AI Monitoring & Alerting System

– NuFo

Team members:
M.JANANI
K.KRISHNAVENI
L.MAJELA ISABELLA
S.MYTHILI

Problem statement:

- Now-a-days there are many accidents and robberies are occurred in our surroundings .
- The mishap about the accidents are delayed to police or hospital .
- This is the major problem nowadays .
- So , we are here to propose a solution for this problem.

Proposed Solution:

- The solution for the above stated problem is to fix cameras that is capable for detecting the occurred incident is accident or robbery.
- This detected information is send as current location of the incident occurred to police and hospital.

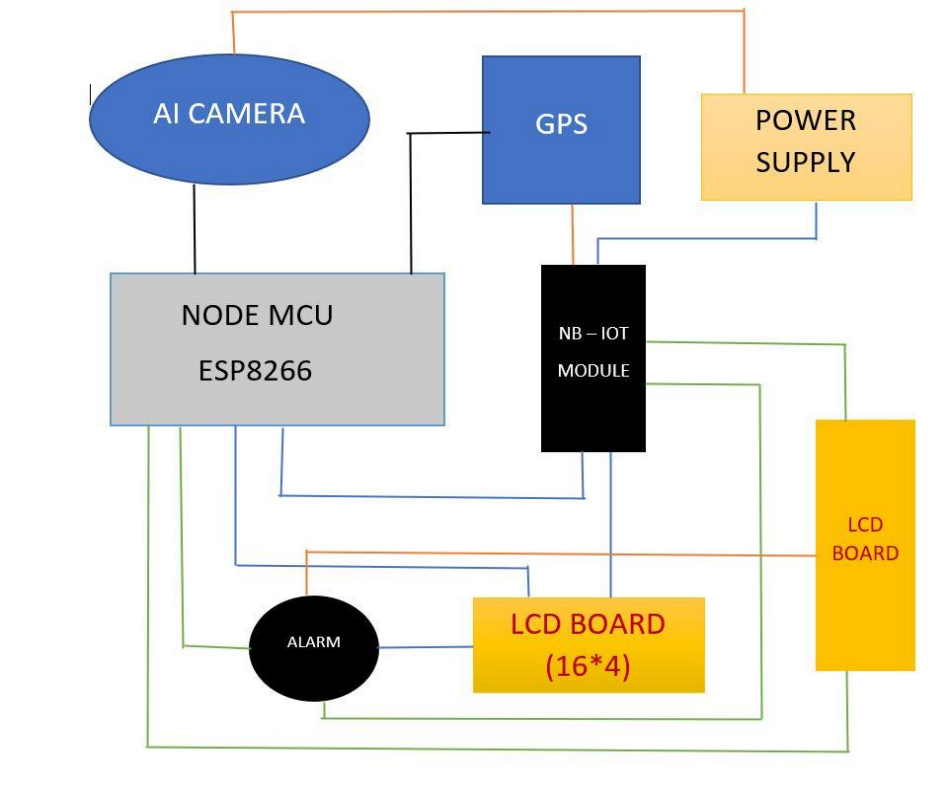
Hardware Component:

- NodeMCU-ESP8266
- GPS module
- GSM module
- AI camera or DISCREET CCTV
- Connection wires
- LCD(16*4)
- LED
- Resistor
- Alarm
- Accelerometer
- piezo

Methodology

- The Microcontroller here we are going to use NodeMCU-ESP8266.
- The accident or robbery that is taking place on local areas are detected through AI cameras or discreet CCTV.
- If anything occurred ,like robbery then the current location of the incident is send to police station through GPS module / NB – IOT module.
- If anything occurred ,like accident then the current location of the incident is send to the police as well as near by government hospital placed there through GPS module / NB – IOT module.
- This location is displayed to police or to hospital through LCD display and along with it, an alarm is set to give sound alert to the police or hospital.

