VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAUM 59014

Internet of Things Project Report on

"ACCIDENT ALERT SYSTEM"

By

RIYANCHHI AGRAWAL (1BM17CS076)

SHAKSHI PANDEY(1BM17CS092)

SHEHYAAZ KHAN NAYAZI(1BM17CS094)

SHIVANGI BALODIA(1BM17CS096)

Under the Guidance of

Dr. K. PANIMOZHI

Assistant Professor, Department of CSE

BMS College of Engineering

IoT Application Development carried out at

Department of Computer Science and Engineering

BMS College of Engineering

(Autonomous college under VTU)

P.O. Box No.: 1908, Bull Temple Road, Bangalore-560019

2019-2020

BMS COLLEGE OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Internet of Things project titled "**Accident Alert System**" has been carried out by RIYANCHHI AGRAWAL(1BM17CS076), SHAKSHI PANDEY(1BM17CS092), SHEHYAAZ KHAN NAYAZI(1BM17CS094) and SHIVANGI BALODIA(1BM17CS096**)** during the academic year 2019-2020.

Signature of the guide

PanMozhi K

Assistant Professor

Department of Computer Science and Engineering

BMS College of Engineering, Bangalore

Examiners

Name **Signature**

BMS COLLEGE OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECALARATION

We, RIYANCHHI AGRAWAL(1BM17CS076), SHAKSHI PANDEY(1BM17CS092), SHEHYAA KHAN NAYAZI(1BM17CS094) and SHIVANGI BALODIA(1BM17CS096) students of 5th Semester, B.E, Department of Computer Science and Engineering, BMS College of Engineering Bangalore, hereby declare that, this IoT Application development work entitled "Women Securit System" has been carried out by us under the guidance of **PaniMozhi K**, Assistant Professor, Department of CSE, BMS College of Engineering, Bangalore during the academic semester Aug-Dec 2019.

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

Signature

RIYANCHHI AGRAWAL (1BM17CS076)

SHAKSHI PANDEY (1BM17CS092)

SHEHYAAZ KHAN NAYAZI (1BM17CS094)

SHIVANGI BALODIA (1BM17CS096)

Objective of the project

ABSTRACT :

An accident is an unplanned event that sometimes has unconvenient or undesirable consequences, other times being inconsequential. The main objective of our project is to provide an optimum solution to traffic hazards and road accidents. The application, which is proposed, has access to track location of the vehicle and send alert messages to the nearby hospitals and the scanned phone numbers so that following action to be taken by sending rescue team. This application helps people stuck in accident to reach out for help and also inform to their family and friends.

INTRODUCTION:

According to this project when a vehicle meets with an accident, immediately accelerometer sensor will detect the signal and sends it to Arduino Uno R3. Microcontroller sends the alert message through the GSM MODEM including the location to a rescue team and through WIFI data (latitude, longitude and car number)to cloud. So the police stations and hospitals near by can immediately trace the location through the WIFI after receiving the information. The privacy is very high in the Internet of Things. The sensors are developed in such a way that there is a automatic response without any triggering buttons. This can help people can overcome difficulties. ACCELEROMETERS SENSOR is used in our model to sense the orientation of vehicle to detect the accident. Our Iot application, which is proposed, has access to track location of the vehicle and will send alert messages to the nearby hospitals so that following action can be taken later.

Literature Survey

Sl.No	Name of the Project or Product (Existing)	Commercial or Non-Commercial	Features
1	Cognitio	Non-Commercial	quick emergency communication

Existing Project

Worrying statistics that Team Dhruva's prototype Cognitio, a bike crash detector with an SOS alert system, is aiming to change. Unveiled at the Smart India Hackathon 2019-hardware edition. Cognitio detects a crash, confirms it, and sends an SOS to a centralised command centre, getting emergency medical care swiftly. They presented their design at IIT-Delhi, which was among 19 nodal centres across India, and won a cash prize of Rs 75,000.
With Cognitio mounted on a bike, a crash can be easily ascertained. However, since crash detection does not necessarily signify risk to a biker, a confirmation of the crash is also processed. Emergency communication through an SOS alert is sent immediately. "The entire device is fool-proof. We have tested a wide range of accident scenarios on a bicycle. Crash detection, confirmation, and communication have been functioning wellHowever, the team sought help from their professor and also increased their personal contribution to manage the production expenses. Currently, the device has been designed using readymade modules. With mass production, the estimated cost is Rs 1,000. The team plans to use generic components, and reduce its size to make it cost-effective. The end goal for Cognitio is to be a low-cost and fool-proof device that saves lives, unlike other similar products.
https://yourstory.com/2019/08/team-dhruva-smart-india-hackathon-bike-crash-alert

