpleEsp32.h>

#define DHT11_PIN 18
```

```
#define DS18B20 5
#define REPORTING_PERIOD_MS     1000

float temperature;
float Humidity;
float bodytemperature;
float BPM,SpO2;

WiFiClient client;
BlynkTimer timer;
WidgetLCD lcd1(V1);
/*Put your SSID & Password*/
const char* ssid = "OnePlus 7";
const char* password = "12345678";
char auth[] = "UfVwwD8sC5kX_H8JPSSEyKBsD_x1lH3R";

const char* host = "api.thingspeak.com";
const char* api_key= "QGNW8F7ZVXA145PU";
unsigned long chanNum = 1698963;

DHTStable DHT;
PulseOximeter pox;
uint32_t tsLastReport = 0;
OneWire oneWire(DS18B20);
DallasTemperature sensors(&oneWire);


WebServer server(80);

void onBeatDetected()
{
```

```
  Serial.println("Beat!");
}

 void SEND_LCD()
{
  String temperature_text = "Temp: ";
  String humidity_text = "Humidity: ";

  temperature_text += temperature;
  temperature_text += " °C";
  humidity_text += Humidity;
  humidity_text += "%";

  lcd1.clear();
  lcd1.print(0, 0, temperature_text);
  lcd1.print(0, 1, humidity_text);

  Blynk.virtualWrite(V2, bodytemperature);
  Blynk.virtualWrite(V3, BPM);
  Blynk.virtualWrite(V4, SpO2);
  if (bodytemperature>37)
  {
    Blynk.notify("The patient is suffering from fever. Kindly give attention!!");
  }
}

void setup() {
  Serial.begin(115200);
  pinMode(19, OUTPUT);
  delay(100);
  Blynk.begin(auth, ssid, password);
```

```
  timer.setInterval(1000L, SEND_LCD);
  ThingSpeak.begin(client);
  Serial.println("Connecting to ");
  Serial.println(ssid);

  //connect to your local wi-fi network
  WiFi.begin(ssid, password);

  //check wi-fi is connected to wi-fi network
  while (WiFi.status() != WL_CONNECTED) {
  delay(1000);
  Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected..!");
  Serial.print("Got IP: ");  Serial.println(WiFi.localIP());

  server.on("/", handle_OnConnect);
  server.onNotFound(handle_NotFound);

  server.begin();
  Serial.println("HTTP server started");
  pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);
}


void loop() {
  Blynk.run();
  timer.run();
  server.handleClient();
  pox.update();
```

```
sensors.requestTemperatures();
int chk = DHT.read11(DHT11_PIN);

temperature = DHT.getTemperature();
Humidity = DHT.getHumidity();
BPM = pox.getHeartRate();
SpO2 = pox.getSpO2();
bodytemperature = sensors.getTempCByIndex(0);


if (millis() - tsLastReport > REPORTING_PERIOD_MS)
{
  Serial.print("Room Temperature: ");
  Serial.print(DHT.getTemperature(),1);
  Serial.println("°C");

  Serial.print("Room Humidity: ");
  Serial.print(DHT.getHumidity(),1);
  Serial.println("%");

  Serial.print("Body Temperature: ");
  Serial.print(bodytemperature);
  Serial.println("°C");

  Serial.print("BPM: ");
  Serial.println(BPM);

  Serial.print("SpO2: ");
  Serial.print(SpO2);
  Serial.println("%");
```

```
    Serial.println("*********************************");
    Serial.println();
    ThingSpeak.setField(1,temperature);
    ThingSpeak.setField(2,Humidity);
    ThingSpeak.setField(3,bodytemperature);
    ThingSpeak.setField(4,BPM);
    ThingSpeak.setField(5,SpO2);
    ThingSpeak.writeFields(chanNum, api_key);
    delay(100);

    tsLastReport = millis();
  }

}

void handle_OnConnect() {

  server.send(200, "text/html", SendHTML(temperature, Humidity, BPM, SpO2,
bodytemperature));
}

void handle_NotFound(){
  server.send(404, "text/plain", "Not found");
}

  String SendHTML(float temperature,float Humidity,float BPM,float SpO2,float
bodytemperature){
  String ptr = "<!DOCTYPE html>";
  ptr +="<html>";
  ptr +="<head>";
  ptr +="<title>ESP32 Patient Health Monitoring</title>";
```

```
  ptr +="<meta name='viewport' content='width=device-width, initial-scale=1.0'>";
  ptr +="<link href='https://fonts.googleapis.com/css?family=Open+Sans:300,400,600'
rel='stylesheet'>";
  ptr +="<style>";
  ptr +="html { font-family: 'Open Sans', sans-serif; display: block; margin: 0px auto; text-align:
center;color: #444444;}";
  ptr +="body{margin: 0px;} ";
  ptr +="h1 {margin: 50px auto 30px;} ";
  ptr +=".side-by-side{display: table-cell;vertical-align: middle;position: relative;}";
  ptr +=".text{font-weight: 600;font-size: 19px;width: 200px;}";
  ptr +=".reading{font-weight: 300;font-size: 50px;padding-right: 25px;}";
  ptr +=".temperature .reading{color: #F29C1F;}";
  ptr +=".humidity .reading{color: #3B97D3;}";
  ptr +=".BPM .reading{color: #FF0000;}";
  ptr +=".SpO2 .reading{color: #955BA5;}";
  ptr +=".bodytemperature .reading{color: #F29C1F;}";
  ptr +=".superscript{font-size: 17px;font-weight: 600;position: absolute;top: 10px;}";
  ptr +=".data{padding: 10px;}";
  ptr +=".container{display: table;margin: 0 auto;}";
  ptr +=".icon{width:65px}";
  ptr +=".keerthi{display: block;width: 100%; border: none;background: #FF0000; color: white;
padding: 14px 28px; font-size: 20px; cursor: pointer; text-align: center;}";
  ptr +="</style>";
  ptr +="</head>";
  ptr +="<body>";
  ptr +="<h1>ESP32 based Smart Health Monitoring</h1>";
  ptr +="<h3>IoT Domain Analyst Project</h3>";
  ptr +="<div class='container'>";

  ptr +="<div class='data temperature'>";
  ptr +="<div class='side-by-side icon'>";
```