Block diagram:



Coding:

```
#include<SoftwareSerial.h>
SoftwareSerial Serial1(2,3);
 //make RX arduino line is pin 2, make TX arduino line is pin 3.
SoftwareSerial gps(10,11);
#include<LiquidCrystal.h>
LiquidCrystal lcd(4,5,6,7,8,9);
#define x A1
#define y A2
#define z A3
int xsample=0;
int ysample=0;
int zsample=0;
#define samples 10
#define minVal -50
#define MaxVal 50
int i=0,k=0;
int  gps_status=0;
float latitude=0;
float logitude=0;
```

```
String gpsString="";

char *test="$GPRMC";

void initModule(String cmd, char *res, int t){

while(1){
    Serial.println(cmd);
    Serial1.println(cmd);
    delay(100);
    while(Serial1.available()>0){
       if(Serial1.find(res)){
        Serial.println(res);
        delay(t);
        return;
       }
else{
        Serial.println("Error");
       }
    }
    delay(t);
  }
}         return;
       }

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

```
void  setup(){
  Serial1.begin(9600);
  Serial.begin(9600);
  lcd.begin(16,2);
  lcd.print("Accident Alert  ");
  lcd.setCursor(0,1);
  lcd.print("     System     ");
  delay(2000);
  lcd.clear();
  lcd.print("Initializing");
  lcd.setCursor(0,1);
  lcd.print("Please Wait...");
  delay(1000);
  Serial.println("Initializing....");
  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("
```

```
lcd.clear();
lcd.print("Initialized");
lcd.setCursor(0,1);
lcd.print("Successfully");
delay(2000);
lcd.clear();
lcd.print("Callibrating ");
lcd.setCursor(0,1);
lcd.print("Acceleromiter");
for(int  i=0;i<samples;i++)
{
  xsample+=analogRead(x);
  ysample+=analogRead(y);
  zsample+=analogRead(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..");
} void loop()
{
    int value1=analogRead(x);
    int value2=analogRead(y);
    int value3=analogRead(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)
    {
      get_gps();
      show_coordinate();
      lcd.clear();
      lcd.print("Sending SMS ");
      Serial.println("Sending SMS");
      Send();
      Serial.println("SMS Sent");
      delay(2000);
      lcd.clear();
      lcd.print("System Ready");
    }
}
```

```
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(logitude);
    Serial.print("Latitude:");
    Serial.println(latitude);
    Serial.print("Longitude:");
    Serial.println(logitude);
    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();
     logitude=degree+minut;
} }
 void Send()
{
   Serial1.println("AT");
   delay(500);
   serialPrint();
   Serial1.println("AT+CMGF=1");
   delay(500);
   serialPrint();
   Serial1.print("AT+CMGS=");
   Serial1.print('"');
```

```
Serial1.print("+977980000000"); //mobile no. for SMS alert
   Serial1.println('"');
   delay(500);
   serialPrint();
   Serial1.print("Latitude:");
   Serial1.println(latitude);
   delay(500);
   serialPrint();
   Serial1.print(" longitude:");
   Serial1.println(logitude);
   delay(500);
   serialPrint();
   Serial1.print(" Speed:");
   Serial1.print(Speed);
   Serial1.println("Knots");
   delay(500);
   serialPrint();
   Serial1.print("http://maps.google.com/maps?&z=15&mrt=yp&t=k&q=");
   Serial1.print(latitude,6);
   Serial1.print("+");          //28.612953,77.231545   //28.612953,77.2293563
   Serial1.print(logitude,6);
   Serial1.write(26);
   delay(2000);
   serialPrint();
}
```

```
void  serialPrint()
{
  while(Serial1.available()>0)
  {
    Serial.print(Serial1.read());
  }
}
```

Conclusion:

- we would like to conclude that , if this project is implemented it will be easy for police to rush to the location immediately where the incident occurred and take appropriate actions needed .
- Also in case of accident it will be easy for ambulance to rush to the spot immediately without any delay to save a life.
- This is mainly implemented in local area where public rush is less and also in night times.
- so it is efficient if anything occurred and to take appropriate solution.

THANK YOU….