PROJECT TITLE: **PATIENT MONITORING ALERT SYSTEM AND INTELLIGENT TRAFFIC SIGNALLING PRIORITY SYSTEM FOR AMBULANCES USING GSM TECHNOLOGY.**

TEAM MEMBERS:

MANCHIMSETTI VYSHNAVI – 22BCE9484

PADMARAJU INDHU SREE – 22BCE9633

NALLAGATLA JAHNAVI – 22BCE9634

T REVANTH – 22BCE9026

BOBBURI HEMANTH BABU – 22BCE7018

BUSI SAI DHANUSH – 22BCE7594

UNDER THE GUIDANCE OF: PROF. SHEELA JAYACHANDRAN

SUMMARY:

Healthcare is given the extreme importance now a- days by each country with the advent of the novel corona virus. So in this aspect, an IoT based health monitoring system is the best solution for such an epidemic. Internet of Things (IoT) is the new revolution of internet which is the growing research area especially in the health care. With the increase in use of wearable sensors and the smart phones, these remote health care monitoring has evolved in such a pace.

IoT monitoring of health helps in preventing the spread of disease as well as to get a proper diagnosis of the state of health, even if the doctor is at far distance. In this project a portable physiological checking framework is displayed, which can constantly screen the patient's heartbeat, temperature and other basic parameters of the room. We proposed a nonstop checking and control instrument to screen the patient condition and store the patient information's in server utilizing Wi-Fi Module based remote correspondence.

A remote health monitoring system using IoT is proposed where the authorized personal can access these data stored using any IoT platform and based on these values received, the diseases are diagnosed by the doctors from a distance ,

In addition to these we are adding the priority based traffic signalling system for emergency vehicles such as ambulances. In some cases the patients who are under any heart arrest or undergone any accidents need to be transferred to hospitals in much possible less time , but the traffic in the major cities is standing as the major burden. In such situations this ambulance priority system helps the people to some extent giving the information at the traffic signals that from where the ambulance is coming and to which direction it will be going.

The miniature processing units and communication networks combinely work together to make this system a success. An android and cloudbased control system using the GSM module is an effective and price competitive solution that can solve this problem. The system comprises of 5 stages which are android mobile device, GSM module, MQTT (IoT) for Arduino IDE, Arduino Uno microcontroller and traffic signals. The developed system has allowed the android mobile device (emergency vehicle) to override the traditional operation of a stoplight.

The main purpose of this device is to allow the ambulance to reach a specific location without making it stop somewhere before it reaches the destination.

INTRODUCTION:

- It is essential to stay healthy to lead a happy and comforting life. According to the world health organization having good health is a fundamental right for every individual, and unfortunately, it is a global problem to confront in the future. We use the concept of the internet of things, which is well heard and resourcefully put into application these days. In hospitals there are provisions for continuous monitoring of patients. Their ECGs, heartbeat, are continuously monitored. There is no provision to check the parameters when they return to home .

- Also there is no need for a number of medical personnel for accompanying with patients to be physically present to check the health condition of the coma affected person, The goal is to remotely monitor health condition of any individual .In the recent days, the health care system is advancing to provide better facilities to the mankind there is a major role played by technology in this marathon of advancement lifestyle always makes things easier for oneself. Integration of technology with the patient health monitoring, facilitates the health care providers and the patients as well. There are multiple health parameters which requires continuous monitoring in the hospital setup or at home for a chronically ill patient.

- A very rapid population growth in cities has resulted in tremendous road traffic within the city.In addition, in recent times the number of deaths due to delays in the arrival of emerg ency vehicle has risen to greater extent. Hence emergency services such a s ambulances and fire engines must be on time to avoid loss of human life .In the current traffic situation, therefore, helping an emergency vehicle move out of traffic congestion is very much important. To solve the problems given above. In this paper, we have come up with the ‘Smart Ambulance and traffic controlling system’. The traffic signalling device inserted helps to give fast response navigates ambulances to find the shortest possible paths till their destination, but also presents a counter measure to get rid the problem of the traffic light system when it is hacked during its operation.

BACKGROUND:

Going back to the past 3 years, we suffered a lot at the hardest time of covid-19, as as many doctors were not available and even they could not give proper treatment due to this communicable virus, in case of emergency situations a doctor cannot be with the same patient all the time as he have to monitor on numerous patients, in those situations we thought of this device where a sensor is attached to the patients body, where we can monitor their minimum aspects such as temperature, oxygen levels which can notify the doctors immediately when there is a fluctuation in any of the aspect.

PROBLEM DEFINITION:

In hospitals now a days, the ratio of number of doctors is less when compared to the ratio of incoming and hospitalized patients and the doctors cannot manage all the patients at the single time.

