IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

# IMPLEMENTATION PAPER ON IOT BASED VEHICLE ACCIDENT PREVENTION AND DETECTION SYSTEM

**1 SUSHMA TELRANDHE ,2 ADIBA SADAF, 3 RAFAT TASNEEM, 4 SHIFFA TAHA,**

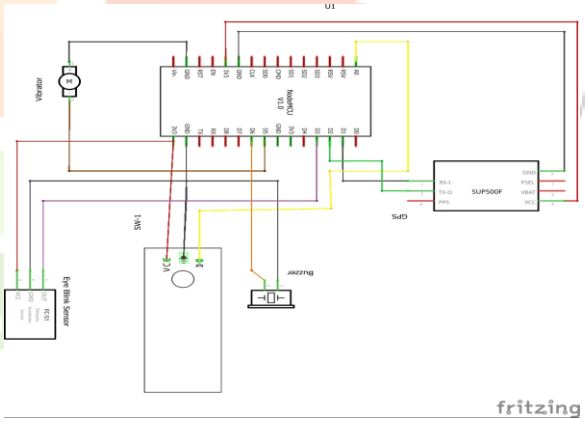
**5 NILIMA JARAD**

GURU NANAK INSTITUTE OF ENGINEERING AND TECHNOLGY NAGPUR

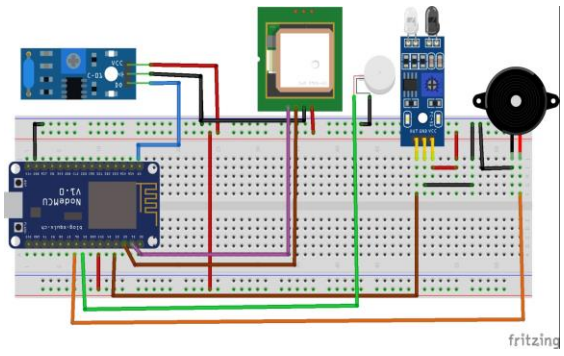
## INTRODUCTION

- The main purpose behind the implementation of the accident prevention is to minimize the roads accidents which causes the loss of valuable human life another valuable goods.
- Accident prevention and detection system (APDS) save lives by reducing the time require for emergency responders to arrive.
- We are developing one IoT application which will help to reduces percentage of accident.
- According to the statistics, reducing accident delay time by even 1 minute can save 6 % of lives.

## BLOCK DIAGRAM




## CIRCUIT DIAGRAM




## WORKING

1.For accident prevention, we have incorporated an IR proximity sensor that detects drowsiness in the driver. Whenever the sensor

detects drowsiness, the MCU raises an alarm and activates the vibrator to alert the driver. Additionally, the system sends an SMS to the registered number, informing them of the driver's condition. For accident detection, we have integrated a vibration sensor and a GPS module. The vibration sensor is placed in the vehicle and detects any sudden movements or impacts that could indicate an accident.

2.henever an accident is detected, the MCU sends an IFTTT request using Webhook and GPS location to the registered mobile number, notifying them of the accident and providing the exact location of the vehicle.

3.By combining these various sensors and modules, our system aims to improve road safety by preventing accidents caused by drowsy driving and quickly detecting accidents to ensure timely assistance to those involved.

Send Alert SMS using IFTTT Webhook:

IFTTT stands for “If This Than That”, and it is a free web-based service to create chains of simple conditional statements called applets.


This means you can trigger an event when something happens. In this example, the applet sends three random values to your email when the ESP8266 makes a request. You can replace those random values with useful sensor readings.

1. Open the left menu and click the “**Create**” button.
2. Click on the “**this**” word. Search for the “**Webhooks**” service and select the Webhooks icon.
3. Choose the “**Receive a web request**” trigger and give a name to the event. In this case, I’ve typed “**test_event**”. Then, click the “**Create trigger**” button.
4. Click the “**that**” word to proceed. Now, define what happens when the event you’ve defined is triggered. Search for the “**Email** / Android SMS” service and select it. You can leave the default options.
5. . Press the “**Finish**” button to create your Applet.