Proposed Project

According to this project when a vehicle meets with an accident, immediately accelerometer sensor will detect the signal and sends it to Arduino Uno R3. Microcontroller sends the alert message through the GSM MODEM including the location to police control room or a rescue team and through WIFI to cloud. ACCELEROMETERS SENSOR is used in our model to sense the orientation of vehicle to detect the accident.

Hardware Requirements

Hardware requirement

1) **Arduino Uno**
2) **Bread Board**
3) **Accelerometer Sensor(ADXL 345)**
4) **ESP 8266 NodeMCU Wifi Module**
5) **GSM SIM 900 A**
6) **LCD display 16x2**
7) **Jumper Wires**
8) **Buzzer**
9) **Toy Car**
10) **USB cable**

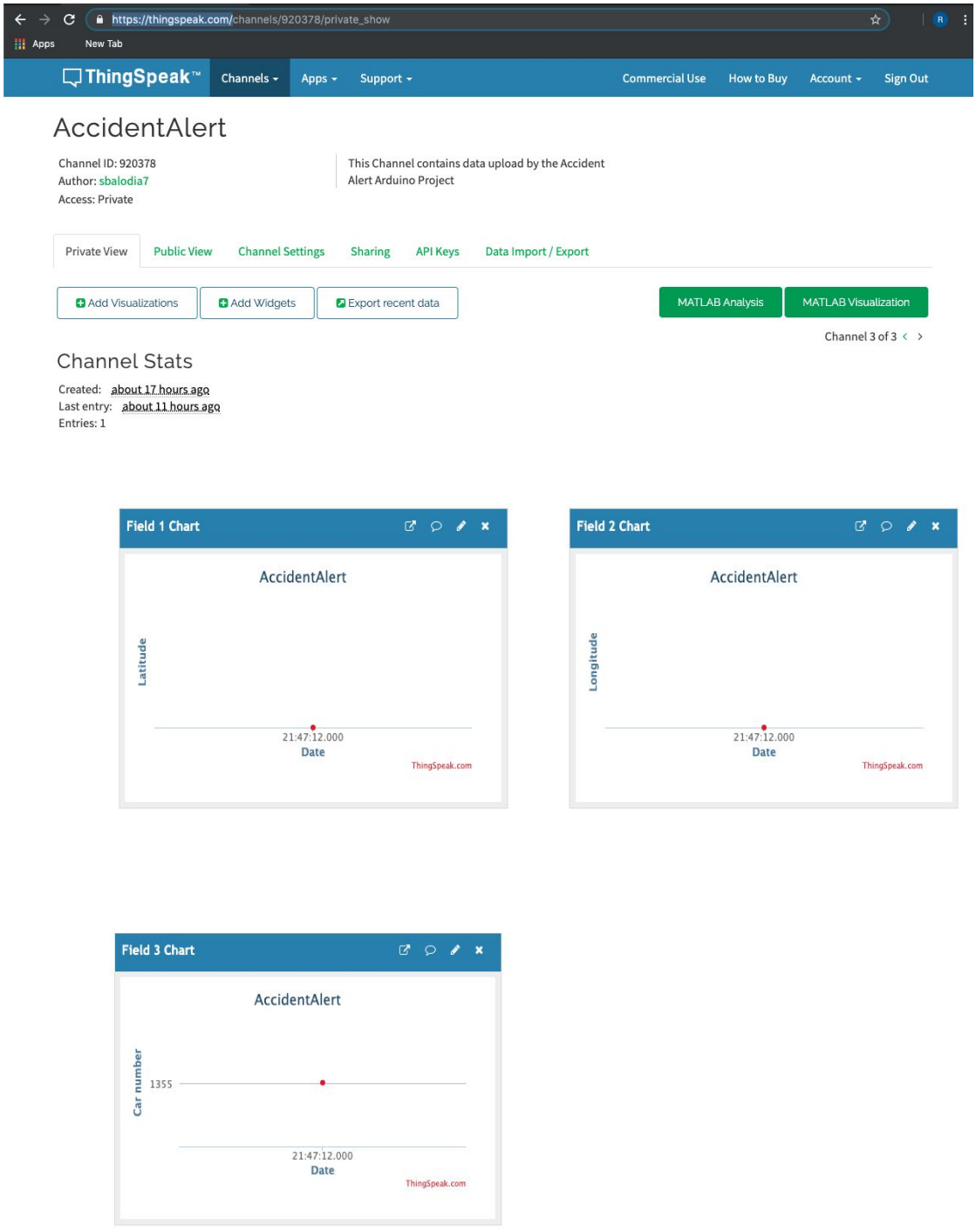
Cost Analysis of the above components

- **Arduino Uno** **Rs 300/-**
- **Bread Board** **Rs 50/-**
- **Accelerometer Sensor** **Rs 250/-**
- **ESP 8266 NodeMCU Wifi Module** **Rs.240/-**
- **GSM SIM 900 A** **Rs.875/-**
- **Jumper Wires** **Rs.90/-**
- **LCD** **Rs.90/-**
- **Buzzer** **Rs.20/-**
- **USB cable** **Rs.40/-**
- **Toy Car** **Rs.20/-**

Software Requirements

1.Cloud:

https://thingspeak.com/



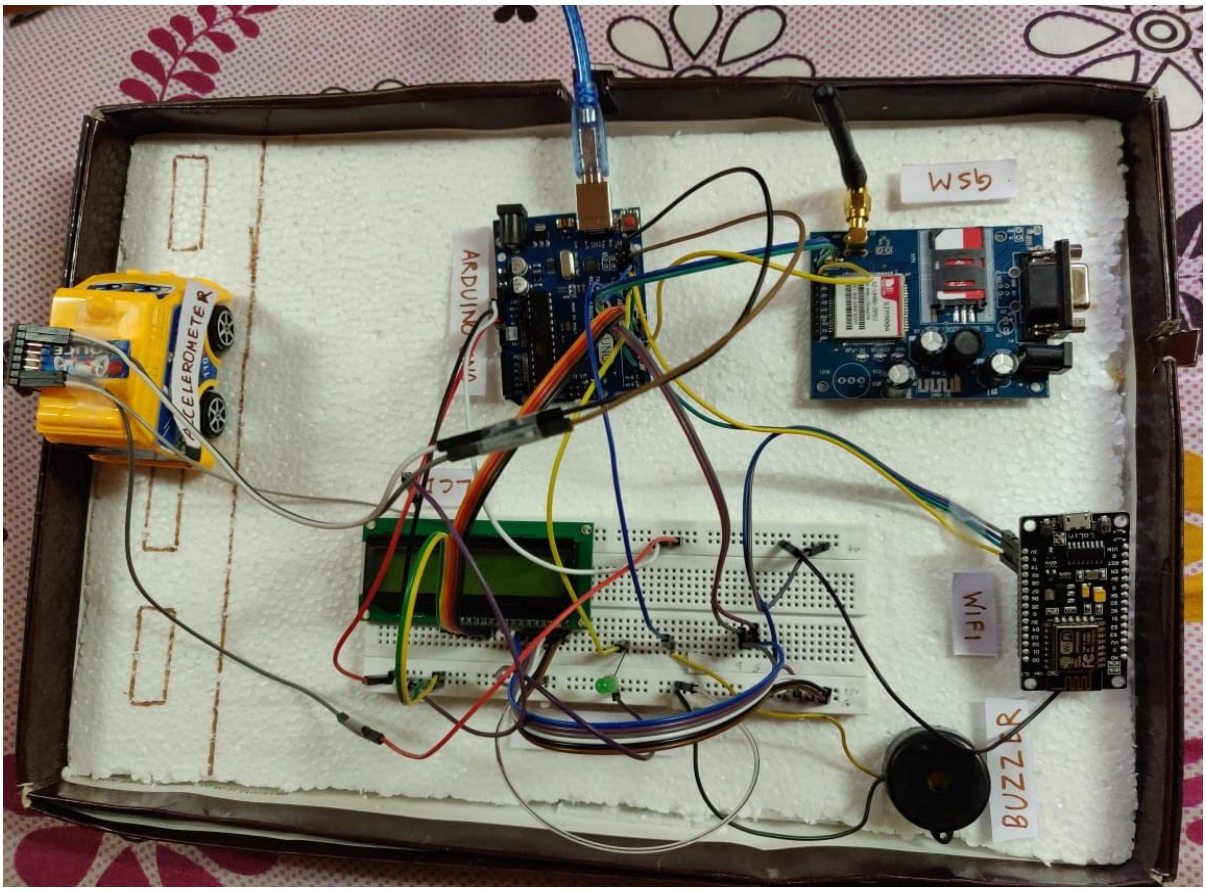
2. Operating System:

Mac OS X is used for system.

Arduino IDE is used for running interface code .

Design

Architectural diagram or Circuit diagram



Explanation about design {interaction between elements}

ARDUINO

Arduino - Uno board is used as a micro controller.

These sensors and components are connected to the Arduino – Uno.

GSM

Global System for Mobile communication is used for transmitting mobile voice and data services.

In our project GSM sends sms and calls to the number given once accident is detected.

ACCELEROMETER

An accelerometer is an electromechanical sensor used to measure acceleration and to detect orientation. Once orientation is determined. The Arduino can react accordingly.

WIFI MODULE

Wifi module is a wirelesss network used to send data to cloud(thingspeak.com). TCP/IP protocol stack can give a microcontroller access to wifi network.

LCD MONITOR

LCD is an electronic display module which uses liquid crystal to produce a visible image. The 16x2 displays 16 characters per line in two such lines.

BUZZER

Buzzer or beeper is an audio signaling device ,it beeps when accident is detected.

Implementation

Steps to be followed:

1.The connections are made as per the circuit diagram.

2. As the accident is detected by the accelerometer the wifi and gsm are initialized and lcd is used for the display.

3. The buzzer beeps for 2 seconds and SMS is sent to the administrator of the alert system.

4. After this, the latitude and longitude values of the location are uploaded to the cloud along with the car number which met with the accident.

5. Using the location of accidents we can track the location of the places where frequent accidents take place, using the graph.

6. A separate webpage will access the data in the cloud and show the accident location on Google maps along with few hospitals near the accident spot.