So, where a large number of patients whose physical conditions have to be monitored frequently as a part of diagnostic procedure the need for a cost effective and fast responding alert mechanism is needed.

Traffic congestion worldwide has led to loss of human lives due to failure in transporting accident victims, critical patients, medical, equipments and medicines on time.

Advantages:

1.Improved Chronic Condition Management

Improved chronic condition management refers to the enhancement or optimization of strategies, treatments, and approaches aimed at caring for individuals with long-term health conditions. Advanced strategies aiding in the effective handling and control of persistent health issues

.Reduced Emergent Situations and Readmissions

Healthcare context where improved care or interventions aim to lessen unexpected health crises or hospital readmissions. It suggests strategies or systems in place to mitigate sudden health issues or prevent patients from needing to return to the hospital shortly after discharge, emphasizing the importance of proactive care and interventions to maintain health stability.

3.Increased Revenue Streams

Increased revenue streams in patient monitoring systems can be achieved through various means such as offering additional services like remote monitoring, providing real-time data analytics, and partnering with healthcare organizations for expanded market reach. These strategies can help generate more revenue while improving patient care.

4.Reduced Burden on Healthcare Systems

Reduced burden on healthcare systems in patient monitoring systems can be achieved by enabling remote monitoring, which reduces the need for frequent hospital visits, optimizing resource allocation, and improving overall efficiency in healthcare delivery. This helps healthcare systems allocate their resources effectively and provide timely care to patients. .

5.Improved Patient Outcomes

Improved patient outcomes in patient monitoring systems are achieved through timely detection of health issues, personalized treatment plans, and continuous monitoring to track progress. This helps healthcare providers deliver proactive care, leading to better patient outcomes and overall well-being

6.Better Quality of Care

With patient monitoring systems, healthcare providers can deliver better quality of care by providing real-time data for accurate diagnosis, enabling timely interventions, and promoting personalized treatment plans. This leads to improved patient satisfaction and overall healthcare outcomes.

7.Increased Education

In patient monitoring systems, increased education can be achieved through interactive platforms that provide educational resources, personalized health coaching, and regular updates on the latest advancements in healthcare. This empowers patients to actively participate in their own care and make informed decisions about their health.

8.Increased Patient Accountability

In patient monitoring systems, increased patient accountability is fostered by empowering patients to actively engage in their own health management, encouraging adherence to treatment plans, and promoting self-monitoring of vital signs. This leads to improved patient outcomes and a sense of ownership over their health journey.

Disadvantages:

- Is not accessible for everyone. RPM requires good broadband connectivity, which is hard to achieve for small healthcare institutions and rural hospitals. ...
- Patients and doctors' skepticism.
- The need of additional custom healthcare software
- Doubtful reliability.

OBJECTIVES:

Materials required:

- Raspberry pi pico(hardware)
- Max30102 sensor(heart beat,oxygen level)

- Temperature sensor
- 16*2 lcd
- Esp8266 wifi module
- Gsm module
- Jumpwires
- Rf transmitter and rfid reader(for traffic signalling)

PROCEDURE:

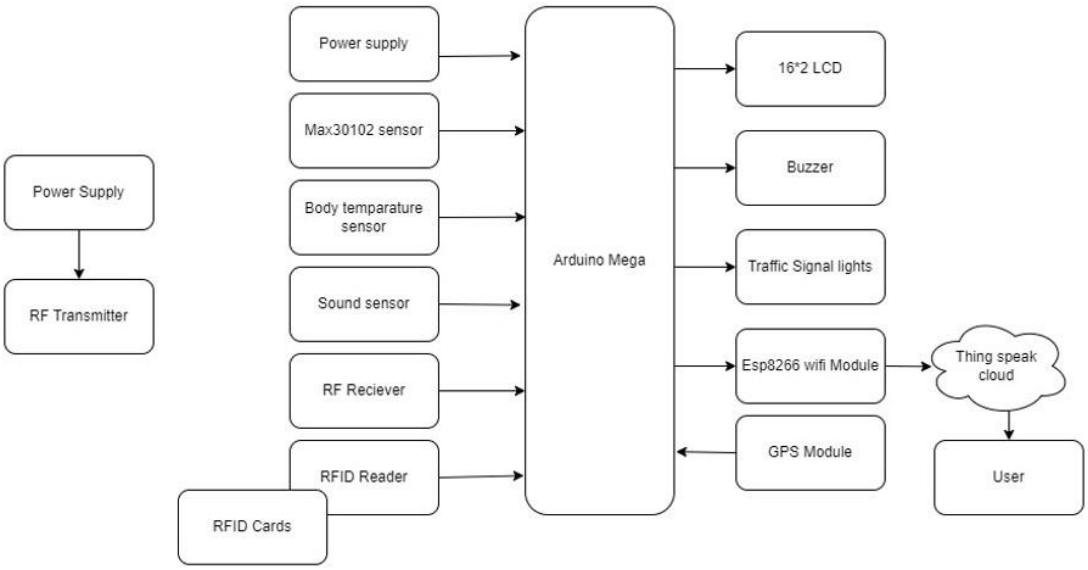
With tons of new healthcare technology start-ups, IoT is rapidly revolutionizing the healthcare industry. In this project, we have designed the IoT Based Patient Health Monitoring System using ESP8266 & Arduino. The IoT platform used in this project is ThingSpeak. ThingSpeak is an open-source Internet of Things (IoT) application and API to store and retrieve data from things using the HTTP protocol over the Internet or via a Local Area Network. This IoT device could read the pulse rate and measure the surrounding temperature. It continuously monitors the pulse rate and surrounding temperature and updates them to an IoT platform.

