Project Title: Smart Stick for Blind People

Team 6 Members

1. RUGEMA EMMANUEL 220018751
2. GEOFREY TWESIGYE 220014931

Project Supervisor

1. Prof KAYALVIZHI JAYAVEL

Introduction

Technology has indeed revolutionized the world. It has been integrated into all fields and has made human lives more efficient and productive. One of the beneficiaries of technology integration is persons with disabilities where visually impaired persons are a part of Even with many technological developments still, a lot of blind persons use the conventional and archaic way of navigating, this makes them have little contact with the surrounding and has to find other means of navigating to ensure their safety, e.g., bringing a family member or friend. Generally we observe that white cane is the best friend of visually impaired person. But many a times this cane is not useful. In an unfamiliar surrounding visually impaired person might get confused. So this restricts their mobility. This makes them dependent on others. Their inability to move freely opposes them from interacting with other humans and limiting them from joining social activities.

Main Objectives

The aim of this project was to the navigation of visually impaired persons using the latest IoT technologies.The specific objectives were:

1. The ultrasonic sensor is used to detect the presence of obstacle and calculates the distance between the source and destination.
2. Light sensor is used to detect the presence or absence of light.
3. Water sensor is used to detect the presence of water.
4. Anti-theft protection.
5. Warning through voice and vibration

Requirements

The following are needed in order to undertake the project:

1. **Hardware**

a). **Ultrasonic Distance Sensor - HC-SR04**

This is the HC-SR04 ultrasonic distance sensor. This economical sensor provides 2cm to 400cm of non-contact measurement functionality with a ranging accuracy that can reach up to 3mm. Each HC-SR04 module includes an ultrasonic transmitter, a receiver and a control circuit.

b**). LED**

A light-emitting diode (LED) is a semiconductor device that emits light when an electric current is passed through it.

c). **Jumper Wires**

Jumper wires are simply wires that have connector pins at each end, allowing them to be used to connect two points to each other without soldering.

d). **Buzzer**

The buzzer consists of an outside case with two pins to attach it to power and ground. When current is applied to the buzzer it causes the ceramic disk to contract or expand. Changing the This then causes the surrounding disc to vibrate. That's the sound that you hear.

e). **Moisture Sensor**

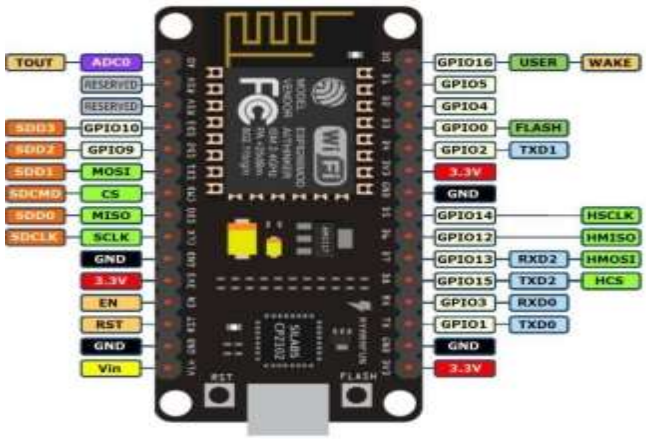
This soil moisture sensor can be used to detect moisture. When the soil is dry, the sensor's output analog value will decrease, while moist soil will result in an increased analog value output. A key application for this sensor is for an automatic watering device. The sensor can sense whether your plant is thirsty or not, preventing over- or under-watering.