CODE

```
#include <ESP8266WiFi.h>

#include <SoftwareSerial.h>

#include <TinyGPS++.h>

TinyGPSPlus gps;

SoftwareSerial ss(4, 5);

const char* ssid     = "SSID";   //replace with your own SSID
```

```
const char* password = "1234567890";
//replace with your own password

const char* host = " ";

const char* resource =
"/trigger/SMS/with/key/dTzkjdLWLg8xxxxxxxxx";

const char* server = "maker.ifttt.com";

String member = "";

//https://maker.ifttt.com/trigger/SMS/with/key/dTz
kjdLWxxxxxxxx

float latitude , longitude;

int year , month , date, hour , minute , second;

String date_str , time_str , lat_str , lng_str;

int pm;

void setup()

{

  ss.begin(9600);



  Serial.println();

Serial.print("Connecting to WiFi...        ");

  Serial.print("Connecting to ");
Serial.println(ssid);

  WiFi.mode(WIFI_STA);

  WiFi.begin(ssid, password);

while (WiFi.status() != WL_CONNECTED)

  {

    delay(500);

    Serial.print(".");

  }

  Serial.println("");

  Serial.println("WiFi connected");

  Serial.println("IP address: ");

  Serial.println(WiFi.localIP());

   Serial.print("Connecting to ");

  Serial.print(server);

  WiFiClient client;

  int retries = 5;

  while(!!!client.connect(server, 80) && (retries--
> 0)) {

    Serial.print(".");

  }

Serial.println();

  if(!!!client.connected()) {

    Serial.println("Failed to connect...");

  }

Serial.print("Request resource: ");

  Serial.println(resource);

  String jsonObject = String("{\"value1\":\"") +
member + "\",\"value2\":\""

+ "Present" + "\",\"value3\":\"" + "" + "\"}";

 client.println(String("POST ") + resource + "
HTTP/1.1");

  client.println(String("Host: ") + server);
```

```
  client.println("Connection: close\r\nContent-
Type: application/json");

  client.print("Content-Length: ");

  client.println(jsonObject.length());

  client.println();

  client.println(jsonObject);

  int timeout = 5 * 10; // 5 seconds

  while(!!!client.available() && (timeout-- > 0)){

    delay(100);

  }

  if(!!!client.available()) {

    Serial.println("No response...");

  }

  while(client.available()){

    Serial.write(client.read());

  }

  Serial.println("\nclosing connection");

  client.stop();

}

void loop()

{

}

void getgps()

{

while (ss.available() > 0)

  if (gps.encode(ss.read()))

  {

    if (gps.location.isValid())

    {

   latitude = gps.location.lat();

      lat_str = String(latitude , 6);

      longitude = gps.location.lng();

     lng_str = String(longitude , 6);

      Serial.print("Latitude:");

      Serial.print(latitude);

      Serial.print("longitude:");

      Serial.print(longitude);

}

    if (gps.date.isValid())

    {

      date_str = "";

      date = gps.date.day();

      month = gps.date.month();

      year = gps.date.year();

      if (date < 10)

        date_str = '0';

date_str += String(date);

      date_str += " / ";
```

```
      if (month < 10)
       date_str += '0';
date_str += String(month);
      date_str += " / ";
      if (year < 10)
 date_str += '0';
 date_str += String(year);
}
 if (gps.time.isValid())
    {
      time_str = "";
      hour = gps.time.hour();
      minute = gps.time.minute();
      second = gps.time.second();
      minute = (minute + 30);
      if (minute > 59)

      {

        minute = minute – 60
         hour = hour + 1;
  }
 hour = (hour + 5) ;
   if (hour > 23)
 hour = hour - 24;
 if (hour >= 12)
 pm = 1;
 else
pm = 0;
 hour = hour % 12;
if (hour < 10)
       time_str = '0';
      time_str += String(hour);
      time_str += " : ";
      if (minute < 10)
 time_str += '0';
  time_str += String(minute);
time_str += " : ";     if (second < 10)
       time_str += '0';
time_str += String(second);
      if (pm == 1)
 time_str += " PM ";
else
       time_str += " AM ";
 }
  }
}
```

## APPLICATION

- Transportation business where almost daily accidents occur due to driver fatigue.
- Operators those working on machine and nuclear power plants where continuous monitoring is necessary.
- To alert the nearby medical services about the accident so as to provide immediate medical aid.
- Military applications where high intensity monitoring of soldier is needed.
- At night car drivers sleeping during driving get help due to his system. Security guard cabins.

## ALGORITHM

Step 1: System will get initialised and it will sense blinking of eyes through IR sensor

Step 2: If it doesn't detect blinking then it will send alert signals as vibration output for the driver to wakeup.

Step 3: System will check for signals recieved from vibration sensor.

Step 4: If vibration sensor value is greater than threshold value then the signal access will go to IFTTT through webhook.

Step 5: IFTTT service will request sms from mobile to send GPS co-ordinates of the vehicle and will send it in a url format through sms.

## LIST AND COST OF COMPONENTS

Node MCU : 500

GPS Modem: 1000

Vibration sensor: 200

Eyeblink sensor : 1000

Buzzer module: 150

Vibrator:100

PCB: 250

Connecting wires:

7805: 20

Bridge rectifier: 50

Capacitor: 100uf 10uf

Resistor: 220 ohm, 1K

LEDs:

LCD: 250

I2C module:100

Motor: 150.

AIM

Our project aims to improve road safety by incorporating various sensors and modules into a Node MCU-based system. The system has two main parts: accident prevention and accident detection.