The Arduino Sketch running over the device implements the various functionalities of the project like reading sensor data, converting them into strings, passing them to the IoT platform, and displaying measured pulse rate and temperature on character LCD.

You can check previous projects related to Pulse Sensor if you are a beginner:

1. Heartbeat/Pulse/BPM Rate Monitor using Arduino & Pulse Sensor.
2. Pulse Rate Monitoring over Internet using ThingSpeak & ESP8266.
3. ECG Display using Pulse Sensor with OLED & Arduino.

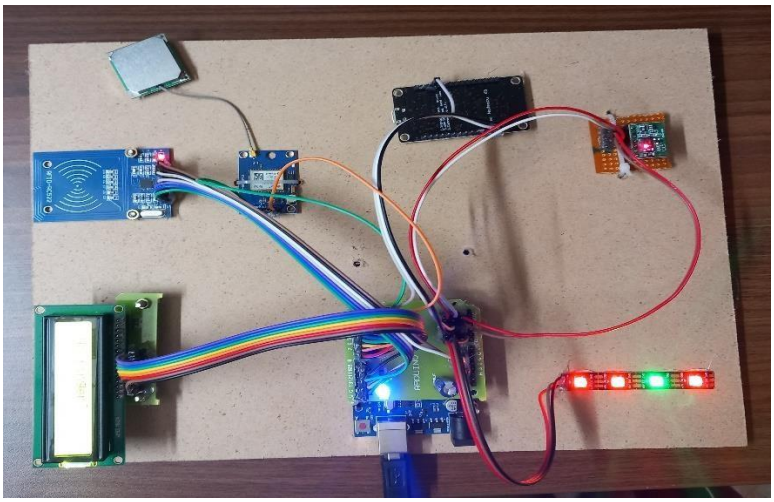
Block diagram:



RESULT AND DISCUSSION:

- When using the health monitoring system , when the finger is placed on the sensor then the objectives such as oxygen levels , heart beat and temperature are recorded.
- When using the traffic signalling system, when the sensor connecting the ambulances are placed then , it sends the signals to traffic signals which direction the ambulances need to go.