The sensor's surface is made from metal to extend its life. Insert it into the soil, then read it using an AD converter.

f).Stick

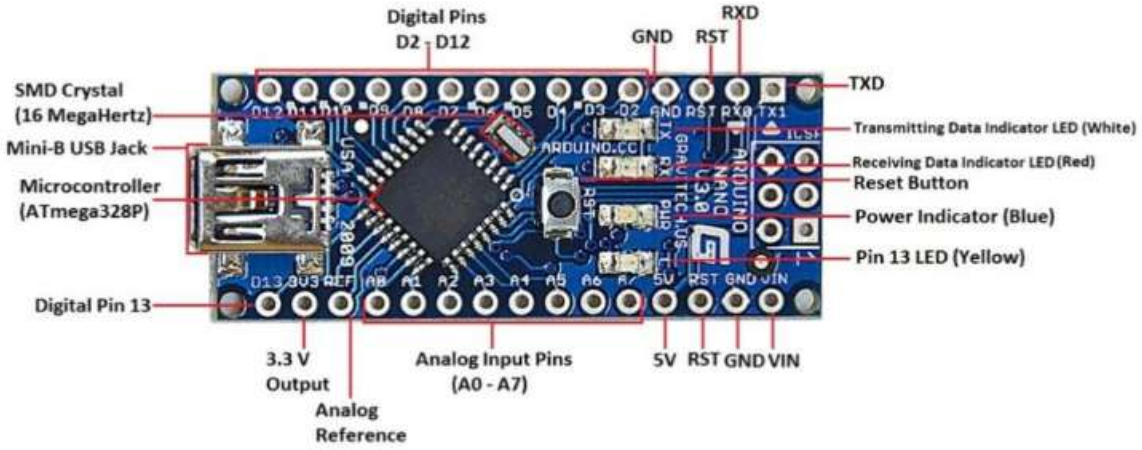
The white cane helps people who are blind or severely visually impaired know when there are tripping hazards such as cracks, poles, etc. The cane is swept from side to side to clear one's path from these and other obstacles. A blind person is guided by someone else by holding on to their arm.

g). NodeMCU Microcontroller with a USB cable



NodeMCU is an open source platform based on ESP8266 which can connect objects and let data transfer using the Wi-Fi protocol. In addition, by providing some of the most important features of microcontrollers such as GPIO, PWM, ADC, and etc, it can solve many of the project's needs alone.

h). Arduino Nano



Arduinonano, Microcontroller Arduino can control the environment by receiving input signals (Digital/Analog) and can the small size of obstacles on ground, IR sensor will send the signal to the Arduino, as result it will send a voice instruction for small obstacle available. And at the same time it will enable the buzzer for informing the blind person about presence of obstacles on ground.

2. SOFTWARE

a. Arduino IDE

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

b. Blynk

Both the Blynk server and Blynk library are open source, while the Blynk app is available free for iOS and Android. Blynk is a new platform that allows you to quickly build interfaces for controlling and monitoring your hardware projects from your iOS and Android device. After downloading the Blynk app, you can create a project dashboard and arrange buttons, sliders, graphs, and other widgets onto the screen.

Hardware Connection

The above hardware components will be connected as follows:

a. Buzzer

Attach the other ends of the wires to the buzzer. Connect each wire to one of the buzzer's terminals and wrap electrical tape around the bare wires to keep them from touching. Test the circuit by placing the battery in the battery holder. The buzzer should begin sounding as soon as you install the battery.

b. Connection to LED

A node of the LED is connected to the D6 pin of the NodeMCU, the cathode of the LED is connected with the one terminal of the resistor and another terminal of the resistor is connected to the ground pin.

c. Connections to Moisture sensor

The wiring connections for the temperature sensor are made as follows:

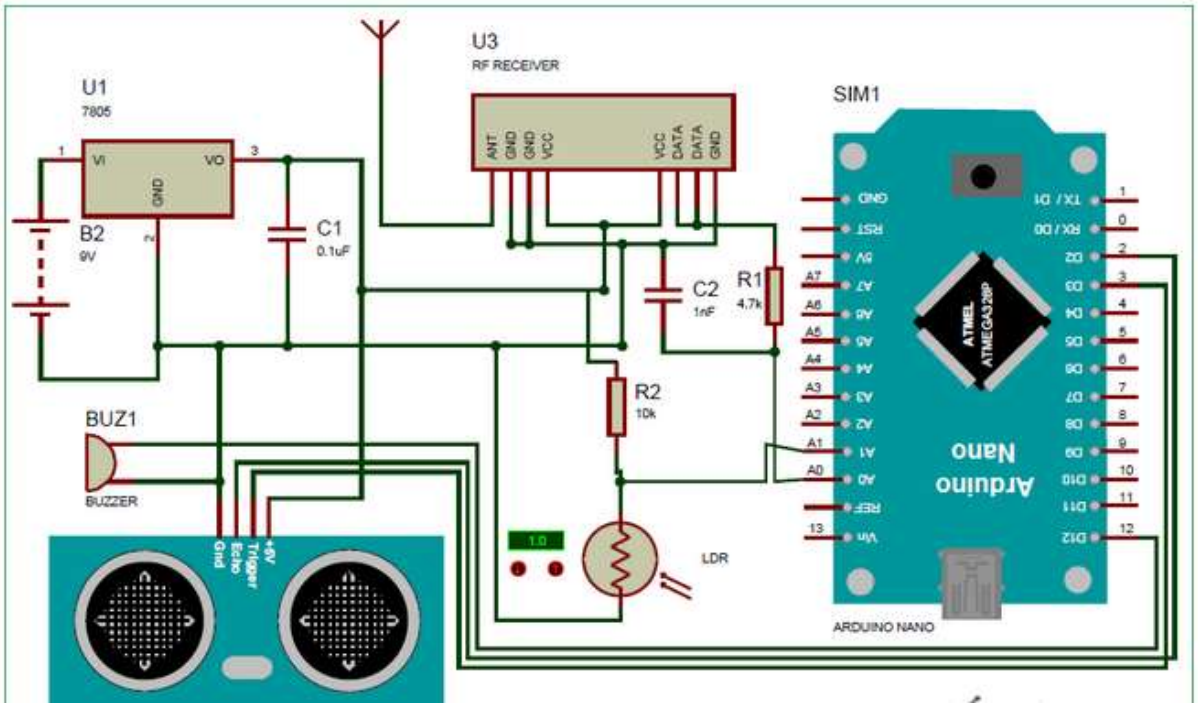
• Pin 1 of the Moisture sensor goes into +3v of the NodeMCU.

• Pin 2 of the Moisture sensor goes into Digital Pin D3 of the NodeMCU.

• Pin 3 of the Moisture sensor goes into Ground Pin (GND) of the NodeMCU

After all the components have been correctly connects the USB can then be connected to the computer for uploading of code.

Circuit Diagram or Block Diagram



Software Set Up

The following steps will be followed in setting up the software

1. Arduino IDE installation

To download the software go to the Arduino site: https://www.arduino.cc

a. Click on software click on Windows, Mac or Linux based on your Operating System.
b. You can donate if you want or just download.
c. When this is done, you will simply need to continue the steps to download it to your computer.

2. Arduino IDE setup

After downloading the Arduino IDE navigate to:

a. File tab and then click on Preferences.