```
  ptr +="<svg enable-background='new 0 0 19.438 54.003'height=54.003px id=Layer_1
version=1.1 viewBox='0 0 19.438 54.003'width=19.438px x=0px xml:space=preserve
xmlns=http://www.w3.org/2000/svg xmlns:xlink=http://www.w3.org/1999/xlink
y=0px><g><path
d='M11.976,8.82v-2h4.084V6.063C16.06,2.715,13.345,0,9.996,0H9.313C5.965,0,3.252,2.715,3.
252,6.063v30.982";
  ptr
+="C1.261,38.825,0,41.403,0,44.286c0,5.367,4.351,9.718,9.719,9.718c5.368,0,9.719-4.351,9.71
9-9.718";
  ptr
+="c0-2.943-1.312-5.574-3.378-7.355V18.436h-3.914v-2h3.914v-2.808h-4.084v-2h4.084V8.82
H11.976z M15.302,44.833";
  ptr
+="c0,3.083-2.5,5.583-5.583,5.583s-5.583-2.5-5.583-5.583c0-2.279,1.368-4.236,3.326-5.104V2
4.257C7.462,23.01,8.472,22,9.719,22";
  ptr
+="s2.257,1.01,2.257,2.257V39.73C13.934,40.597,15.302,42.554,15.302,44.833z'fill=#F29C21
/></g></svg>";
  ptr +="</div>";
  ptr +="<div class='side-by-side text'>Room Temperature</div>";
  ptr +="<div class='side-by-side reading'>";
  ptr +=(int)temperature;
  ptr +="<span class='superscript'>&deg;C</span></div>";
  ptr +="</div>";

  ptr +="<div class='data humidity'>";
  ptr +="<div class='side-by-side icon'>";
  ptr +="<svg enable-background='new 0 0 29.235 40.64'height=40.64px id=Layer_1
version=1.1 viewBox='0 0 29.235 40.64'width=29.235px x=0px xml:space=preserve
xmlns=http://www.w3.org/2000/svg xmlns:xlink=http://www.w3.org/1999/xlink y=0px><path
```

```
d='M14.618,0C14.618,0,0,17.95,0,26.022C0,34.096,6.544,40.64,14.618,40.64s14.617-6.544,14.
617-14.617";
  ptr +="C29.235,17.95,14.618,0,14.618,0z
M13.667,37.135c-5.604,0-10.162-4.56-10.162-10.162c0-0.787,0.638-1.426,1.426-1.426";
  ptr
+="c0.787,0,1.425,0.639,1.425,1.426c0,4.031,3.28,7.312,7.311,7.312c0.787,0,1.425,0.638,1.425
,1.425";
  ptr +="C15.093,36.497,14.455,37.135,13.667,37.135z'fill=#3C97D3 /></svg>";
  ptr +="</div>";
  ptr +="<div class='side-by-side text'>Room Humidity</div>";
  ptr +="<div class='side-by-side reading'>";
  ptr +=(int)Humidity;
  ptr +="<span class='superscript'>%</span></div>";
  ptr +="</div>";

  ptr +="<div class='data Heart Rate'>";
  ptr +="<div class='side-by-side icon'>";
  ptr +="<svg enable-background='new 0 0 40.542 40.541'height=40.541px id=Layer_1
version=1.1 viewBox='0 0 40.542 40.541'width=40.542px x=0px xml:space=preserve
xmlns=http://www.w3.org/2000/svg xmlns:xlink=http://www.w3.org/1999/xlink
y=0px><g><path
d='M34.313,20.271c0-0.552,0.447-1,1-1h5.178c-0.236-4.841-2.163-9.228-5.214-12.593l-3.425,
3.424";
  ptr
+="c-0.195,0.195-0.451,0.293-0.707,0.293s-0.512-0.098-0.707-0.293c-0.391-0.391-0.391-1.023,
0-1.414l3.425-3.424";
  ptr
+="c-3.375-3.059-7.776-4.987-12.634-5.215c0.015,0.067,0.041,0.13,0.041,0.202v4.687c0,0.552
-0.447,1-1,1s-1-0.448-1-1V0.25";
```