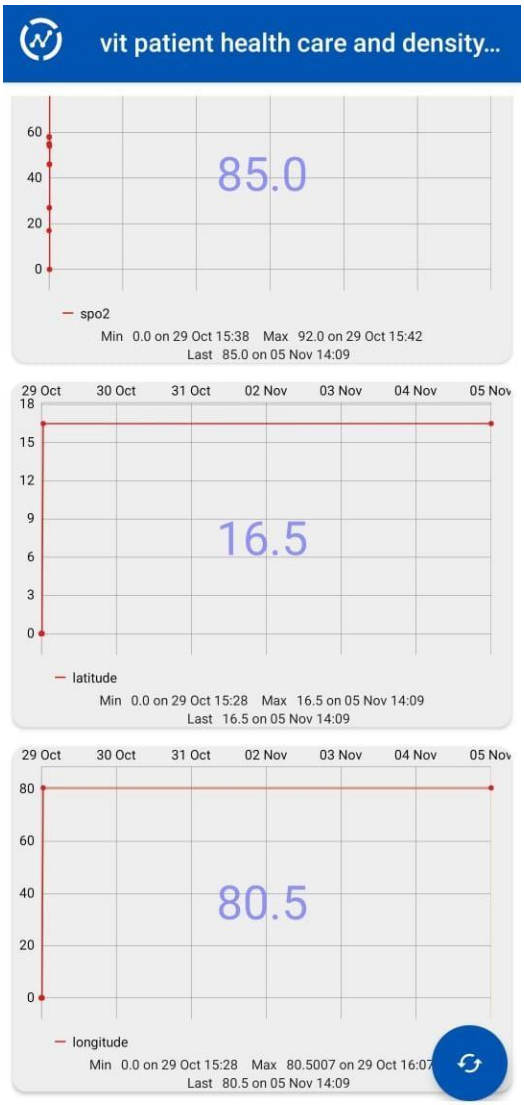
Image of final prototype:-

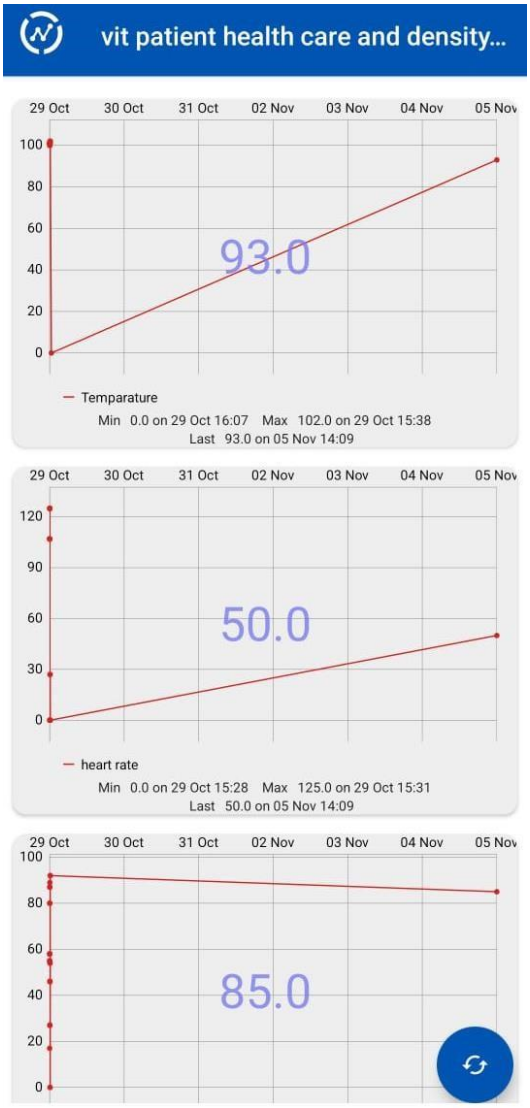


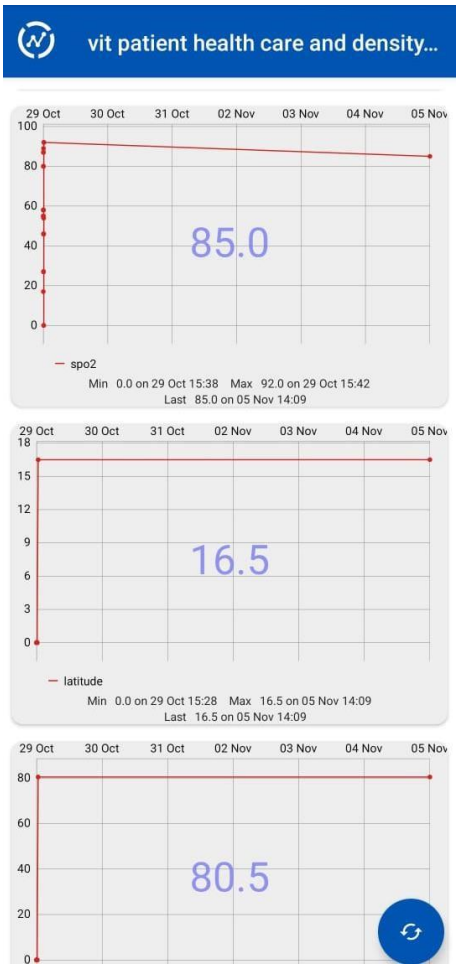
Output of patient monitoring:-

Output of traffic signalling:-

Graphs:


vit patient health care and density...
85.0
spo2
Min 0.0 on 29 Oct 15:38 Max 92.0 on 29 Oct 15:42
Last 85.0 on 05 Nov 14:09
16.5
latitude
Min 0.0 on 29 Oct 15:28 Max 16.5 on 05 Nov 14:09
Last 16.5 on 05 Nov 14:09
80.5
longitude
Min 0.0 on 29 Oct 15:28 Max 80.5007 on 29 Oct 16:07
Last 80.5 on 05 Nov 14:09


vit patient health care and density...
93.0
Temparature
Min 0.0 on 29 Oct 16:07 Max 102.0 on 29 Oct 15:38
Last 93.0 on 05 Nov 14:09
50.0
heart rate
Min 0.0 on 29 Oct 15:28 Max 125.0 on 29 Oct 15:31
Last 50.0 on 05 Nov 14:09
85.0