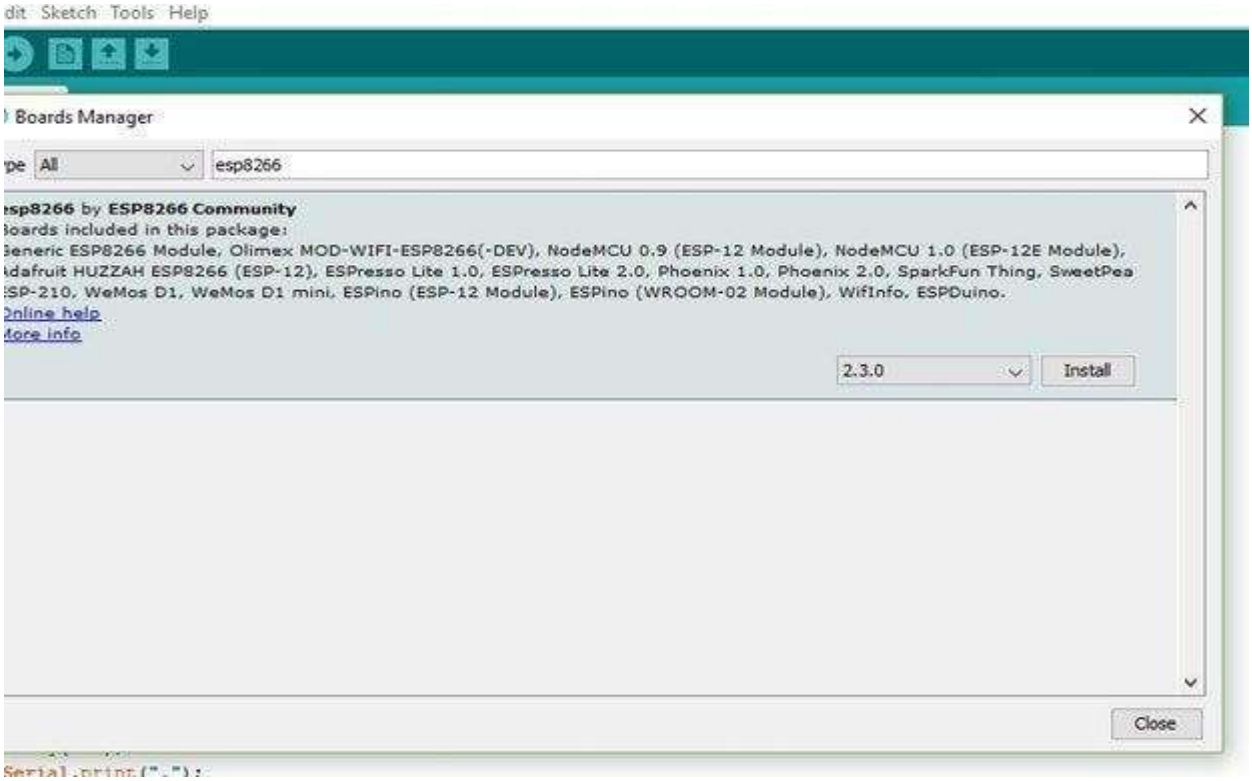
b. In the additional Boards Manager URLs add the following link

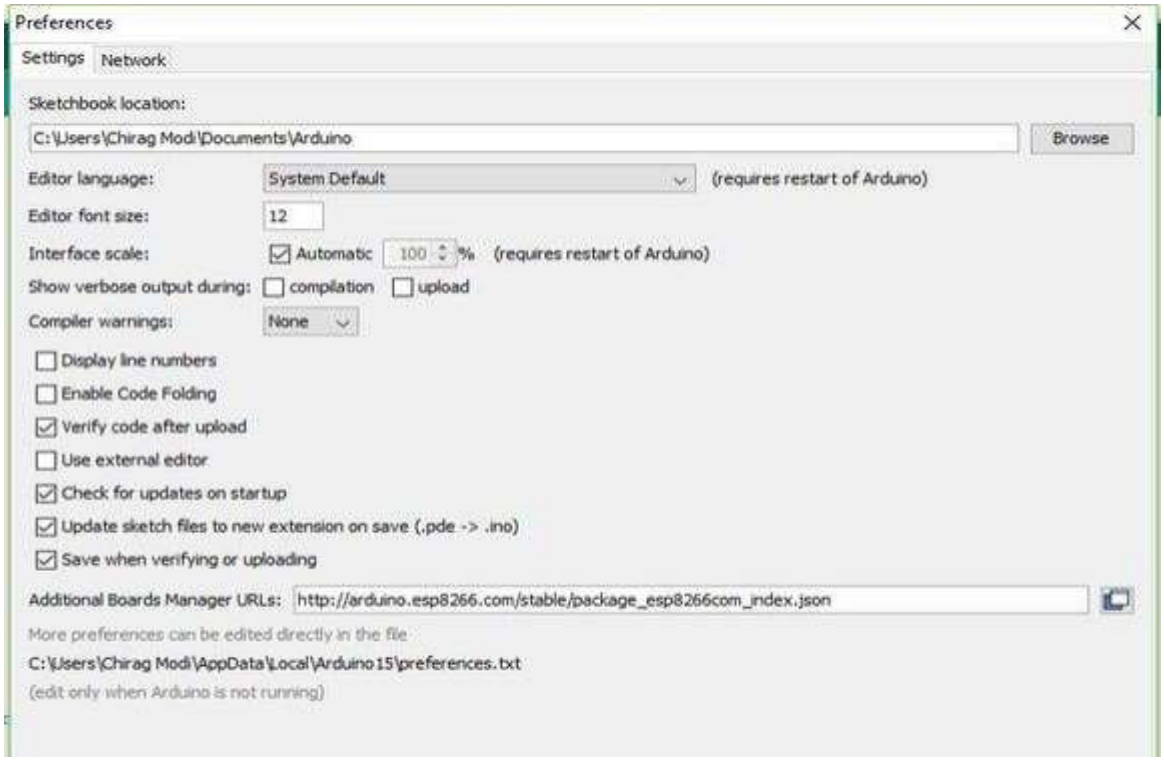
(http://arduino.esp8266.com/stable/package_esp8266com_index.json)

