

IIT ALLAHABAD

TESLA PROJECT

REPORT ON

“WOMEN’S SAFETY DEVICE”

BY

PRANAVI TADIVALASA

ABSTRACT

Women have no safety in society. A report noted that reported cases of crimes against women rose 83 percent between 2007 and 2016, where there were four cases of rape every hour. Most women may not achieve their dreams because of these reasons. Due to the increase in crime rate, women are feeling insecure to step out of the house. With the help of technology, we can build devices to enhance the safety of women.

So, the motivation behind this project is to make women feel that they are secure in all conditions. This device will generate an emergency alarm, will send the message, and also the SOS call to the concerned person. Women carrying this device can send the information to the police by pressing the SOS button.

In this project NodeMCU is interfaced with IFTTT (If This Than That) for getting google map location URL and sending alert SMS and SOS Call. When the SMS button is pressed the whole system gets active and sends the message which is attached with the google map location URL which is fetched from geolocation API, this API fetches live location data and sends the data to the predefined number via SMS using IFTTT applets like Android SMS. Through this geolocation API, other than ESP board not even a GPS module is needed to get our live coordinates. When the SOS Call button is pressed the whole system gets active and sends a call to the predefined number via VoIP Calls using IFTTT applets. The operating voltage of the project is 5v, This project can be powered using a power bank or batteries.

The main aim is to construct a safe device for women that can help them to communicate with someone, for example, relatives or police.

INTRODUCTION

Though we are living in 21st century, people are facing lots of problems such as poverty, pollution, corruption, illiteracy and women safety. Due to increase in crime rate, women are feeling insecure to step out of the house. In many sectors working hours are till the late evening and there are night shifts too. Women working at this hour have to travel at late night, which bring an insecure feeling. With the help of technology, devices can be built to enhance the safety of the women.

This device will generate an emergency alarm and will have option to call and also send the message to concerned person. Women carrying this device can send the information to the police by pressing SOS button which will send SMS and Call with the current location, using this information police will save the victim. In this project NodeMCU is interfaced with IFTTT for getting location co-ordinates, sending alert SMS and also SOS Call.

Objective:

- To design a device that can help women in any extreme critical condition such as abduction, violation, despoliation, maltreatment, molestation, etc.
- This device should also be used on children to control child trafficking or even human trafficking.

Problem Statement:

This project is an attempt to make a change in the life of every woman in terms of their safety and security. Many females or children face the same problem of being troubled in some or the other way, to stop this kind of activity that is to bring fear in the people who dare to do such activities, a solution has to be found. The result of this project or device should make a woman independent and feel safe.

PROPOSED METHODOLOGY

- The main purpose of this project is to provide safety when women feel insecure. The main two components are Pushbuttons and NodeMCU esp8266.
- For getting the Live Location of the person, I have used Geolocation API named [FreeGeolpAPI](#), which fetches the current location.
- NodeMCU is a WIFI module that connects to the internet using the WIFI Network provided.
- Here the live coordinates fetched from Geolocation API is send to a mobile phone via SMS using the IFTTT applets like Webhook and Android SMS.
- The GPS Coordinates received from the API is sent to the IFTTT server Webhook using NodeMCU and the IFTTT server will send the Alert message and google map URL to the Mobile Number provided.
- By Press and holding the SMS button, the entire system gets activated and immediately a SMS is sent to the concerned person with GPS Coordinates.
- By Press and holding the SOS button , the entire system gets activated again and immediately sends an SOS call to the concerned person asking for help.

PROJECT DESCRIPTION

In this project we are using the concept of IoT (Internet of things) for implementing the safety of women in the society. The device will get activated as soon as SMS/SOS Call button is pressed and hold, the device will send her exact location google map URL, with an alert message to the concerned person for getting help as soon as possible.

I have built the device using mainly two components that is, NodeMCU esp8266 and Pushbuttons.

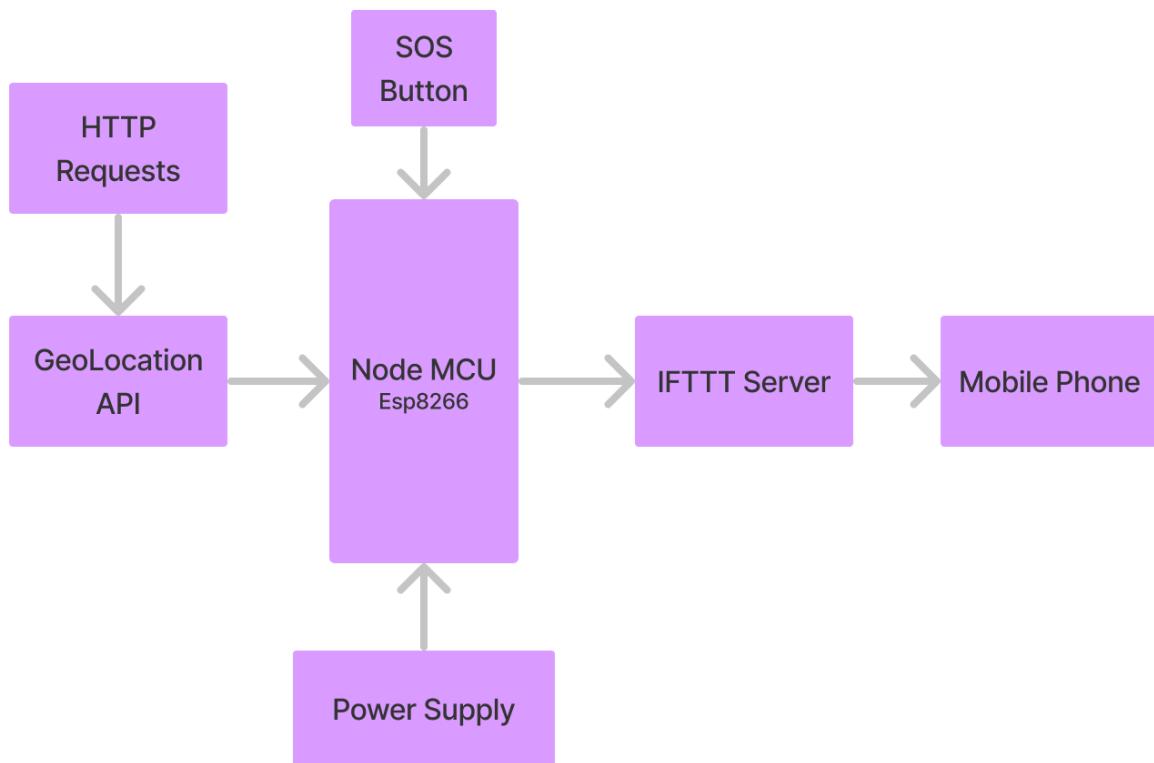


Fig 1.1: Block diagram

The above block diagram explains the working of the device in brief manner. The block diagram has mainly 4 important blocks that is Nodemcu,GeoLocation API, SOS, IFTTT server.