Conclusion and future scope:

- The Internet of Things is considered now as one of the feasible solutions for any remote value tracking especially in the field of health monitoring. It facilitates that the individual prosperity parameter data is secured inside the cloud, stays in the hospital are reduced for conventional routine examinations and most important that the health can be monitored and disease diagnosed by any doctor at any distance. In this paper, an IoT based health monitoring system was developed. The system monitored body temperature, pulse rate and room humidity and temperature using sensors, which are also displayed on a LCD. These sensor values are then sent to a medical server using wireless communication. These data are then received in an authorized personals smart phone with IoT platform. With the values received the doctor then diagnose the disease and the state of health of the patient. The gsm module which is used can easily help to give the priority system for the way for ambulances.

- In future, we strongly believe that in this era of machinazation and ai , this project would greatly help doctors as well as the patients in providing and recieving best treatment. And also in future
  we will help coma patients to monitor their movement, and some more features to give directions to ambulances.

References:

These are the references that I we refered

- Aieshwarya. B. Chavan Patil "An IoT Based Health Care and Patient Monitoring System to Predict Medical Treatment using Data Mining Techniques: Survey", International Journal of Advanced Research in Computer and Communication Engineering(IJARCCE), ISO 3297:2007 Certified Vol. 6, Issue 3, March 2017

- [2] Tamilselvi V, Sribalaji S, Vigneshwaran P, Vinu P, J.GeethaRamani, "IoT Based Health Monitoring System", 2020 6th International Conference on Advanced Computing & Communication Systems (ICACCS)

- [3] Taniya Shirely Stalin, Abey Abraham, "IOT BASED HEALTH MONITORING SYSTEM AND TELEMEDICINE", International Research Journal of Engineering and Technology (IRJET), Volume: 07 Issue: 03 | Mar 2020

- [4] Ufoaroh S.U , Oranugo C.O, Uchechukwu M.E , HEARTBEAT MONITORING AND ALERT SYSTEM USING GSM TECHNOLOGY , International Journal of Engineering Research and General Science Volume 3, Issue 4, July-August, 2015 ISSN 2091-2730

Codes in Appendix:

This system is covertable based on where it has its importance such as in hospitals it can be used to monitor the patients condition where as in the other way its can be used by ambulances to give the way giving them the priority.