Source code:

```
#include<SoftwareSerial.h>

#include<LiquidCrystal.h>

#include <Wire.h>

#include <Adafruit_Sensor.h>

#include <Adafruit_ADXL345_U.h>


#define samples 10
```

```
#define minVal -4

#define MaxVal 4

#define led 11

#define buzzer 10


/* Assign a unique ID to this sensor at the same time */

Adafruit_ADXL345_Unified accel = Adafruit_ADXL345_Unified(12345);


SoftwareSerial gsm(2,3); //make RX arduino line is pin 2, make TX arduino line is pin 3.

SoftwareSerial wifi(12,13);//make RX arduino line is pin 12, make TX arduino line is pin 13.

LiquidCrystal lcd(4,5,6,7,8,9); // rs = 4,Contrast=GND, E = 5, 6 = D4, 7 = D5, 8 = D6, 9 = D7

//SoftwareSerial gps(10,11); //make RX arduino line is pin 10, make TX arduino line is pin 11.


int xsample=0;

int ysample=0;

int zsample=0;


String carNum = "1355";

//String mySSID = "Shivu";       // WiFi SSID

//String myPWD = "9661481381"; // WiFi Password


String mySSID = "Riyanchhi";       // WiFi SSID

String myPWD = "riya1054"; // WiFi Password

String myAPI = "80SI6JU4TQ0UFLH7";   // Write API Key
```

```
String myHOST = "api.thingspeak.com";
String myPORT = "80";
int i=0,k=0;
// Latitude and Longitude values
float latitude= 12.940084;
float longitude= 77.564927;
//int  gps_status=0;
//String Speed="";
//String gpsString="";
//String test ="$GPRMC";

void initModule(String cmd,String res, int t)
{
  char response[res.length()];
  res.toCharArray(response,res.length());
  while(1)
  {
    Serial.println(cmd);
    gsm.println(cmd);
    delay(100);
    while(gsm.available()>0)
    {
      if(gsm.find(response))
      {
```

```
        Serial.println(res);
        delay(t);
        return;
       }

       else
       {
        Serial.println("Error");
       }
    }
    delay(t);
  }
}

void espData(String command, const int timeout)
{
  Serial.print("AT Command ==> ");
  Serial.print(command);
  Serial.println("     ");

  String response = "";
  wifi.println(command);
  long int time = millis();
  while ( (time + timeout) > millis())
```

```
  {
    while (wifi.available())
    {
      char c = wifi.read();
      response += c;
    }
  }
  //Serial.print(response);
}

void setup()
{
  pinMode(led,OUTPUT);
  pinMode(buzzer,OUTPUT);
  if(!accel.begin())
  {
    /* There was a problem detecting the ADXL345 ... check your connections */
    Serial.println("Ooops, no ADXL345 detected ... Check your wiring!");
    while(1);
  }
  wifi.begin(115200);
  gsm.begin(9600);
  Serial.begin(9600);
  lcd.begin(16,2);
```

```
lcd.print("Accident Alert");
lcd.setCursor(0,1);
lcd.print("     System     ");
delay(2000);
lcd.clear();
Serial.println("Initializing....");
lcd.setCursor(0,1);
lcd.print("Please Wait...");
delay(1000);
// GSM AT commands
initModule("AT","OK",1000);
initModule("ATE1","OK",1000);
initModule("AT+CPIN?","READY",1000);
initModule("AT+CMGF=1","OK",1000);
initModule("AT+CNMI=2,2,0,0,0","OK",1000);
Serial.println("Initialized GSM");
lcd.clear();
lcd.print("Initialized GSM");
//Wifi AT Commands
espData("AT+RST", 1000);                      //Reset the ESP8266 module
espData("AT+CWMODE=1", 1000);                 //Set the ESP mode as station mode
espData("AT+CWJAP=\""+ mySSID +"\",\""+ myPWD +"\"", 1000);   //Connect to WiFi network
//while(!wifi.find("OK")) ;//Wait for connection
```

```
Serial.println("Initialized Wifi");
lcd.clear();
lcd.print("Initialized Wifi");
delay(1000);
lcd.clear();
lcd.print("Initialized");
lcd.setCursor(0,1);
lcd.print("Successfully");
delay(2000);
lcd.clear();
lcd.print("Caliberating ");
lcd.setCursor(0,1);
lcd.print("Accelerometer");
sensors_event_t event;
accel.getEvent(&event);

for(int i=0;i<samples;i++)
{
  xsample+=event.acceleration.x;
  ysample+=event.acceleration.y;
  zsample+=event.acceleration.z;
}

xsample/=samples;
```

```
  ysample/=samples;
  zsample/=samples;

  Serial.println(xsample);
  Serial.println(ysample);
  Serial.println(zsample);
  delay(1000);

  lcd.clear();
  //lcd.print("Waiting For GPS");
//  lcd.setCursor(0,1);
//  lcd.print("     Signal    ");
//  delay(2000);
//  gps.begin(9600);
//  get_gps();
//  show_coordinate();
//  delay(2000);
//  lcd.clear();
//  lcd.print("GPS is Ready");
//  delay(1000);
//  lcd.clear();
//  lcd.print("System Ready");
//  Serial.println("System Ready..");
 }//end of setup
```

```
void loop()
{
    /* Get a new sensor event */
   sensors_event_t event;
   accel.getEvent(&event);
   int value1=event.acceleration.x;
   int value2=event.acceleration.y;
   int value3=event.acceleration.z;

   int xValue=xsample-value1;
   int yValue=ysample-value2;
   int zValue=zsample-value3;

   Serial.print("x=");