The steps followed in the project are as follows:

- Check the status of SOS button.
- Collect the location details of the device from GeoLocation API
- Connect to given WIFI and establish a connection with IFTTT server.
- Send the GPS details with alert message and sos call to the concerned person.

The circuit diagram of the project is given below:

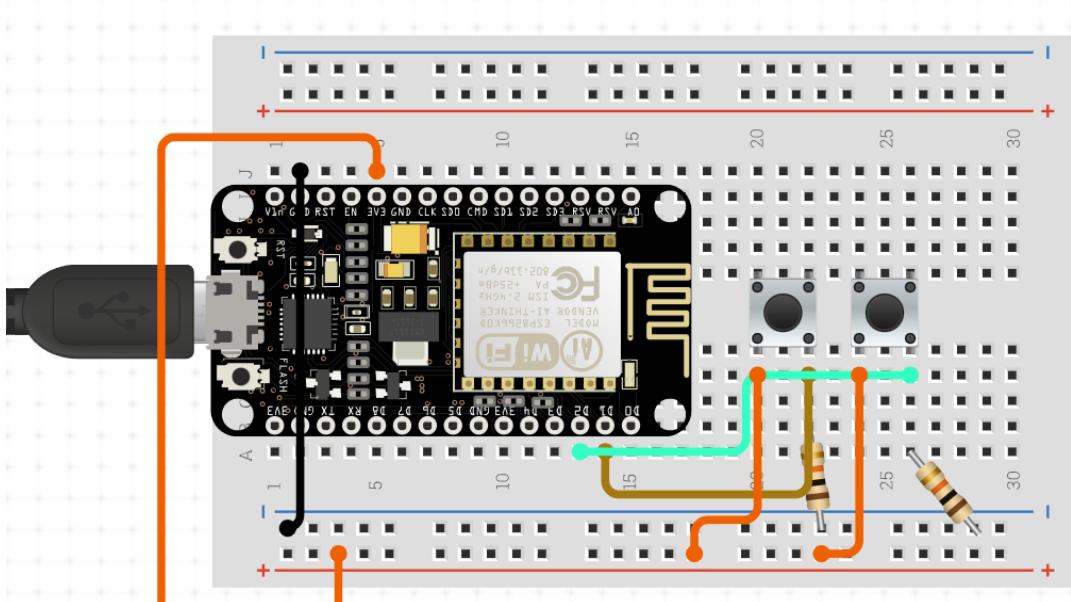


Fig1.2: Circuit Diagram

The components used in the circuit are:

- NodeMCU esp8266
- Push Buttons
- Resistors
- Power Source

NodeMCU esp8266:

NodeMCU is an open source IoT platform, based on Lua firmware which runs on ESP8266 WIFI SoC module from Espressif System and based on ESP-12 module hardware. It was developed to replace AT commands with Lau scripting to make it easier for the developers. The NodeMCU is a Low cost WIFI chip with full TCP/IP stack and complete microcontroller features to help the developers to use is in IoT applications. Developed by Shanghai-based Chinese manufacturer, Espressif.

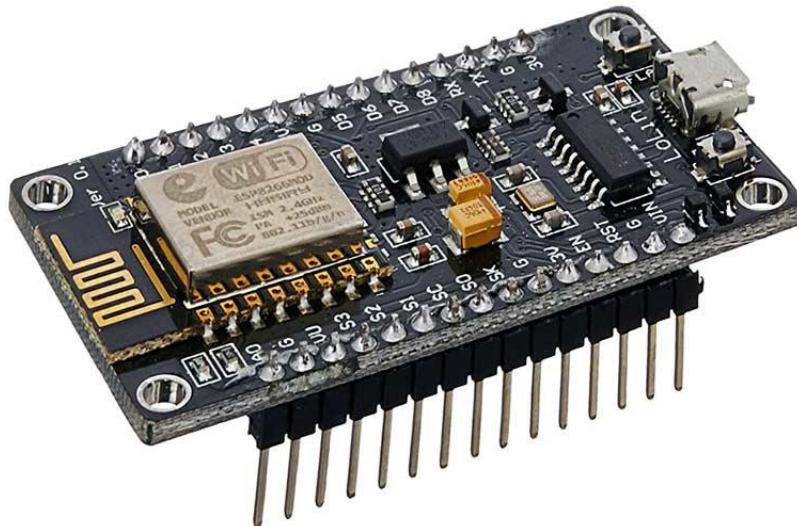


Fig1.3: NodeMCU esp8266

Features of NodeMCU esp8266:

- Interactive.
- Low Cost.
- WIFI enabled.
- USB-TTL included.
- Smart.
- Programmable.
- Open-source.
- Plug N play.

Specifications of NodeMCU esp8266:

- Type : Single-board microcontroller
- Developer : ESP8266 Opensource Community
- Operating system : XTOS
- CPU : ESP8266
- Digital I/O Pins (DIO) : 16
- Analog Input Pins (ADC) : 1
- Flash Memory : 4 MB
- SRAM : 64 KB
- Clock Speed : 80 MHz
- Power by : USB
- Power Voltage : 3v, 5v
- Code : Arduino Cpp
- IDE Used : Arduino IDE