```
  ptr
+="c0-0.071,0.026-0.134,0.041-0.202C14.39,0.279,9.936,2.256,6.544,5.385l3.576,3.577c0.391,
0.391,0.391,1.024,0,1.414";
  ptr
+="c-0.195,0.195-0.451,0.293-0.707,0.293s-0.512-0.098-0.707-0.293L5.142,6.812c-2.98,3.348-
4.858,7.682-5.092,12.459h4.804";
  ptr
+="c0.552,0,1,0.448,1,1s-0.448,1-1,1H0.05c0.525,10.728,9.362,19.271,20.22,19.271c10.857,0,1
9.696-8.543,20.22-19.271h-5.178";
  ptr +="C34.76,21.271,34.313,20.823,34.313,20.271z
M23.084,22.037c-0.559,1.561-2.274,2.372-3.833,1.814";
  ptr
+="c-1.561-0.557-2.373-2.272-1.815-3.833c0.372-1.041,1.263-1.737,2.277-1.928L25.2,7.202L2
2.497,19.05";
  ptr +="C23.196,19.843,23.464,20.973,23.084,22.037z'fill=#26B999 /></g></svg>";
  ptr +="</div>";
  ptr +="<div class='side-by-side text'>Heart Rate</div>";
  ptr +="<div class='side-by-side reading'>";
  ptr +=(int)BPM;
  ptr +="<span class='superscript'>BPM</span></div>";
  ptr +="</div>";

  ptr +="<div class='data Blood Oxygen'>";
  ptr +="<div class='side-by-side icon'>";
  ptr +="<svg enable-background='new 0 0 58.422 40.639'height=40.639px id=Layer_1
version=1.1 viewBox='0 0 58.422 40.639'width=58.422px x=0px xml:space=preserve
xmlns=http://www.w3.org/2000/svg xmlns:xlink=http://www.w3.org/1999/xlink
y=0px><g><path
d='M58.203,37.754l0.007-0.004L42.09,9.935l-0.001,0.001c-0.356-0.543-0.969-0.902-1.667-0.9
02";
```

```
  ptr
+="c-0.655,0-1.231,0.32-1.595,0.808l-0.011-0.007l-0.039,0.067c-0.021,0.03-0.035,0.063-0.054,
0.094L22.78,37.692l0.008,0.004";
  ptr
+="c-0.149,0.28-0.242,0.594-0.242,0.934c0,1.102,0.894,1.995,1.994,1.995v0.015h31.888c1.101
,0,1.994-0.893,1.994-1.994";
  ptr +="C58.422,38.323,58.339,38.024,58.203,37.754z'fill=#955BA5 /><path
d='M19.704,38.674l-0.013-0.004l13.544-23.522L25.13,1.156l-0.002,0.001C24.671,0.459,23.88
5,0,22.985,0";
  ptr
+="c-0.84,0-1.582,0.41-2.051,1.038l-0.016-0.01L20.87,1.114c-0.025,0.039-0.046,0.082-0.068,0.
124L0.299,36.851l0.013,0.004";
  ptr
+="C0.117,37.215,0,37.62,0,38.059c0,1.412,1.147,2.565,2.565,2.565v0.015h16.989c-0.091-0.25
6-0.149-0.526-0.149-0.813";
  ptr +="C19.405,39.407,19.518,39.019,19.704,38.674z'fill=#955BA5 /></g></svg>";
  ptr +="</div>";
  ptr +="<div class='side-by-side text'>Blood Oxygen</div>";
  ptr +="<div class='side-by-side reading'>";
  ptr +=(int)SpO2;
  ptr +="<span class='superscript'>%</span></div>";
  ptr +="</div>";

  ptr +="<div class='data Body Temperature'>";
  ptr +="<div class='side-by-side icon'>";
  ptr +="<svg enable-background='new 0 0 19.438 54.003'height=54.003px id=Layer_1
version=1.1 viewBox='0 0 19.438 54.003'width=19.438px x=0px xml:space=preserve
xmlns=http://www.w3.org/2000/svg xmlns:xlink=http://www.w3.org/1999/xlink
y=0px><g><path
d='M11.976,8.82v-2h4.084V6.063C16.06,2.715,13.345,0,9.996,0H9.313C5.965,0,3.252,2.715,3.
252,6.063v30.982";
```

```
  ptr
+="C1.261,38.825,0,41.403,0,44.286c0,5.367,4.351,9.718,9.719,9.718c5.368,0,9.719-4.351,9.71
9-9.718";
  ptr
+="c0-2.943-1.312-5.574-3.378-7.355V18.436h-3.914v-2h3.914v-2.808h-4.084v-2h4.084V8.82
H11.976z M15.302,44.833";
  ptr
+="c0,3.083-2.5,5.583-5.583,5.583s-5.583-2.5-5.583-5.583c0-2.279,1.368-4.236,3.326-5.104V2
4.257C7.462,23.01,8.472,22,9.719,22";
  ptr
+="s2.257,1.01,2.257,2.257V39.73C13.934,40.597,15.302,42.554,15.302,44.833z'fill=#F29C21
/></g></svg>";
  ptr +="</div>";
  ptr +="<div class='side-by-side text'>Body Temperature</div>";
  ptr +="<div class='side-by-side reading'>";
  ptr +=(int)bodytemperature;
  ptr +="<span class='superscript'>&deg;C</span></div>";
  ptr +="<a class='keerthi' href=\"https://thingspeak.com/channels/1698963/private_show\"
target=\"_blank\">THINGSPEAK CLOUD</a>";
  ptr +="</div>";
  ptr +="</div>";
  ptr +="</body>";
  ptr +="</html>";
  return ptr;
}
```

MATLAB Analysis Code

```
readChannelID = 1698963;
BPMFieldID = 4;

readAPIKey = '2JA514BJ0CVH7N8A';