   Serial.println(xValue);
   Serial.print("y=");
   Serial.println(yValue);
   Serial.print("z=");
   Serial.println(zValue);

   if(xValue < minVal || xValue > MaxVal  || yValue < minVal || yValue > MaxVal  || zValue < minVal ||
zValue > MaxVal)
   {
     /*Accident has occurred*/
```

```
  //get_gps();
  //show_coordinate();
  digitalWrite(led,HIGH);
  digitalWrite(buzzer,HIGH);
  delay(2000);
  digitalWrite(led,LOW);
  digitalWrite(buzzer,LOW);
  lcd.clear();
  lcd.print("Accident!");
  delay(2000);
  lcd.clear();
  lcd.print("Sending SMS ");
  Serial.println("Sending SMS");
  Send();
  Serial.println("SMS Sent");
  delay(2000);
  lcd.clear();
  lcd.print("Upload to Cloud");
  Upload();
  delay(2000);
  lcd.clear();
  lcd.print("Open Web Page");
  delay(5000);
 }//end of if
```

```
    else{
      lcd.clear();
      lcd.print("Driving Safely:)");
    }
}//end of loop


void Upload(){
  String sendData = "GET /update?api_key="+ myAPI
+"&field1="+String(latitude)+"&field2="+String(longitude)+"&field3="+carNum ;
    espData("AT+CIPMUX=1", 1000);       //Allow multiple connections
    espData("AT+CIPSTART=0,\"TCP\",\""+ myHOST +"\","+ myPORT, 1000);
    espData("AT+CIPSEND=0," +String(sendData.length()+4),1000);
    wifi.find(">");
    wifi.println(sendData);
    Serial.print("Sent Latitude: ");
    Serial.println(latitude);
    Serial.print("Sent Longitude: ");
    Serial.println(longitude);
    espData("AT+CIPCLOSE=0",1000);
    delay(2000);
    Serial.println("Upload Successful");
    lcd.clear();
    lcd.print("Upload done");
}//end of method
/*void gpsEvent()
```

```
{
 gpsString="";
 while(1)
 {
  while (gps.available()>0)            //Serial incoming data from GPS
  {
   char inChar = (char)gps.read();
    gpsString+= inChar;                    //store incoming data from GPS to temparary string str[]
    i++;
   // Serial.print(inChar);
    if (i < 7)
    {
     if(gpsString[i-1] != test[i-1])         //check for right string
     {
       i=0;
       gpsString="";
     }
    }
   if(inChar=='\r')
   {
    if(i>60)
    {
      gps_status=1;
      break;
```

```
    }
    else
    {
      i=0;
    }
   }
  }
  if(gps_status)
   break;
  }
}

void get_gps()
{
  lcd.clear();
  lcd.print("Getting GPS Data");
  lcd.setCursor(0,1);
  lcd.print("Please Wait.....");
   gps_status=0;
   int x=0;
   while(gps_status==0)
   {
    gpsEvent();
    int str_lenth=i;
```

```
    coordinate2dec();
    i=0;x=0;
    str_lenth=0;
   }
}

void show_coordinate()
{
    lcd.clear();
    lcd.print("Lat:");
    lcd.print(latitude);
    lcd.setCursor(0,1);
    lcd.print("Log:");
    lcd.print(longitude);
    Serial.print("Latitude:");
    Serial.println(latitude);
    Serial.print("Longitude:");
    Serial.println(longitude);
    Serial.print("Speed(in knots)=");
    Serial.println(Speed);
    delay(2000);
    lcd.clear();
    lcd.print("Speed(Knots):");
    lcd.setCursor(0,1);
```

```
   lcd.print(Speed);
}

void coordinate2dec()
{
  String lat_degree="";
    for(i=20;i<=21;i++)
      lat_degree+=gpsString[i];

  String lat_minut="";
     for(i=22;i<=28;i++)
      lat_minut+=gpsString[i];

  String log_degree="";
    for(i=32;i<=34;i++)
      log_degree+=gpsString[i];

  String log_minut="";
    for(i=35;i<=41;i++)
      log_minut+=gpsString[i];

    Speed="";
    for(i=45;i<48;i++)          //extract longitude from string
      Speed+=gpsString[i];
```

```
    float minut= lat_minut.toFloat();
    minut=minut/60;
    float degree=lat_degree.toFloat();
    latitude=degree+minut;

    minut= log_minut.toFloat();
    minut=minut/60;
    degree=log_degree.toFloat();
    longitude=degree+minut;
}
*/
void Send()
{
  gsm.println("AT");
  delay(500);
  serialPrint();
  gsm.println("AT+CMGF=1");
  delay(500);
  serialPrint();
  gsm.print("AT+CMGS=");
  gsm.print('"');
  gsm.print("9972047474");    //mobile no. for SMS alert
  gsm.println('"');
```

```
  delay(500);
  serialPrint();
  gsm.println("Accident Alert !!");
  gsm.println("Location of accident:");
  gsm.print("Latitude:");
  gsm.println(latitude);
  delay(500);
  serialPrint();
  gsm.print(" longitude:");
  gsm.println(longitude);
  delay(500);
  serialPrint();
//  gsm.print(" Speed:");
//  gsm.print(Speed);
//  gsm.println("Knots");
  delay(500);
  serialPrint();
  gsm.print("http://maps.google.com/maps?&z=15&mrt=yp&t=k&q=");
  gsm.print(latitude,6);
  gsm.print("+");            //28.612953, 77.231545   //28.612953,77.2293563
  gsm.print(longitude,6);
  gsm.write(26);
  delay(2000);
  serialPrint();
```

```
}


void serialPrint()

{

  while(gsm.available()>0)

    Serial.print(gsm.read());

}
```