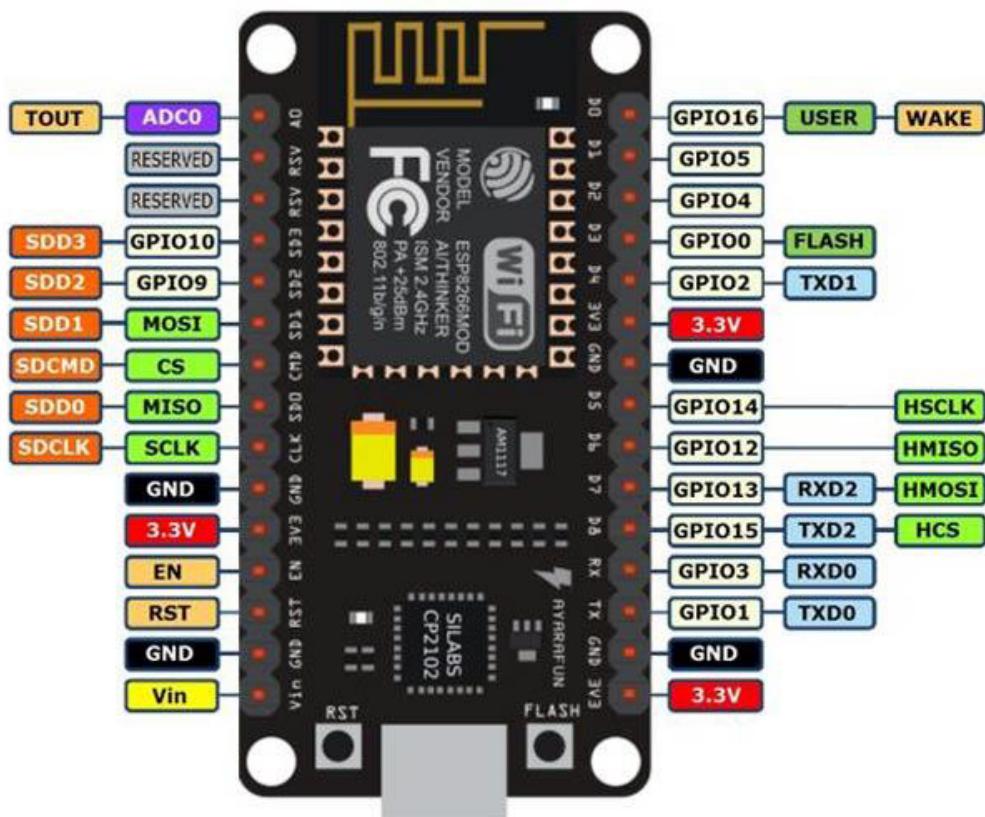


Fig 1.4: Pin Diagram of NodeMCU esp8266

Application of NodeMCU:

- Creating prototypes of IoT devices.
- Low power consumption projects.
- Projects which need multiple I/O interface with the WIFI.

Advantages:

- Low cost
- Small size microcontroller
- Low energy consumption
- Built-in WIFI support
- Programmable using Arduino IDE.

Disadvantages:

- Need to learn Lua scripting.
- Pinout of NodeMCU is confusing compared to Arduino.

Push button/SOS button:

A push button is a component which can be used as a switch to control an action in a machine or some type of process. It is made of plastic with metal connectors. We used it with a 10k resistor to act as a pulldown switch.



Fig 1.5: Pushbutton

Resistor:

Resistors are the electronic components who have a specific, constant electrical resistance. The resistor's resistance will limit the flow of the electrons in the circuit. It is a passive component that is the only consume power but can't generate the power. They are commonly used with the active components like op-amps, microcontrollers, and other integrated circuits. Commonly they are used to limit current, divide voltages.

The resistance of the resistor is measured in ohms. The symbol of the resistance is given by a Greek letter omega: Ω .

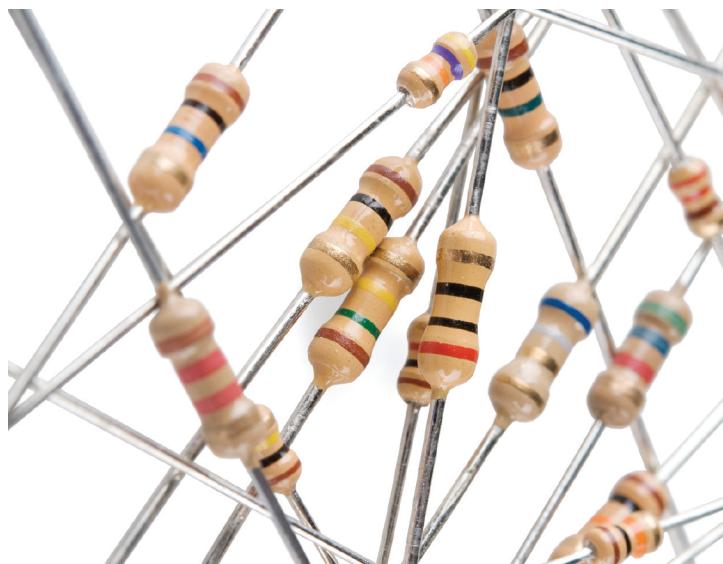


Fig 1.6: Resistors

Power Source:

Any battery or power bank can be used to power the system, operating voltage of the system is 5V, we are using a power bank with output voltage of 5V.

Software Specification:

The main two Software used to build the project are:

- Arduino Software IDE
- IFTTT

Arduino Software (IDE):

The Arduino IDE (Integrated Development Environment) is a Board development application that is written in function of C or C++. It is used to write, compile and upload programs to Arduino board and other third party microcontrollers which support the Arduino software.

Developers: Arduino Software

Software used: Arduino 1.8.13



Fig 1.7: Arduino Software IDE

Advantages of Arduino IDE:

- Open Source
- Portable
- User friendly programming language
- Can program third party microcontrollers as well.

IFTTT(If This Than That):

It is a free web-based service, where we can create chain of simple conditional statements, called Applets. A applets is triggered by changes which occur within other web-services such as Facebook, Gmail, Pinterest, etc.

I have used IFTTT to create an applet which has Webhook and Android Message in chain for building our project.



Fig 1.8:IFTTT(If This Then That)

Advantages of IFTTT:

- Simplifies automation
- Saves time
- Wide-ranging support
- Free of cost
- Ready-made applets

Disadvantages of IFTTT:

- Limited triggers and actions
- They wont work as expected
- No multi-level support

Programming of NodeMCU:

- Install the latest Arduino IDE at the Arduino website, install any version from 1.8 level or later.

- Open Arduino IDE and open preference window. Post https://arduino.esp8266.com/stable/package_esp8266com_index.json into Additional Board Manager URLs field.