Coding for health monitoring:

```
#include <Wire.h>
#include "MAX30105.h"
#include "heartRate.h"
MAX30105 particleSensor; const int rs =2, en =3,
d4 =4, d5 = 5, d6 = 6, d7 =7;
#include <LiquidCrystal.h>
int buz=13;
```

```
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
int spo2=0,ssum=0,hsum=0;
int cnt=0; float
temperatureF;
const byte RATE_SIZE = 4; //Increase this for more averaging. 4 is good.
byte rates[RATE_SIZE]; //Array of heart rates byte
rateSpot = 0;
long lastBeat = 0; //Time at which the last beat
occurred float beatsPerMinute; int beatAvg; int kk=0;
int ab=0; void setup(){   Serial.begin(115200);
lcd.begin(16, 2);
  pinMode(buz,OUTPUT);
digitalWrite(buz,0);
digitalWrite(buz,1);   lcd.print("
WELCOME");
  if (!particleSensor.begin(Wire, I2C_SPEED_FAST)) //Use default
I2C port, 400kHz speed {
    Serial.println("MAX30105 was not found. Please check wiring/power.
");
   while (1)   }    particleSensor.setup(); //Configure sensor with
default  settings       particleSensor.setPulseAmplitudeRed(0x0A);
//Turn Red LED to low to indicate sensor is running
  particleSensor.setPulseAmplitudeGreen(0); //Turn off Green LED
particleSensor.enableDIETEMPRDY();   delay(2000);   wifi_init()
}
void loop() {
  long irValue = particleSensor.getIR();
if (checkForBeat(irValue) == true)
  {
   //We sensed a beat! long
delta = millis() - lastBeat;
lastBeat = millis();
   beatsPerMinute = 60 / (delta / 1000.0);   if
(beatsPerMinute < 255 && beatsPerMinute > 20)
   {
     rates[rateSpot++] = (byte)beatsPerMinute; //Store this reading in
the array       rateSpot %= RATE_SIZE;
//Wrap variable    //Take average of readings
beatAvg = 0;       for (byte x = 0 ; x <
RATE_SIZE ; x++)         beatAvg += rates[x];
beatAvg /= RATE_SIZE;
   }
  }
 //Serial.print("IR=");
  //Serial.print(irValue);
  //Serial.print(", BPM=");  //
Serial.print(beatsPerMinute);
  float temperatureF = particleSensor.readTemperatureF()-6;
//Because I am a bad global citizen
beatAvg=beatsPerMinute;
if(beatAvg<40)   spo2=0;   else
if(beatAvg>140)
spo2=map(beatAvg,140,255,80,50);
else if(beatAvg>=40 && beatAvg<60)
spo2=map(beatAvg,40,60,70,100);
  else
  spo2=map(beatAvg,60,140,100,80);
ssum=ssum+spo2;
hsum=hsum+beatAvg;   cnt=cnt+1;
if(cnt==10)
```

```
  {     cnt=0;
spo2=ssum/10;
beatAvg=hsum/10;
ssum=0; hsum=0;
if (irValue < 50000)
  {      lcd.clear();
lcd.print(" No finger?");
Serial.println(" No finger?");
beatAvg=0;
  spo2=0;
temperatureF=0;
  }
else
  {
  if(beatAvg>40)
  {    lcd.clear();         lcd.print("H:"+ String(beatAvg) + "
S:"+ String(spo2));         lcd.setCursor(0,1);
lcd.print("t:"+ String(temperatureF));
        Serial.println("H:");
        Serial.println(beatAvg);
        Serial.println("S:");
        Serial.println(spo2);
        Serial.println("T:");           Serial.println(temperatureF);
upload_iot(temperatureF,beatAvg,spo2);           if((spo2>30 &&
spo2<70) || beatAvg>90 || temperatureF>100)
        {
          digitalWrite(buz,1);
delay(1000);
digitalWrite(buz,0);
ab=ab+1;           if(ab>4
&& kk==0)
          {
kk=1;
          }
}
else
        {
          digitalWrite(buz,0);
ab=0;           kk=0;
        }   }   else   {
lcd.clear();
lcd.print("Reading..");
beatAvg=0;   spo2=0;
temperatureF=0;
  }}  }  }
void wifi_init()  {
Serial.println("AT+RST");   delay(2000);
  Serial.println("AT+CWMODE=1");
delay(1000);
  Serial.print("AT+CWJAP=");
  Serial.write("");
  Serial.print("project"); // ssid/user name
  Serial.write("");
  Serial.write(',');
  Serial.write("");
  Serial.print("12345678"); //password
  Serial.write("");
Serial.println();
delay(1000); }
void upload_iot(int x,int y,int z){
```

```
  String cmd = "AT+CIPSTART=\"TCP\",\"";
cmd += "184.106.153.149"; // api.thingspeak.com
cmd += "\",80";   Serial.println(cmd);
delay(1500);  String getStr ="GET
/update?api_key=I4DK514VX5E7D8UQ&field1=";
  getStr += String(x);
getStr +="&field2=";
getStr += String(y);
getStr +="&field3=";
getStr += String(z);
getStr +="&field4=";
getStr += String(16.4963);
getStr +="&field5=";
getStr += String(80.5007);
getStr += "\r\n\r\n";
  cmd = "AT+CIPSEND=";
  cmd += String(getStr.length());
Serial.println(cmd);
delay(1500);
Serial.println(getStr);
delay(1500);
 }
```