[BPM,timeStamp] = thingSpeakRead(readChannelID,'Fields',BPMFieldID, ...
                                    'numDays',1,'ReadKey',readAPIKey);
[maxBPM,maxBPMIndex] = max(BPM);
[minBPM,minBPMIndex] = min(BPM);

timeMaxbpm = timeStamp(maxBPMIndex);
timeMinbpm = timeStamp(minBPMIndex);

display(maxBPM,'Maximum BPM for the past 24 hours is');
display(minBPM,'Minimum BPM for the past 24 hours is');

if maxBPM > 79
   fprintf(['The patient has higher heart rate. So, he/she needs attention']);
end
if minBPM < 60
   fprintf(['The patient has lower heart rate. So, he/she needs treatment\n']);
end

humidityFieldID = 2;
humidity =
thingSpeakRead(readChannelID,'Fields',humidityFieldID,'NumMinutes',60,'ReadKey',readAPIK
ey);
% Calculate the average humidity
avgHumidity = mean(humidity);
display(avgHumidity,'Average Humidity');
```

```
TemperatureFieldID = 3;
[tempF,timeStamp] = thingSpeakRead(readChannelID,'Fields',TemperatureFieldID, ...
                                   'numDays',1,'ReadKey',readAPIKey);

[maxTempF,maxTempIndex] = max(tempF);
[minTempF,minTempIndex] = min(tempF);

timeMaxTemp = timeStamp(maxTempIndex);
timeMinTemp = timeStamp(minTempIndex);

display(maxTempF,'Maximum Temperature for the past 24 hours is');
display(minTempF,'Minimum Temperature for the past 24 hours is');

SpO2FieldID = 5;
[SpO2,timeStamp] = thingSpeakRead(readChannelID,'Fields',SpO2FieldID, ...
                                   'numDays',1,'ReadKey',readAPIKey);

[maxSpO2,maxSpO2Index] = max(SpO2);
[minSpO2,minSpO2Index] = min(SpO2);

timeMaxSpO2 = timeStamp(maxSpO2Index);
timeMinSpO2 = timeStamp(minSpO2Index);

display(maxSpO2,'Maximum Blood oxygen level for the past 24 hours is');
display(minSpO2,'Minimum Blood oxygen level for the past 24 hours is');
```

CHAPTER 4
RESULT AND ANALYSIS

Code Uploading and Serial Monitor Output

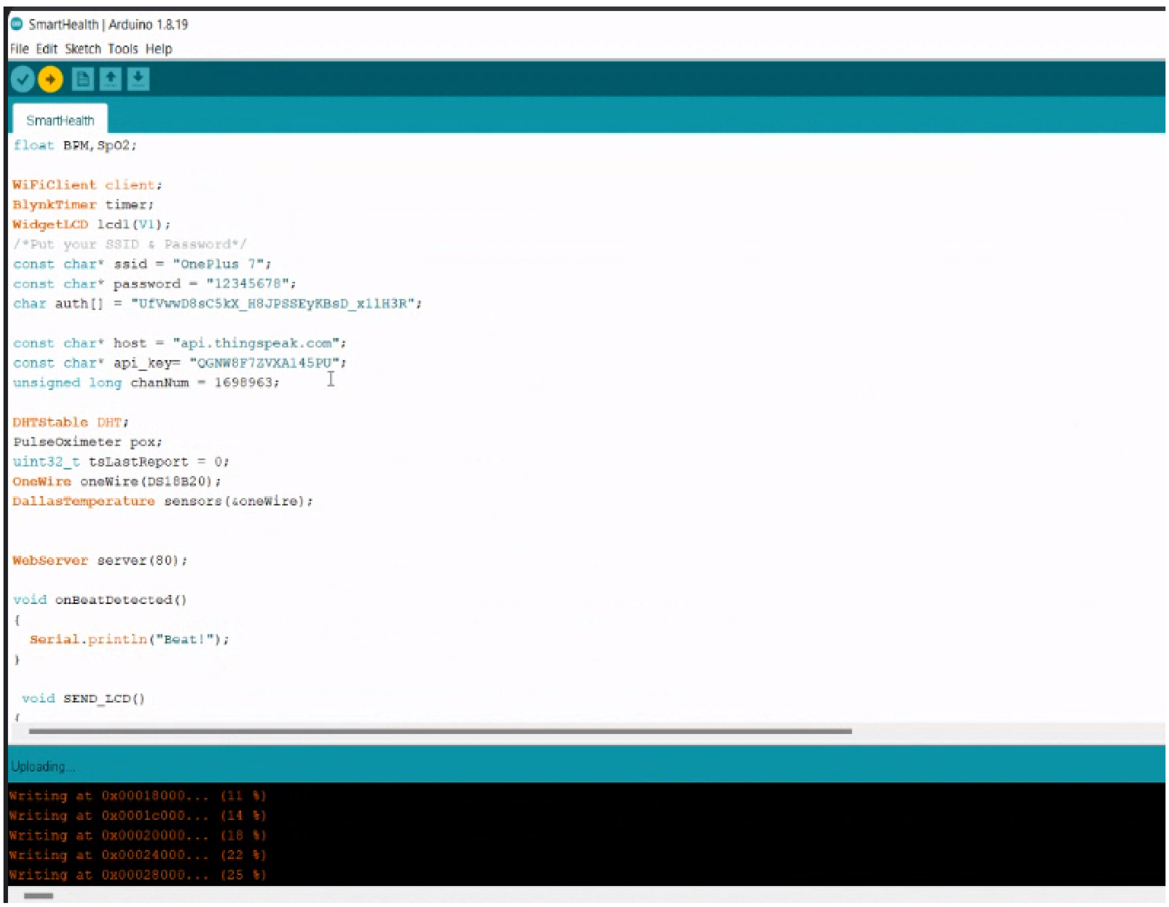
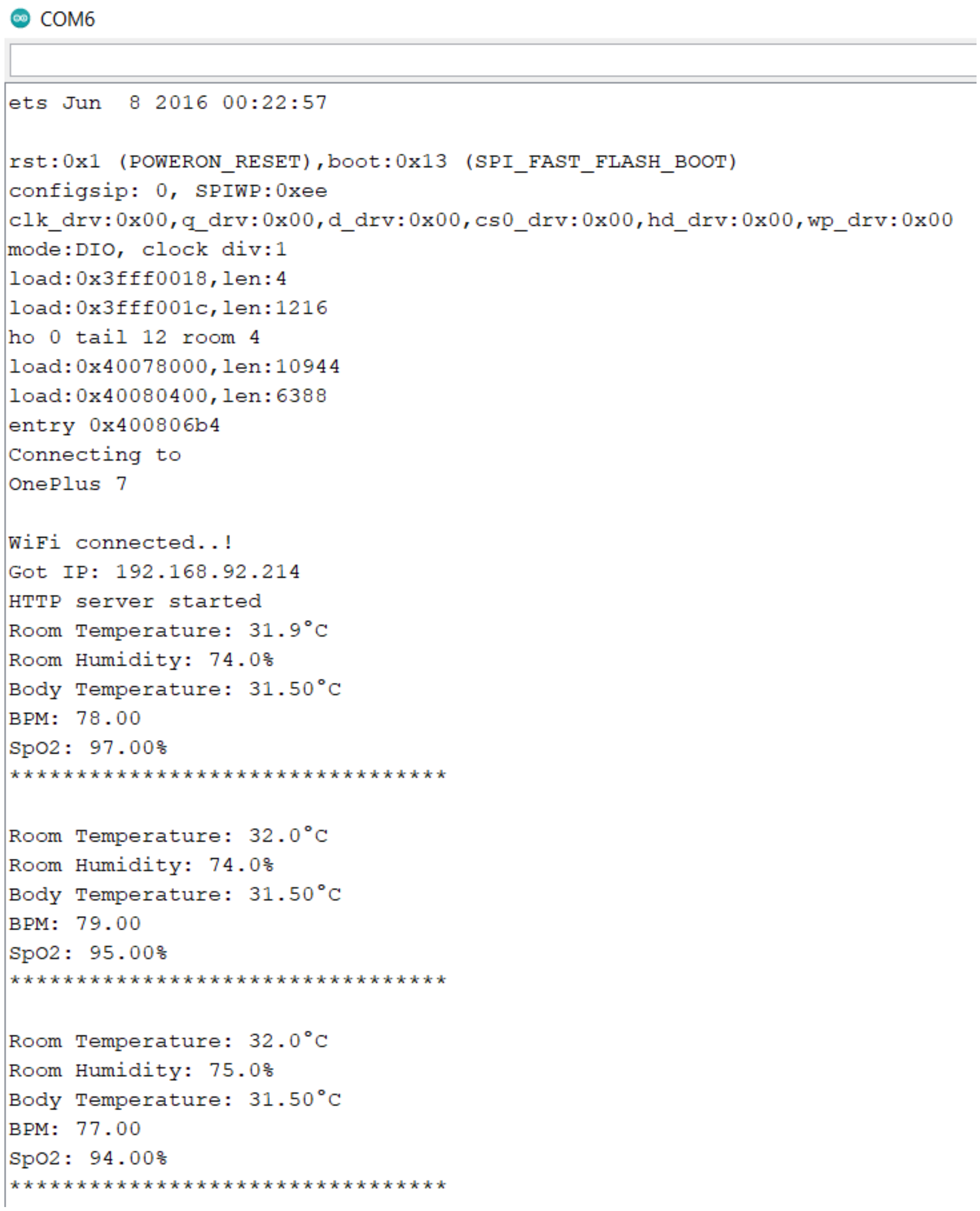


Fig 12: Arduino IDE code uploading