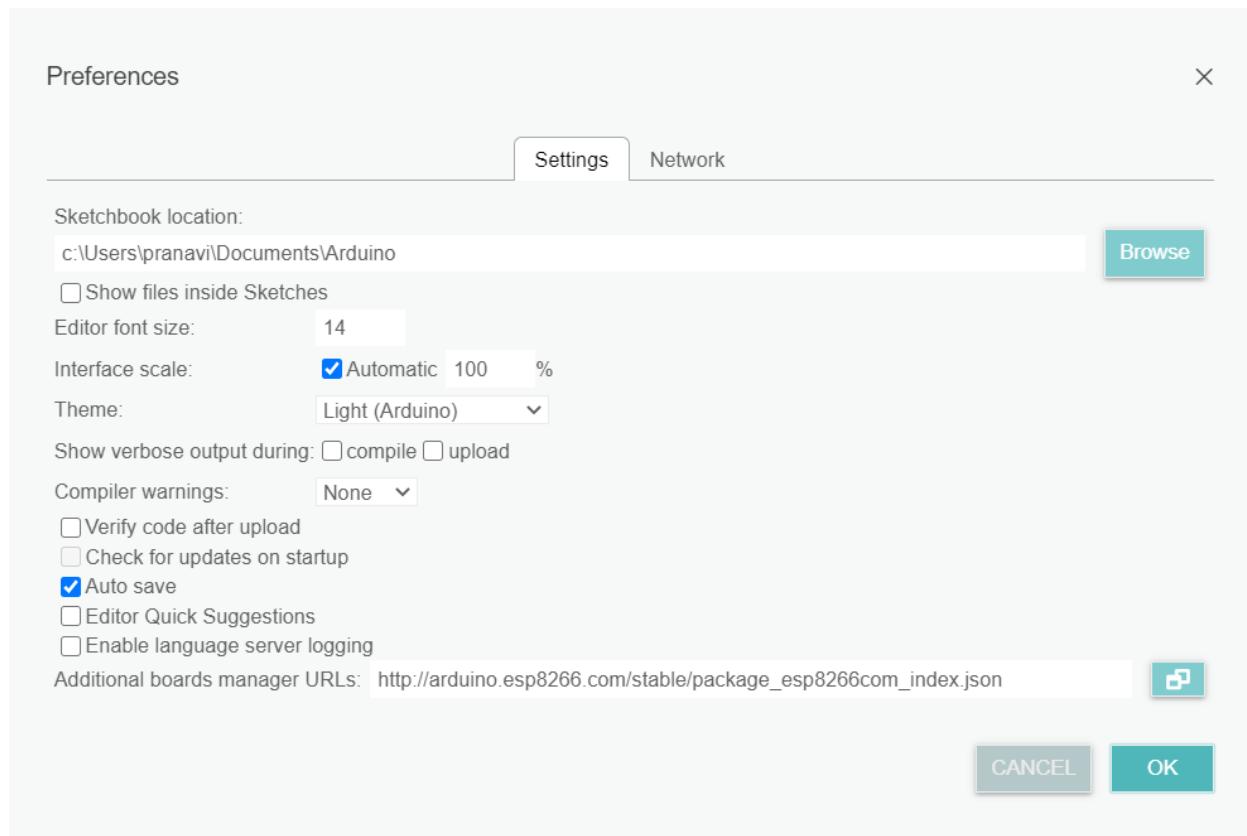


Fig 1.9: Addition of URLs

After Completion:

- Connect the NodeMCU to the PC with the USB port, install the drivers if needed.
- Check which port number is assigned to the board.
- Open Board Manager from tools> esp8266 Modules platform.
- Select Generic ESP8266 module.
- Upload using Serial, 80Mhz CPU frequency, Flash Size 4M, upload speed of 115200
- Upload the code into the NodeMCU after compilation.

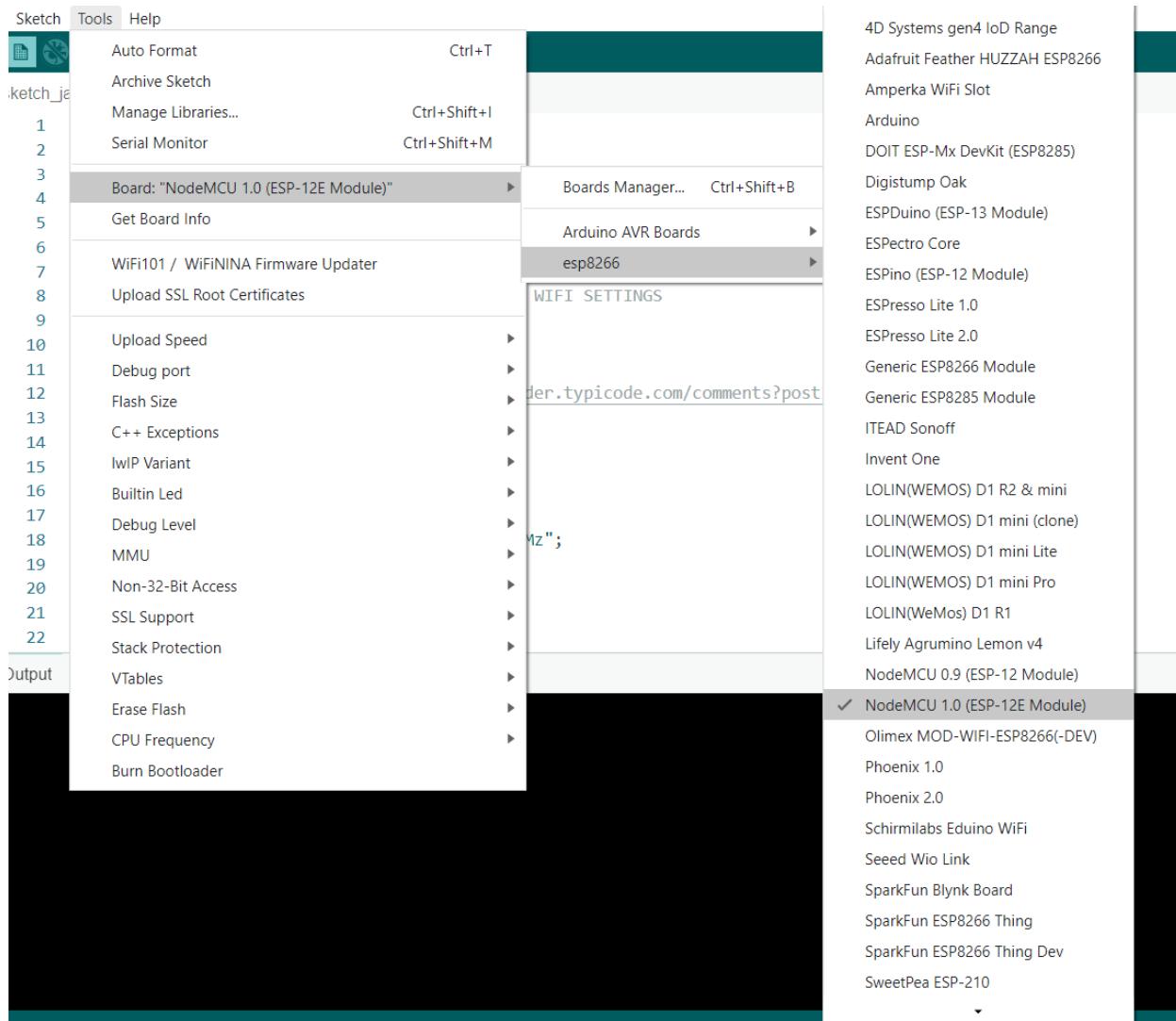


Fig 1.10: Selection of Board specs

Creation of IFTTT applet in IFTTT website:

Here, IFTTT is used to use Webhook and Android SMS service in chain. So, the GPS coordinates are sent to concerned person using NodeMCU. Here Webhook is triggered from NodeMCU to activate Android SMS to send the SMS to the concerned person.

First step is creating account in IFTTT.

Get started with IFTTT

 Continue with Google

 Continue with Facebook

Or use your password to [sign up](#) or [sign in](#)

Fig 1.11: Creating account on IFTT

Sign in to IFTTT using IFTTT account.

After Sign in, click on My Applets and then select New Applet shown below.