Code for ambulance priority:

```
#include <SPI.h>
#include <MFRC522.h>
//#include <Adafruit_GFX.h>
//#include <Adafruit_SSD1306.h>
#include <SPI.h> #include
<LiquidCrystal.h>
#include <Adafruit_NeoPixel.h>
#ifdef _AVR_
 #include <avr/power.h> // Required for 16 MHz Adafruit Trinket
#endif
#define LED_PIN A1
#define LED_COUNT 4
Adafruit_NeoPixel strip(LED_COUNT, LED_PIN, NEO_GRB + NEO_KHZ800);


#define SS_PIN 10
#define RST_PIN 9
MFRC522 mfrc522(SS_PIN, RST_PIN);   // Create MFRC522 instance.
int buz =8; int
es=A3; int
ws=A3; int
ns=A3; int
ss=A3; const
int rs =2, en
=3, d4 =4, d5
=5, d6 =6, d7
=7;
LiquidCrystal
lcd(rs, en, d4,
d5, d6, d7);


int cnt=0,ecnt=0,wcnt=0,ncnt=0,scnt=0,sts=0,i=0; void
setup()  {
```

```
  Serial.begin(9600);   // Initiate a serial communication
 SPI.begin();     // Initiate  SPI bus  mfrc522.PCD_Init();
 // Initiate MFRC522
  Serial.println("Approximate your card to the reader...");
Serial.println();
  pinMode(es,INPUT);
pinMode(ws,INPUT);
pinMode(ns,INPUT);   pinMode(ss,INPUT);
pinMode(buz,OUTPUT);
  lcd.begin(16,2);
lcd.print("  WELCOME");
   strip.begin();           // INITIALIZE NeoPixel strip object (REQUIRED) strip.setBrightness(100);
    strip.setPixelColor(0, strip.Color(255,   0,   0));
strip.setPixelColor(1, strip.Color(255,   0,   0));    strip.setPixelColor(2,
strip.Color(255,   0,   0));    strip.setPixelColor(3, strip.Color(255,   0,
0));
  strip.show();            // Turn OFF all pixels ASAP
  } void
loop()
{            lcd.clear(); cnt=5+ecnt;
lcd.print("EAST :"+ String(cnt));
for(i=cnt;i>0;i--)
 {   lcd.setCursor(6,0);   lcd.print("   ");
lcd.setCursor(6,0);   lcd.print(i);
strip.setPixelColor(0, strip.Color(0,     255,   0));
strip.setPixelColor(1, strip.Color(255,   0,   0));
strip.setPixelColor(2, strip.Color(255,   0,   0));
strip.setPixelColor(3, strip.Color(255,   0,   0));
strip.show();    scan_rfid();
    for(int k=0;k<10;k++)
   {
delay(100);
scan();
   }
 }  cnt=5+ncnt; lcd.clear();
lcd.print("NORTH:"+ String(cnt));
  for(i=cnt;i>0;i--)
 {     lcd.setCursor(6,0);   lcd.print("   ");
lcd.setCursor(6,0);   lcd.print(i);
strip.setPixelColor(0, strip.Color(255,   0,   0));
strip.setPixelColor(1, strip.Color(0,   255,   0));
strip.setPixelColor(2, strip.Color(255,   0,   0));
strip.setPixelColor(3, strip.Color(255,   0,   0));
strip.show();
   scan_rfid();
scan();
   for(int k=0;k<10;k++)
   {
delay(100);
scan();     }


 }
 cnt=5+wcnt;
lcd.clear();
lcd.print("WEST :"+ String(cnt));


 for(i=cnt;i>0;i--)  {     lcd.setCursor(6,0);
lcd.print("   ");   lcd.setCursor(6,0);   lcd.print(i);
strip.setPixelColor(0, strip.Color(255,   0,   0));
strip.setPixelColor(1, strip.Color(255,   0,   0));
```

```
strip.setPixelColor(2, strip.Color(0,   255,   0));
strip.setPixelColor(3, strip.Color(255,   0,   0));
strip.show();    scan_rfid();
   for(int k=0;k<10;k++)
   {
delay(100);
scan();     }
 }
cnt=5+scnt;
lcd.clear();
lcd.print("SOUTH:"+ String(cnt));


 for(i=cnt;i>0;i--)  {     lcd.setCursor(6,0);
lcd.print("    ");   lcd.setCursor(6,0);   lcd.print(i);
strip.setPixelColor(0, strip.Color(255,   0,   0));
strip.setPixelColor(1, strip.Color(255,   0,   0));
strip.setPixelColor(2, strip.Color(255,   0,   0));
strip.setPixelColor(3, strip.Color(0,   255,   0));
strip.show();    scan_rfid();
      for(int k=0;k<10;k++)
   {
delay(100);
scan();     }
 }




}


 void scan_rfid()
 {     if ( !
mfrc522.PICC_IsNewCardPresent())
  {
return;   }


  if ( ! mfrc522.PICC_ReadCardSerial())
  {
return;
  }


  Serial.print("UID tag :");   String
content= "";   byte letter;   for (byte i = 0;
i < mfrc522.uid.size; i++)
  {
     Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");
Serial.print(mfrc522.uid.uidByte[i], HEX);      content.concat(String(mfrc522.uid.uidByte[i]
< 0x10 ? " 0" : " "));      content.concat(String(mfrc522.uid.uidByte[i], HEX));
  }
  Serial.println();
Serial.print("Message : ");
content.toUpperCase();   sts=0;