AccidentAlert.html

```
<!DOCTY
PE html>
                    <html>
                     <head>
                      <title>Accident and Hospital Location</title>
                      <meta name="viewport" content="initial-scale=1.0, user-scalable=no">
                      <meta charset="utf-8">
                      <style>
                       /* Always set the map height explicitly to define the size of the div
                        * element that contains the map. */
                       #map {
                        height: 100%;
                       }
                       /* Optional: Makes the sample page fill the window. */
                       html, body {
                        height: 100%;
                        margin: 0;
                        padding: 0;
                       }
                      </style>
                      <script
                    src="https://maps.googleapis.com/maps/api/js?key=AIzaSyDoLuH2EjnCEDuYrS3ywIW31
                    7mzGtsze3w"></script>
                      <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
                      <script>
```

```
var map,x,y,res, accidentmarker, marker;
var markers=[
            [12.939067,77.578546,"A.V.Hospital"],
            [12.946096,77.567519,"Shekhar Hospital"],
            [12.941447,77.567608,"BMS Hospital"],
            [12.936695,77.584341, "Apollo Hospital"]
];
  function loadmaps(){

$.getJSON("https://api.thingspeak.com/channels/920378/fields/1/last.json?api_key=ITXSEX
FTGQDG1XNB", function(result){

    res = result;
    x=Number(res.field1);
  });

$.getJSON("https://api.thingspeak.com/channels/920378/fields/2/last.json?api_key=ITXSEX
FTGQDG1XNB", function(result){

    res = result;
    y=Number(res.field2);

  }).done(function() {

      initialize();
});

  }

  window.setInterval(function(){
  loadmaps();
    }, 9000);

function initialize() {
    var bounds = new google.maps.LatLngBounds();
  var mapOptions = {
   zoom: 18,
   center: {lat: x, lng: y}
  };
```

```
        map = new google.maps.Map(document.getElementById('map'),
            mapOptions);

        accidentmarker = new google.maps.Marker({
          position: {lat: x, lng: y},
          map: map,
          label: "A",
          title: "Accident Location",
          animation:  google.maps.Animation.BOUNCE,
        });
        // Loop through our array of markers & place each one on the map
            for( i = 0; i < markers.length; i++ ) {
            var position = new google.maps.LatLng(markers[i][0], markers[i][1]);
                    bounds.extend(position);
                    marker = new google.maps.Marker({
                    position: position,
                    map: map,
                    title: markers[i][2]
            });
            map.fitBounds(bounds);
        }

      }//end of initialize

    </script>
  </head>
  <body>
    <div id="map"></div>
  </body>
</html>
```

Conclusion

In Existing System, many applications such as mobile applications like "HELP ME ON MOBILE" is developed. *91# codes are also developed. If there is any emergency to that code women either call or send message In Proposed Work, multiple sensors like the heartbeat sensor, flex sensor, tilt sensor, vibration sensor is used to detect the condition of women if there is any emergency the message and the location is automatically sent to nearby police station and relatives. The Existing system needs a single click to get help. At sometimes women in the situation where a single click also cannot be done. May women be in a block out stage. At that time, body sensors help her to detect automatically.

A safety device for women, which can be carried using GPS and three different sensors has created. This may help women when there is any emergency. The GPS sends message automatically to the nearby police station and relatives by tracking their location. This may help women to move freely wherever she wants.