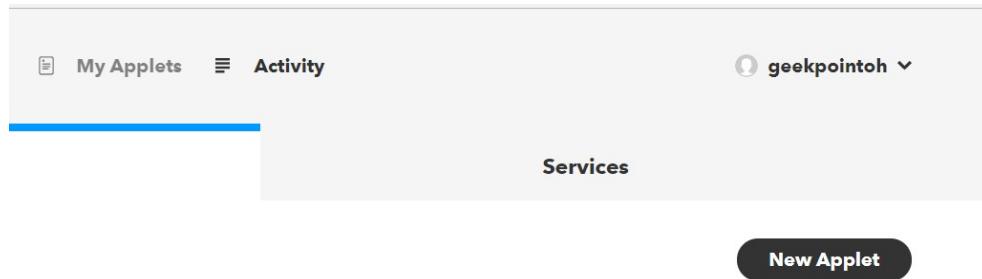


Fig 1.12: Applet creation in IFTTT Website

Now click on **This** as shown below

Create your own



You've created 2 of 3 applets

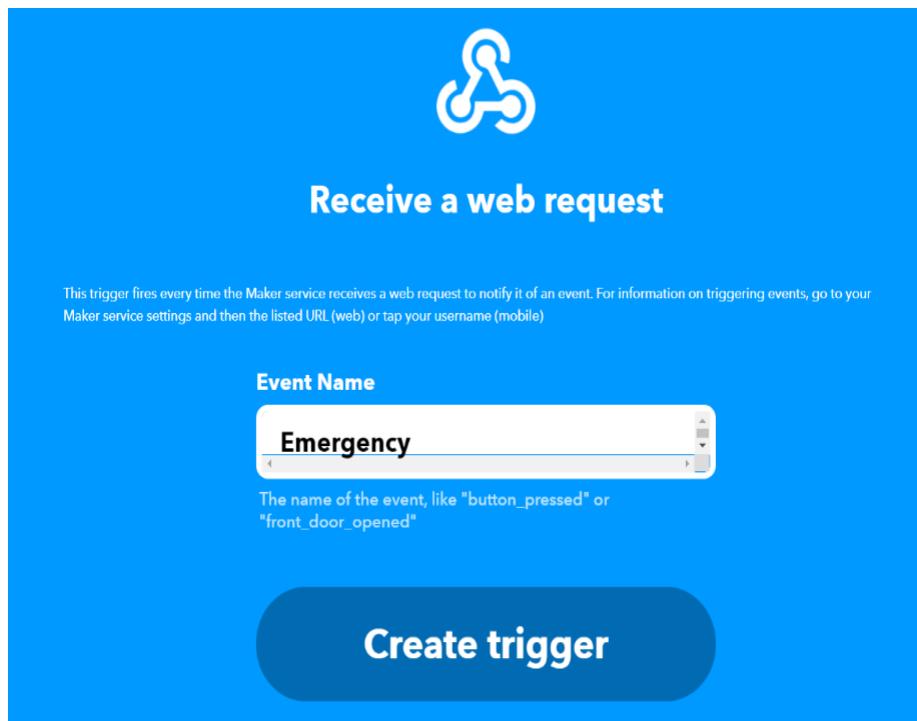
If This

Add

Then That

Fig 1.13: Click on This and select Webhooks

Now Select Webhooks and Add trigger



Now click on That and add the action which is here Android SMS

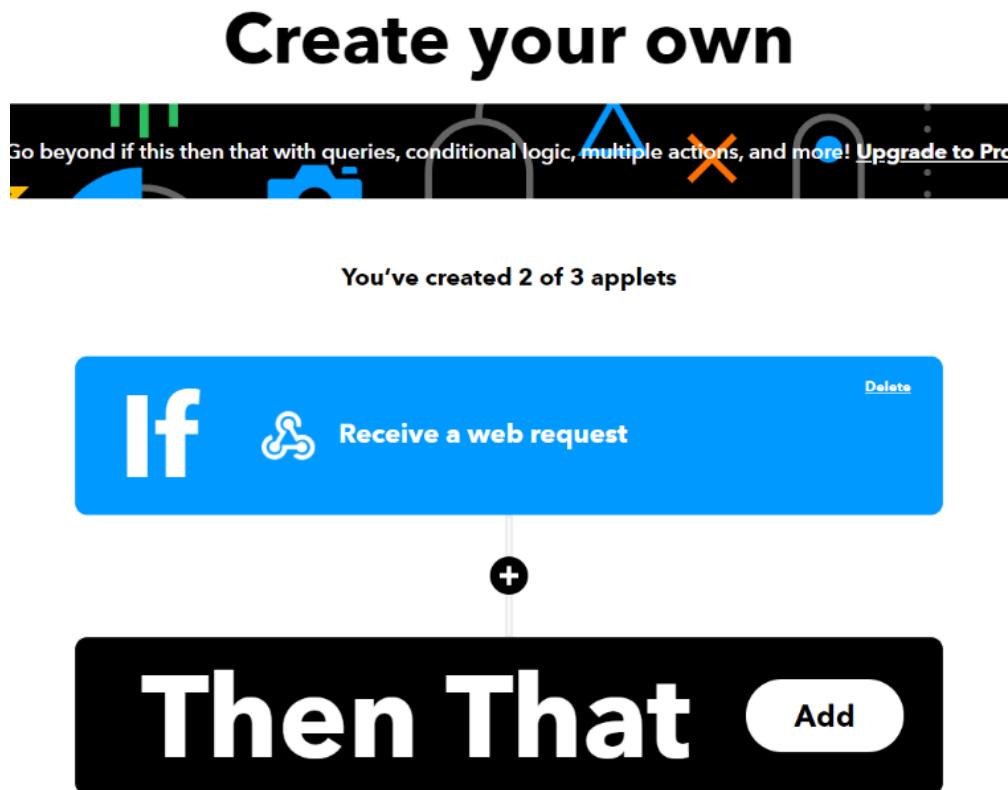
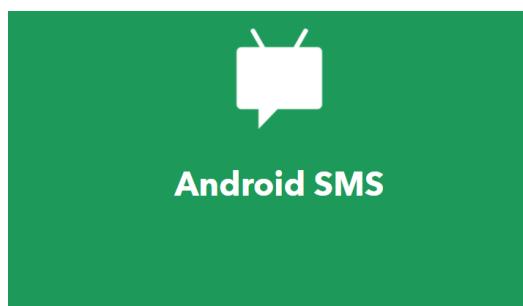


Fig 1.15: Creating THEN THAT statement



Send an SMS
This Action will send an SMS from your Android device to any phone number you specify.

Fig1.16: Select Send SMS

Fill the corresponding details of the alert message and create the action and complete the configuration in IFTTT website.



Fig 1.17: Type the Message and phone number

Now check for the Webhook documentation and make a note of user KEY

A screenshot of the IFTTT Webhook documentation page. It shows a blue icon and the text 'Your key is: XrOJoTMbLRlja[REDACTED]'. Below it, there's a 'Back to service' link. A section titled 'To trigger an Event' provides instructions for triggering the event via POST or GET requests to the URL <https://maker.ifttt.com/trigger/noti/with/key/XrOJoTMbLRljaQdk2vZI->. It also shows an optional JSON body example: { "value1" : "[REDACTED]", "value2" : "[REDACTED]", "value3" : "[REDACTED]" }. A note states that the data is optional and can be passed as query parameters or form variables. It also shows a command line example using curl: curl -X POST https://maker.ifttt.com/trigger/noti/with/key/XrOJoTMbLRljaQdk2vZI-. At the bottom, there's a 'Test It' button and a link to the FAQ.