```
COM6

ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0018,len:4
load:0x3fff001c,len:1216
ho 0 tail 12 room 4
load:0x40078000,len:10944
load:0x40080400,len:6388
entry 0x400806b4
Connecting to
OnePlus 7

WiFi connected..!
Got IP: 192.168.92.214
HTTP server started
Room Temperature: 31.9°C
Room Humidity: 74.0%
Body Temperature: 31.50°C
BPM: 78.00
SpO2: 97.00%
*********************************

Room Temperature: 32.0°C
Room Humidity: 74.0%
Body Temperature: 31.50°C
BPM: 79.00
SpO2: 95.00%
*********************************

Room Temperature: 32.0°C
Room Humidity: 75.0%
Body Temperature: 31.50°C
BPM: 77.00
SpO2: 94.00%
*********************************
```

Fig 13: Serial Monitor

Webpage

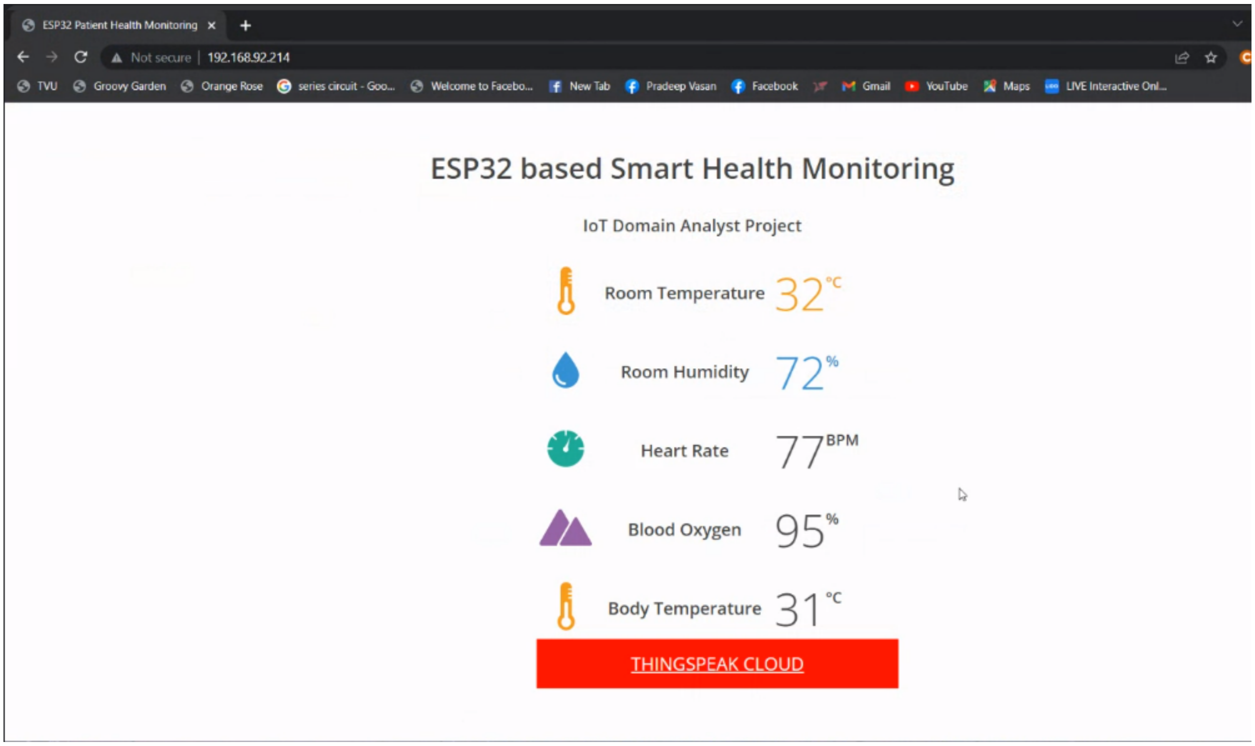


Fig 14: WebPage

ThingSpeak Cloud Dashboard

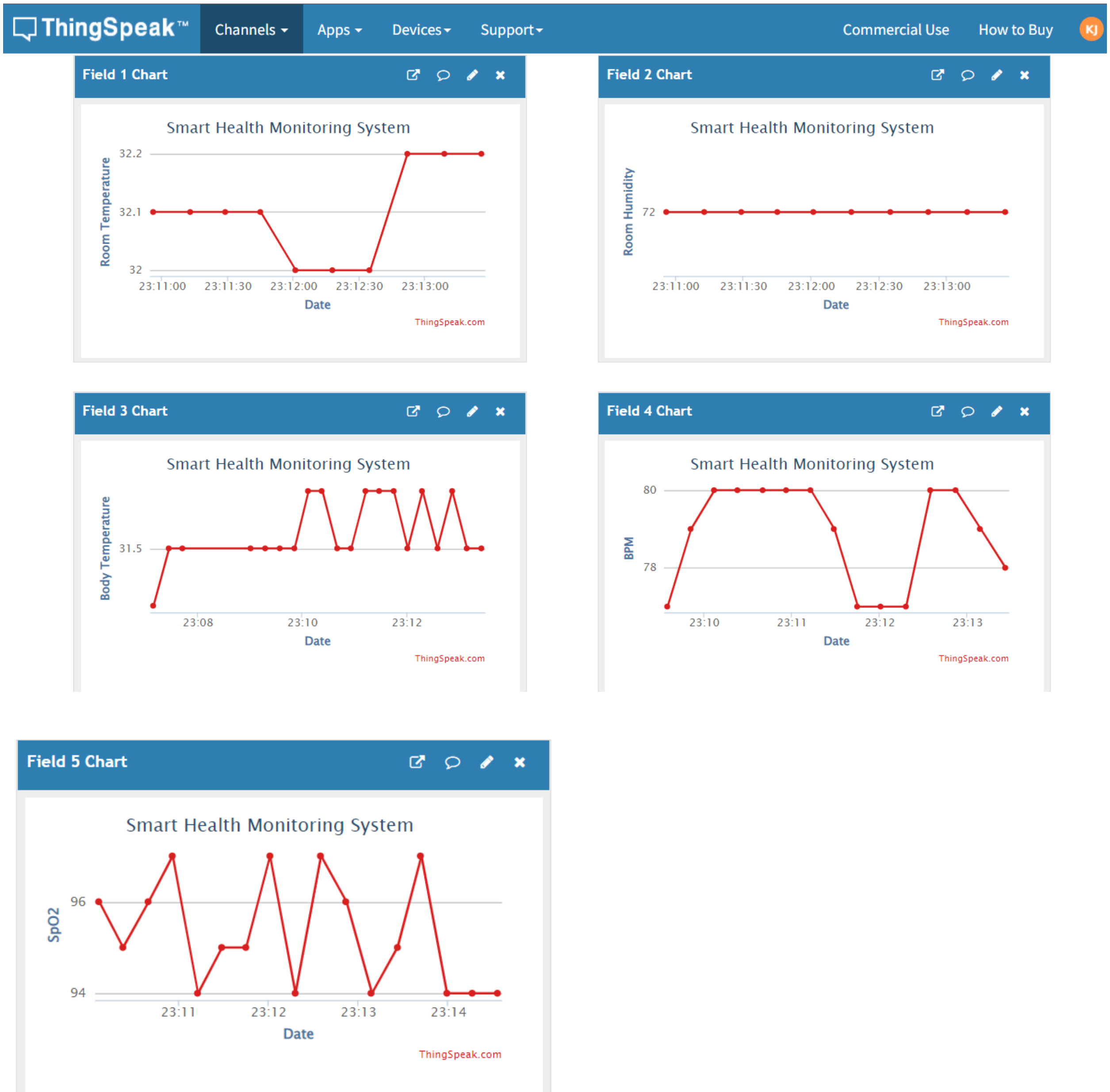


Fig 15:Thingspeak Dashboard

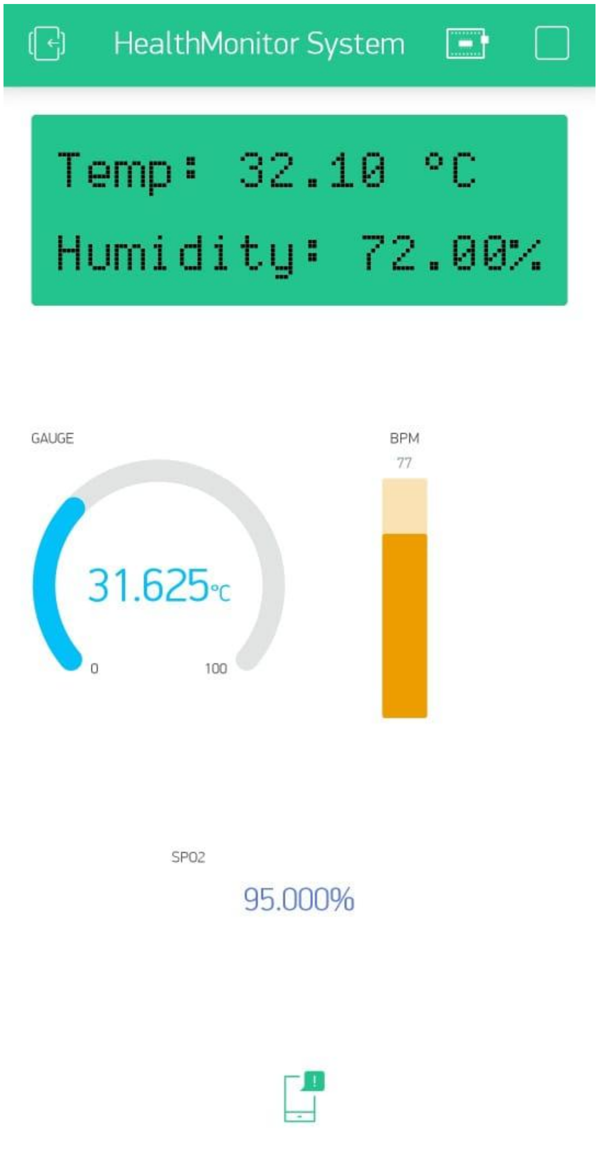
MATLAB Analysis in ThingSpeak

Output

