**ARDUINO UNO CODE:**

#include <LiquidCrystal.h>

#include <SoftwareSerial.h>

SoftwareSerial uart(8, 9);

const int rs = 2, en = 3, d4 = 4, d5 = 5, d6 = 6, d7 = 7;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

#include <TinyGPS.h>

#include <Wire.h>

#include <Adafruit\_Sensor.h>

#include <Adafruit\_ADXL345\_U.h>

#define button A0

Adafruit\_ADXL345\_Unified accel = Adafruit\_ADXL345\_Unified(12345);

TinyGPS gps;

float flat=0, flon=0;

int buz=10;

void read\_gps()

{

    bool newData = false;

  unsigned long chars;

  unsigned short sentences, failed;

  for (unsigned long start = millis(); millis() - start < 1000;)

  {

    while (Serial.available())

    {

      char c = Serial.read();

      if (gps.encode(c))

        newData = true;

    }

  }

  if (newData)

  {

    unsigned long age;

    gps.f\_get\_position(&flat, &flon, &age);

  }

}

int cnt=0;

void setup()

{   lcd.begin(16, 2);

  lcd.clear();

  lcd.setCursor(0, 0);

  lcd.print("Emergency call ");

  lcd.setCursor(0, 1);

  lcd.print("Message system");

  delay(1000);

  lcd.clear();

  lcd.setCursor(0, 0);

  lcd.print("for people with");

  lcd.setCursor(0, 1);

  lcd.print("epilepsy");

  delay(1000);

   Serial.println("AT");

  delay(1500);

  Serial.println("AT+CMGF=1");

  delay(500);

    lcd.clear();

  Serial.begin(9600);

    uart.begin(9600);

  accel.begin();

  pinMode(buz,OUTPUT);

  pinMode(button,INPUT);

  digitalWrite(buz,0);

  delay(2000);

}

void loop()

{   delay(200);

    sensors\_event\_t event;

    accel.getEvent(&event);

    int xval=event.acceleration.x;

    int yval=event.acceleration.y;

    int bval = analogRead(button);

    lcd.clear();

     lcd.setCursor(0,0);

    lcd.print("X:"+ String(xval)+"  Y:"+ String(yval) );

    lcd.setCursor(0,1);

    lcd.print("V:"+ String(bval) );

    cnt++;

if(cnt>4){

 uart.print("2850274,RJOUAMHNC2M35RE6,0,0,project1,12345678,"+String(xval)+","+String(yval)+","+String(bval)+","+String(16.3510)+","+String(81.0426)+",\n");

 cnt=0;

  }

if((bval>600))

    {

      digitalWrite(buz,1);

          send\_sms(1);

          delay(4000);

      makeCall();

      digitalWrite(buz,0);

    }

    if((xval<-5)|| (xval>5) || (yval>5) || (yval<-5))

    { lcd.clear();

      lcd.setCursor(0,0);

    lcd.print(" FALSE   ");

    send\_sms(2);

    }

else

 {

   digitalWrite(buz,0);

  }

}

void makeCall() {

  Serial.println("AT");

  delay(1000);

  Serial.println("ATD+917013451264;"); // Dial the number

  lcd.setCursor(0, 1);

  lcd.print("Call going on...");

  delay(20000);

  Serial.println("ATH");

  delay(1000);

}

void send\_sms(int k)

  {

Serial.println("Sending SMS...");

Serial.println("AT");

delay(1000);

Serial.println("ATE0");

delay(1000);

Serial.println("AT+CMGF=1");

delay(1000);

Serial.print("AT+CMGS=\"7013451264\"\r\n");// Replace x with mobile number

delay(1000);

if(k==1){

Serial.println(" EPILEPSY DETECTED: ");

Serial.println("https://www.google.com/maps/search/?api=1&query=" + String(16.3510)+ "," + String(81.0426));

lcd.clear();

lcd.setCursor(0,0);

lcd.print("message sent ");

}

if(k==2)

Serial.println(" FALSE DETECTION: ");

delay(500);

Serial.print(char(26));

delay(2000);

  }

**NODE MCU CODE:**

#include "ThingSpeak.h"

#include <ESP8266WiFi.h>

int statusCode = 0;

unsigned long lastTime = 0;

unsigned long timerDelay = 16000;

WiFiClient  client;

const int FieldNumber1=1;

String strs[14]={"2838060","P6Z07T35NVBAGTEV","0","0","project1","12345678","0","0","0","0","0","0","0","0"};

int StringCount =0;

int prv=0;

int led=D4;

void setup()

{

  WiFi.mode(WIFI\_STA);

  ThingSpeak.begin(client);

  Serial.begin(9600);

  pinMode(led,OUTPUT);

  delay(1000);

  digitalWrite(led,1);

}

void loop()

{

  //----------------- Network -----------------//

  if (WiFi.status() != WL\_CONNECTED)

  {

    //Serial.print(".");

    delay(1000);

    WiFi.begin(strs[4], strs[5]);

      for(int kk=0;kk<10;kk++)

      {

        digitalWrite(led,0);

        delay(300);

        digitalWrite(led,1);

        delay(300);

      }

   if(WiFi.status() == WL\_CONNECTED)

    Serial.println("ok");

  }

  //---------------- Channel 1 ----------------//

  const char\* string2 = strs[2].c\_str();

  const char\* string3 = strs[3].c\_str();

  int temp = ThingSpeak.readLongField(atol(string2), FieldNumber1, string3);

  statusCode = ThingSpeak.getLastReadStatus();

  if (statusCode == 200)

  {

    if(temp !=prv)

    {

      prv=temp;

       Serial.print(temp);

    }

  }

  delay(100);

if (Serial.available())

{

String rcv = Serial.readStringUntil('\n');

if ((millis() - lastTime) > timerDelay)

{

   StringCount=0;

     while (rcv.length() > 0)

  {

    int index = rcv.indexOf(',');

    if (index == -1) // No space found

    {

      strs[StringCount++] = rcv;

      break;

    }

    else

    {

      strs[StringCount++] = rcv.substring(0, index);

      rcv = rcv.substring(index+1);

    }

  }

 ThingSpeak.setField(1, strs[6]);

 ThingSpeak.setField(2, strs[7]);

 ThingSpeak.setField(3, strs[8]);

 ThingSpeak.setField(4, strs[9]);

 ThingSpeak.setField(5, strs[10]);

 ThingSpeak.setField(6, strs[11]);

 ThingSpeak.setField(7, strs[12]);

 ThingSpeak.setField(8, strs[13]);

 const char\* string0 = strs[0].c\_str();

 const char\* string1 = strs[1].c\_str();

 int x = ThingSpeak.writeFields(atol(string0), string1);

 if(x == 200){

  delay(10);

  }

else{

  delay(10);

  }

  lastTime = millis();

 }

}

  delay(500);

}