Fig 1.18: User Key

Similarly we have to do for call option

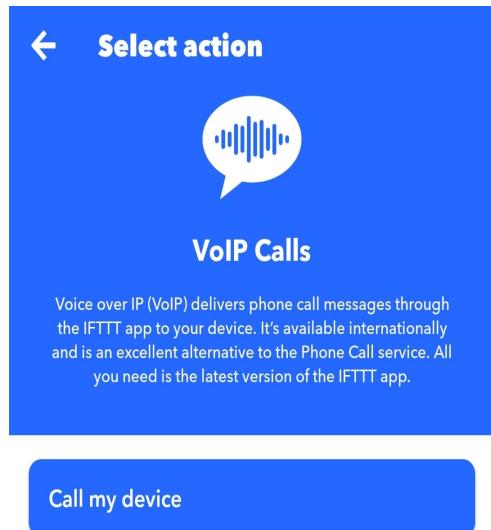


Fig1.19: Select call my device

Then type the voice message you want to send

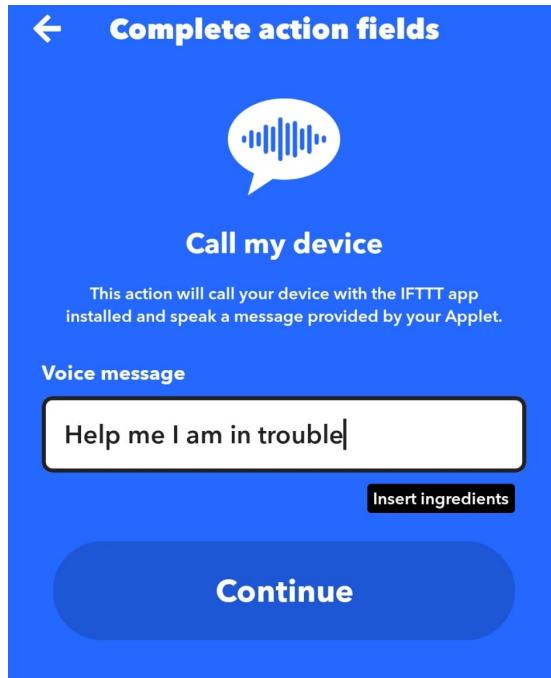
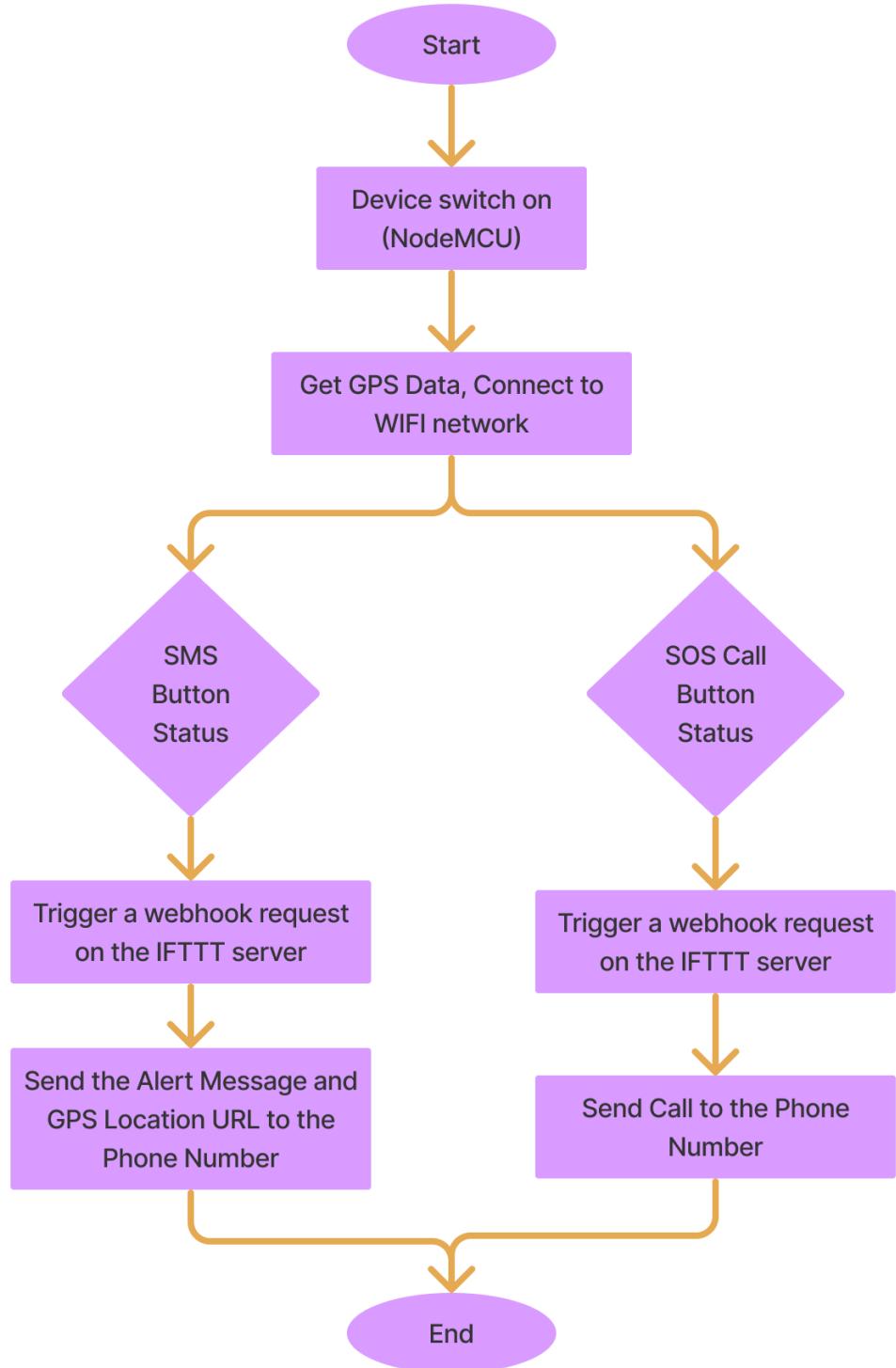


Fig 1.20: Type Voice message

FLOWCHART OF THE PROJECT



Algorithm used in the project:

Step 1: Switch on the NodeMCU.

Step 2: Get the GPS Coordinates and connect to the WIFI network.

Step 3: Check for the status of SMS and SOS Call buttons Status.

Step 4: If the SMS/SOS Call button is clicked then Webhook request is triggered on the IFTTT server.

Step 5: Send the alert message and GPS Coordinates or Send SOS Call to the concerned phone number.