  //////////////////////////
  if (content.substring(1) == "21 5B 1D 85") //change here the UID of the card/cards that you want to give access
  {
    digitalWrite(buz,1);
```

```
   Serial.println("Ambulance..");
Serial.println();     sts=1;
delay(1000);
   digitalWrite(buz,0);


 cnt=5+ecnt;
lcd.clear();
lcd.print("EAST :"+ String(cnt));
lcd.setCursor(11,0);
lcd.print("Amb");  for(i=cnt;i>0;i--)
 {     lcd.setCursor(6,0);   lcd.print("   ");
lcd.setCursor(6,0);   lcd.print(i);
strip.setPixelColor(0, strip.Color(0,     255,   0));
strip.setPixelColor(1, strip.Color(255,   0,   0));
strip.setPixelColor(2, strip.Color(255,   0,   0));
strip.setPixelColor(3, strip.Color(255,   0,   0));
strip.show();    delay(1000);
 }


  }


  if (content.substring(1) == "69 95 BF 15") //change here the UID of the card/cards that you want to give access
  {
   digitalWrite(buz,1);


   Serial.println("Ambulance..");
Serial.println();     sts=1;
delay(1000);
   digitalWrite(buz,0);


 cnt=5+ecnt;
lcd.clear();
lcd.print("West :"+
String(cnt));
lcd.setCursor(11,0);
lcd.print("Amb");
for(i=cnt;i>0;i--)
 {     lcd.setCursor(6,0);   lcd.print("   ");
lcd.setCursor(6,0);   lcd.print(i);
strip.setPixelColor(0, strip.Color(255,     0,   0));
strip.setPixelColor(1, strip.Color(0,   255,   0));
strip.setPixelColor(2, strip.Color(255,   0,   0));
strip.setPixelColor(3, strip.Color(255,   0,   0));
strip.show();    delay(1000);
 }


  }


   if (content.substring(1) == "0A 45 2B 2E") //change here the UID of the card/cards that you want to give access
  {
   digitalWrite(buz,1);


   Serial.println("Ambulance..");
Serial.println();     sts=1;
delay(1000);
   digitalWrite(buz,0);
```

```
 cnt=5+ecnt;
lcd.clear();
lcd.print("North :"+ String(cnt));
lcd.setCursor(11,0);
lcd.print("Amb");  for(i=cnt;i>0;i--)
 {    lcd.setCursor(6,0);   lcd.print("   ");
lcd.setCursor(6,0);   lcd.print(i);
strip.setPixelColor(0, strip.Color(255,    0,   0));
strip.setPixelColor(1, strip.Color(255,   0,   0));
strip.setPixelColor(2, strip.Color(0,   255,   0));
strip.setPixelColor(3, strip.Color(255,   0,   0));
strip.show();    delay(1000);
 }


  }

   if (content.substring(1) == "0B 3F 1E 85") //change here the UID of the card/cards that you want to give access
  {
   digitalWrite(buz,1);


   Serial.println("Ambulance..");
Serial.println();     sts=1;
delay(1000);
   digitalWrite(buz,0);


 cnt=5+ecnt;
lcd.clear();
lcd.print("South :"+ String(cnt));
lcd.setCursor(11,0);     lcd.print("Amb");
for(i=cnt;i>0;i--)
 {    lcd.setCursor(6,0);   lcd.print("   ");
lcd.setCursor(6,0);   lcd.print(i);
strip.setPixelColor(0, strip.Color(255,   0,   0));
strip.setPixelColor(1, strip.Color(255,   0,   0));
strip.setPixelColor(2, strip.Color(255,   0,   0));
strip.setPixelColor(3, strip.Color(0,   255,   0));
strip.show();    delay(1000);
 }


  }

//////////////////////
//   if (content.substring(1) == "0A 45 2B 2E") //change here the UID of the card/cards that you want to give access
//  {
//
//     digitalWrite(buz,1);
//
//
//       lcd.setCursor(11,0);
//    lcd.print("VIP");
//    Serial.println("VIP Vehicle..");
//    Serial.println();
//    sts=1;
//    cnt=5+ecnt;
//      delay(1000);
//    digitalWrite(buz,0);
//  lcd.clear();
```

```
//lcd.print("EAST :"+ String(cnt));
//
//      lcd.setCursor(11,0);
//    lcd.print("VIP");
// for(i=cnt;i>0;i--)
// {
//    lcd.setCursor(6,0);
//  lcd.print("   ");
//  lcd.setCursor(6,0);
//  lcd.print(i);
//   strip.setPixelColor(0, strip.Color(0,     255,   0));
//   strip.setPixelColor(1, strip.Color(255,   0,   0));
//   strip.setPixelColor(2, strip.Color(255,   0,   0));
//   strip.setPixelColor(3, strip.Color(255,   0,   0));
//   strip.show();
//   delay(1000);
// }
//
//  }
}


 void scan()  {
if(digitalRead(es)==0)
  {
   ecnt=ecnt+1;
  }

   if(digitalRead(ws)==0)
  {
   wcnt=wcnt+1;
  }

   if(digitalRead(ns)==0)
  {
   ncnt=ncnt+1;
  }

   if(digitalRead(ss)==0)
  {
   scnt=scnt+1;
  }
 }
```