```
Maximum BPM for the past 24 hours is =

    80


Minimum BPM for the past 24 hours is =

    77

The patient has higher heart rate. So, he/she needs attention
Average Humidity =

   72.0857
```

Fig 16 : Output of the MATLAB Analysis

Blynk Application



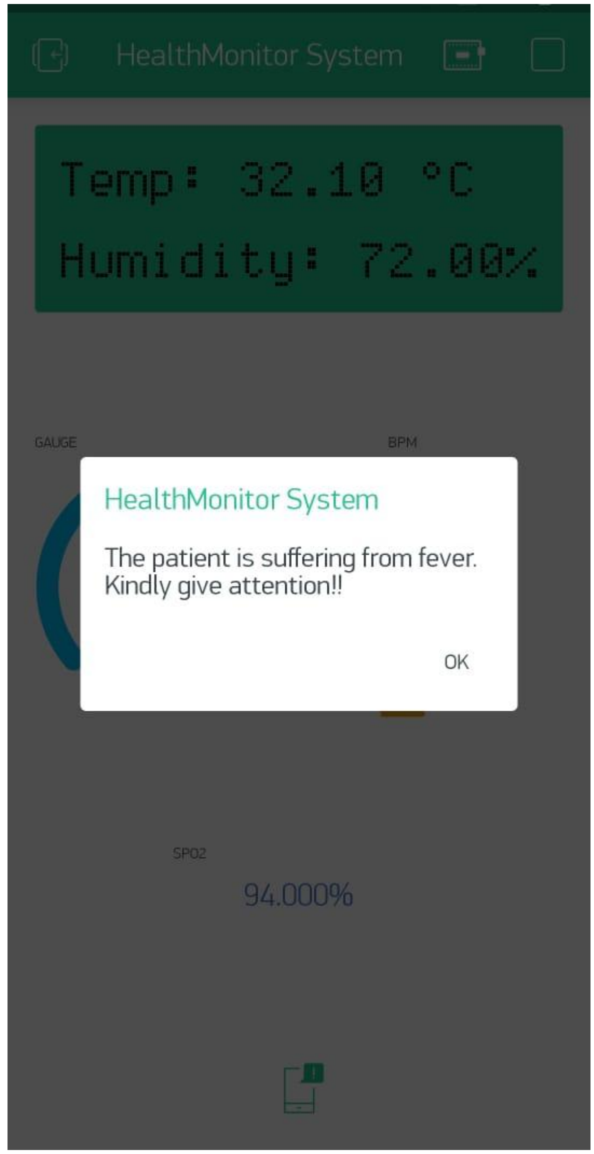


Fig 17: Blynk data Visualization **Fig 18: Alert Notification**

CHAPTER-5

CONCLUSION AND FUTURE ENHANCEMENT

5.1 CONCLUSION AND INFERENCE

The technological advancements have increased exponentially over the past 15 years. This is the period of automation in the history of the human race. The health of people has been largely affected due to improper eating habits and change in lifestyle. The Smart Health Monitoring System gives a great support in building a healthy society with the help of its salient features. The sensors attached in the society gives us accurate information about the person’s health for every instance. When the health values go beyond normal level the person is being alerted immediately through the BLYNK app and also is stored in a cloud. The person’s health status can be monitored by doctors and guardians. Also the system provides us the patient’s room temperature and also the humidity of the room so that the doctors could analyze the patient’s surroundings as well. The cloud saves the patient's previously recorded details so that a complete record of the patient's health history is stored which makes it easier for the doctor’s to analyze. This system brings a great boost to the health and well being of the society.

5.2 FUTURE WORKS

There are a lot of technological advancements that can be added to our project.

Some examples are like

- If we have high body temperature then it indicates to the patient about it and prescribes the medicine which has to be taken for the particular patient.
- If any particular data body temperature goes way beyond the threshold value the doctor who has been registered for the patient immediately gets notified by the call and message.
- Also a live camera feed can be set to the system for the doctor to check upon the patient.
- If the SpO2 level decreases, an oxygen cylinder can be opened and the oxygen can be supplied through the oxygen mask to the patients.

REFERENCES

1. “E-health monitoring system”. Authors: *Aleksandar Kotevski, Natasa Koceska and Saso Koceski.*
2. “Iot Patient Health Monitoring System”. Authors: Shola Usha Rani, Antony Ignatious, Bhava Vyasa Hari, Balavishnu V J.
3. "A Review on IOT Based Health Care Monitoring System". Authors *Pravin Kshirsagar, Akshay Pote, K.K.Paliwal,VaibhavHendre,Pranav Chippalkatti and Nikhil Dhabekar*.
4. "A Review on Health Monitoring Systems using IoT". Authors: *Harshitha Bhat Nishmitha Shetty and Ankitha Shetty.*
5. "A SURVEY ON HEALTH CARE MONITORING SYSTEM USING IOT". Authors : *Sasi Priya Saminathan, K.Geetha.*
6. “Personalized Health Monitoring System of Elderly Wellness at the Community Level in Hong Kong.” AUTHOR: *LISHA YU, WAI MAN CHAN2, YANG ZHAO , AND KWOK-LEUNG TSUI.* IEEE journal.
7. “Wireless, Multipurpose In-Home Health Monitoring Platform: Two Case Trials”. AUTHOR: *Sakari Junnila, Harri Kailanto, Juho Merilahti, Antti-Matti Vainio, Antti Vehkaoja, Mari Zakrzewski, and Jari Hyttinen.* IEEE journal.
8. “Three Tier Architecture for IoT Driven Health Monitoring System Using Raspberry Pi ” . IEEE journal Author: *Neel Kamal, Prasun Ghosal* .
9. “Smart Health Monitoring System through IOT” AUTHOR: *Ananth S, Sathya P and Madhan Mohan P* . IEEE journal
10. “COVID-SAFE: An IoT-Based System for Automated Health Monitoring and Surveillance in Post-Pandemic Life ” AUTHOR: *SEYED SHAHIM VEDAEI , AMIR FOTOVVAT , MOHAMMAD REZA MOHEBBIAN , GAZI*

M. E. RAHMAN , *KHAN A.WAHID, PAUL BABYN , HAMID REZA MARATEB , MARJAN MANSOURIAN , AND RAMIN SAMI.* IEEE journal.

11. "A Smart Health System: Monitoring Comatose Patient's Physiological Conditions Remotely ". AUTHOR: *Tati Erlina, Marreza Rifa Saputra & Rahmi Eka Putri*. IEEE journal
12. "Smart Health Care Monitoring System based on Internet of Things (IOT)". AUTHOR: *V. Baby Shalini.* IEEE journal.
13. "IoT Augmented Health Monitoring System" .IEEE journal AUTHOR: *A. Nishitha Reddy , Achsah Mary Marks , S.R.S Prabaharan, S.Muthulakshmi.*
14. "Health Monitoring System Using Internet of Things". AUTHOR: *Nurassyl Zholdas, Octavian Postolache, Madina Mansurova*. IEEE journal.
15. "Internet of things (IoT) based smart health care system". AUTHOR *Vikas Vippalapalli & Snigdha Ananthula.* IEEE journal.

BIODATA

NAME : KEERTHIVASAN J

MOBILE NUMBER : 8778205789

EMAIL ID: keerthivasan.j2019@vitstudent.ac.in

PERMANENT ADDRESS : B1-6,Abhinayam Phase-3,Sriram Nagar,Nolambur, Chennai-600095

NAME: MOTISH RAJ P

MOBILE NUMBER: 8248006989

Email Id: motishraj.p2019@vitstudent.ac.in

Permanent Address: D1,Metro Castle, Jayam street, Mogappair,Chennai-600037.

NAME : SAI SARAN S

MOBILE NUMBER : 8825659407

EMAIL ID : saisaran.2019@vitstudent.ac.in

PERMANENT ADDRESS: 2/303, Jeeva street, M A Nagar, Redhills, Chennai - 600 052.