Explanation:

When the device is turned on the device connects to the given WIFI network, the Geolocation API fetches Live GPS Coordinates and send it to NodeMCU. Now check for the status of SMS and SOS Call buttons, if the SMS button is pressed than the Webhook request is triggered on the IFTTT server and the GPS Coordinates are sent to concerned phone number, if SOS Call button is pressed a SOS call will be sent to concerned phone number.

Working:

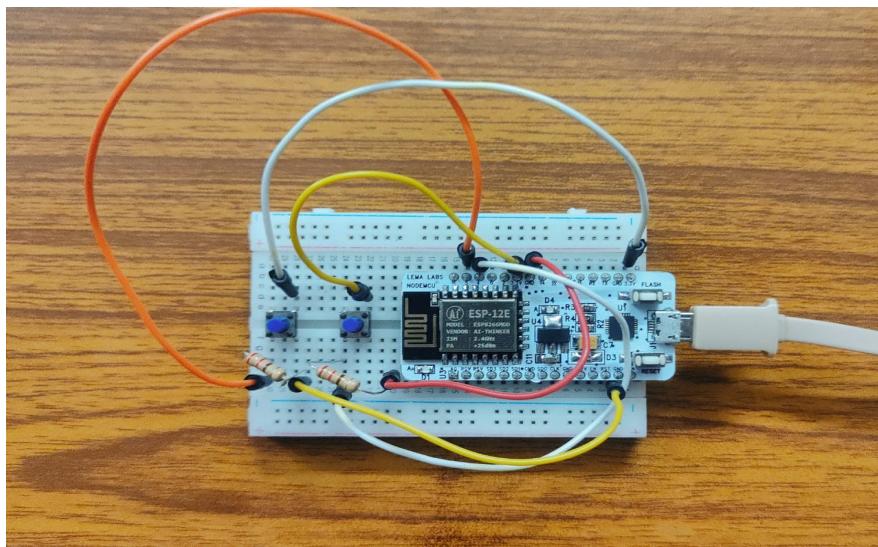
- When the device is turned on using power source, NodeMCU will get connected to the given WIFI Network and GeoLocation API fetches Gps location using HTTP Client library, WiFi Client Secure library, Web server library and SHA1 Fingerprint.
- As soon as SMS button is pressed and Hold, system got activated, NodeMCU will start connecting to IFTTT server and make a Webhook request, along with the Google Map location URL.
- When SOS Call button is pressed and Hold, system got activated, NodeMCU will start connecting to IFTTT server and make a webhook request, send a call to the person.

- As soon as the Webhook request is triggered from NodeMCU, Android message applet gets activated and a message is sent to the mobile number configured in the Android message applet.
- It will also trigger an Emergency Notification alert on the Phone and message sent consists of the GPS Coordinates that was received from the GeoLocation API

RESULT AND DISCUSSION

The project was successfully completed before the due date. The main purpose of this project is to provide safety and security to the women in danger situation. When the SMS button is pressed by women as soon as she feels insecure, then the exact location is sent to the specified contact number in the form of coordinates with the Alert message stating
 "HELP ME
 I'M IN TROUBLE...!
 My Location: google map location url"
 The project is working as per our Objectives.

Hardware Setup of the Project:



The following are the screenshot of the outputs of the project:

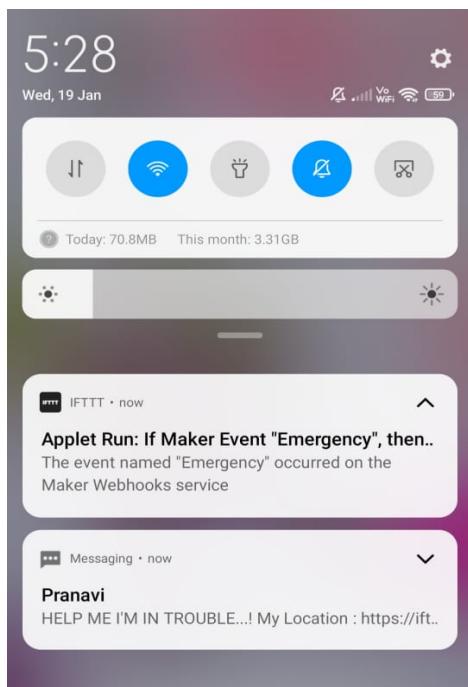


Fig 1.21: Notification sent by IFTTT server to IFTTT application in the phone

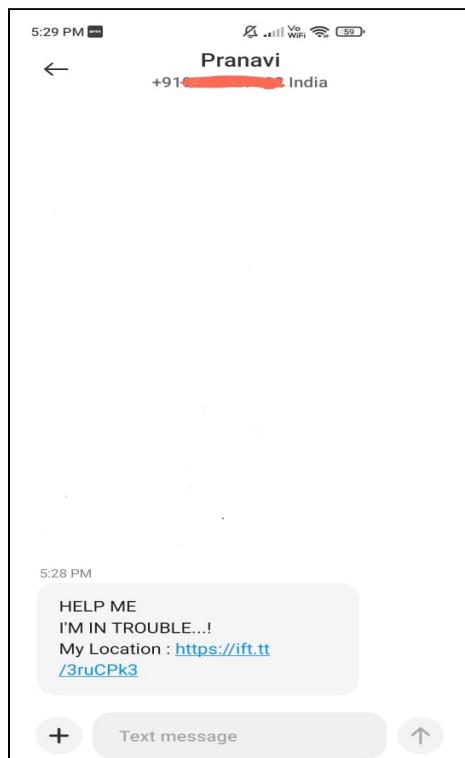


Fig 1.22: Alert message with GPS Coordinates sent to concerned person



Fig 1.23: User tracking location in map using the GPS Coordinates

Similarly when SOS Call button is pressed

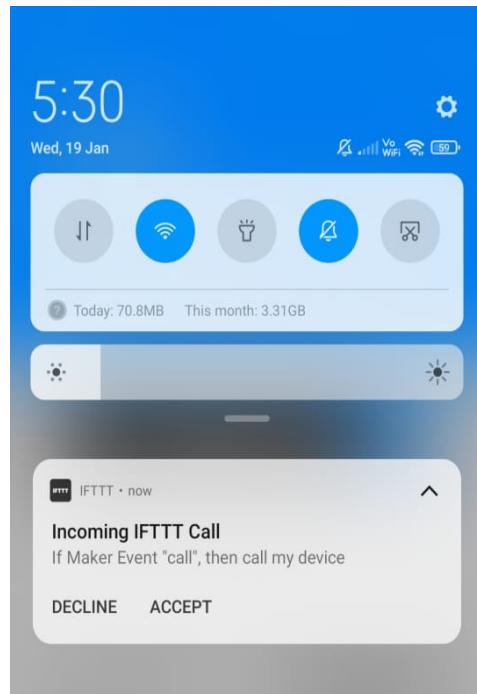


Fig1.24: Sent a Call to the concerned person

ADVANTAGES AND APPLICATIONS

Advantages of the project :

- Can be used as a legal evidence of crime with exact location information for prosecution.
- It is affordable not so expensive
- Comfortable and it is easy to use with just one press on sos button. • The family member can locate women/any person in danger and hence take the necessary action to rescue the women from danger. • It is portable and Wifi- Enabled.

Applications of the Project :

- Can be used in vehicle tracking & safety system.
- Can be used for the safety of women to easily track a woman's location with the help of coordinates sent.
- Can be used for the safety of women.
- Can be used for the safety of children.
- Can be used for the safety of elderly aged people.
- Can be used for the safety of physically challenged people.
- Can be used in Security applications.

CONCLUSION AND FUTURE SCOPE

Conclusion:

The whole idea of the project is to help a needy woman by using her location with the help of a device made up of NodeMCU interfaced with GPS. The proposed system will help women to deal with the critical issues faced by them in their day- to- day life and therefore it will also help to reduce the crime rates against women. By sending location it is easier to help a needy woman. All the applications and devices built using new technologies may be difficult to operate for women in rural areas. This system is easy to use and also affordable.

Hence this system can overcome the fear that scares every woman in the country about her safety and security. The Internet of things has revolutionized the whole security system by simplifying things. It is helpful in providing secure environment for the females. The entire project was quite cost effective and well within our budget of ₹500. The total power consumption of the project is around 5V. We are very much satisfied with our results, but there is always room for improvement.

Future Scope:

For the better functionality of the product new versions of the device can be introduced by adapting new technologies. The present system is working efficiently, but we can still increase the functionality of the device by using various other modules without affecting the present system. Functionalities like voice detection, camera, screaming alarm can be added and some kind of tools for defense like Stun gun (shock generator) can be implemented. Voice detector can be added to give the voice command which can be priorly feed in NodeMCU to indicate emergency /danger. To send the alert message to the persons near to the emergency location. The device can be enhanced much more in the future by using highly compact modules.

APPENDIX

Project Code:

```
#include <ESP8266WiFi.h>
#include <WiFiClientSecure.h>
#include <ESP8266WebServer.h>
#include <ESP8266HTTPClient.h>

const char *ssid = "ENTER YOUR WIFI NAME";
const char *password = "ENTER YOUR WIFI PASSWORD";

const char *host = "ENTER HOST NAME";
const int httpsPort = 443;

const char hos[]="maker.ifttt.com";
const char privateKey[]="ENTER YOUR IFTTT AUTHORISED KEY";
const char event[]="ENTER YOUR EVENT NAME";
const char even[]="ENTER YOUR EVENT NAME";
//HTTPS= 443 and HTTP = 80

//SHA1 finger print of certificate use web browser to view and copy
const char fingerprint[] PROGMEM = "ENTER THE WEBSITE SHA1
FINGERPRINT";

void setup() {
  pinMode(D1, INPUT);
  pinMode(D2,INPUT);
  delay(1000);
  Serial.begin(115200);
```

```

WiFi.mode(WIFI_OFF);      //Prevents reconnection issue (taking too long
to connect)
delay(1000);
WiFi.mode(WIFI_STA);     //Only Station No AP, This line hides the viewing
of ESP as wifi hotspot

WiFi.begin(ssid, password); //Connect to your WiFi router
Serial.println("");


Serial.print("Connecting");
// Wait for connection
while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
}

//If connection successful show IP address in serial monitor
Serial.println("");
Serial.print("Connected to ");
Serial.println(ssid);
Serial.print("IP address: ");
Serial.println(WiFi.localIP()); //IP address assigned to your ESP
}

```

```

void loop() {
  WiFiClientSecure httpsClient; //Declare object of class WiFiClient

  Serial.println(host);

  Serial.printf("Using fingerprint '%s'\n", fingerprint);
  httpsClient.setFingerprint(fingerprint);
  httpsClient.setTimeout(15000); // 15 Seconds

```

```

delay(1000);

Serial.print("HTTPS Connecting");
int r=0; //retry counter
while((!httpsClient.connect(host, httpsPort)) && (r < 30)){
    delay(100);
    Serial.print(".");
    r++;
}
if(r==30) {
    Serial.println("Connection failed");
}
else {
    Serial.println("Connected to web");
}

String ADCData, getData, Link;
//String to interger conversion

//GET Data
Link = "/json/?apikey=ENTER YOUR API KEY";

Serial.print("requesting URL: ");
Serial.println(host+Link);

httpsClient.print(String("GET ") + Link + " HTTP/1.1\r\n" +
    "Host: " + host + "\r\n" +
    "Connection: close\r\n\r\n");

Serial.println("request sent");

while (httpsClient.connected()) {
    String line = httpsClient.readStringUntil('\n');

```

```

if (line == "\r") {
    Serial.println("headers received");
    break;
}

Serial.println("reply was:");
Serial.println("=====");
String line,lat,lon;
int n,p,x,y;

while(httpsClient.available()){
    line = httpsClient.readStringUntil('\n'); //Read Line by Line
    Serial.println(line);
    Serial.println("");
    n = line.indexOf("latitude");
    p = n+10;
    Serial.print("latitute: ");
    lat = line.substring(p,p+7);
    Serial.println(line.substring(p,p+7));
    Serial.print("longitude: ");
    x = line.indexOf("longitude");
    y = x+11;
    lon = line.substring(y,y+7);
    Serial.println(line.substring(y,y+7));

}

WiFiClient client;

if (!client.connect(hos,80)){
    Serial.println("connecting");
    Serial.println("failed");
}

```

```

delay(2000);
return;
}//END IF

//Create the URL that will be used to fire the trigger
//      /trigger/{event}//with/key/{privateKey}
String url = "/trigger/";
    url = url + event + "/with/key/" + privateKey
+/?value1)+"http://maps.google.com/maps?q=loc:"+lat+","+lon;

String ui = "/trigger/";
    ui = ui + even + "/with/key/" + privateKey;
// Make a HTTP request

String http = "GET ";
    http = http + url + " HTTP/1.1\r\n";
    http = http + "Host: " + hos + "\r\n";
    http = http + "connection: close\r\n\r\n";

String hk = "GET ";
hk = hk + ui + " HTTP/1.1\r\n";
hk = hk + "Host: " + hos + "\r\n";
hk = hk + "connection: close\r\n\r\n"; //close the connection after the
request send

// sensd HTTP request to the server to fire the trigger
byte val = digitalRead(D1);
byte lav = digitalRead(D2);
if(val==HIGH){
    client.print(hk);

}

if(lav==HIGH){

}

```

```
client.print(http);
}

client.stop();

//Disconnect from the server

Serial.println("SMS has sent");

//END if (digital >= TRIGGER)

Serial.println("=====");
Serial.println("closing connection");

delay(2000); //GET Data at every 2 seconds
}
```