c. Click OK and then navigate to Tools - Boards - Boards Manager

d. In the search field type esp8266 > click the esp8266 by ESP8266 Community

e. Click Install



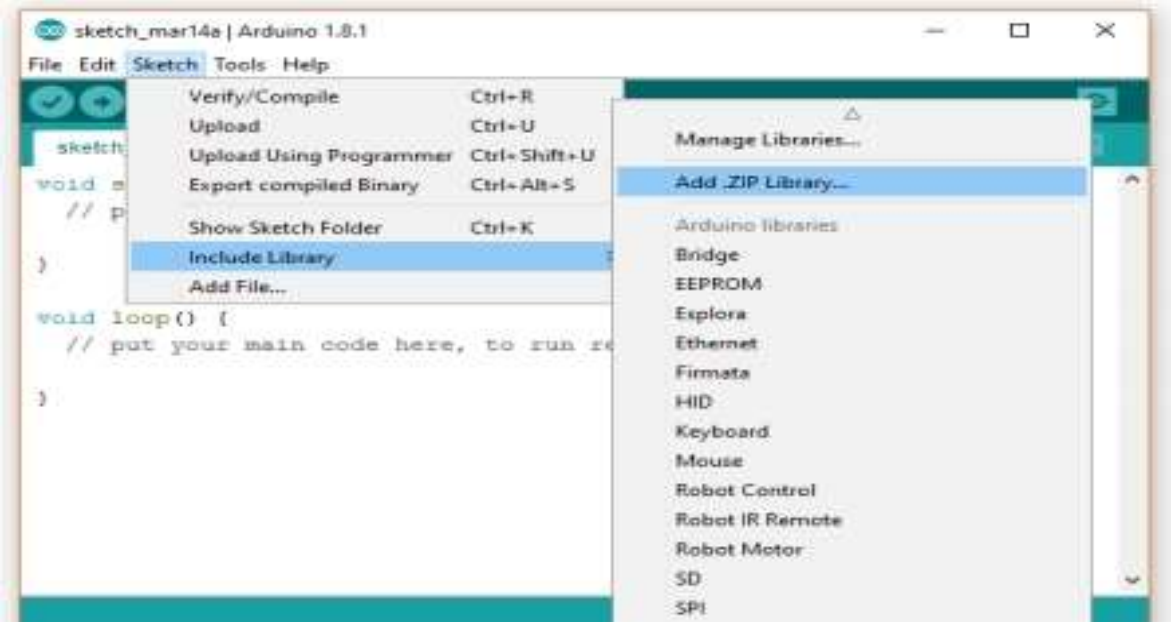


3. Installation of Packages

You will need to install additional libraries for the code to work

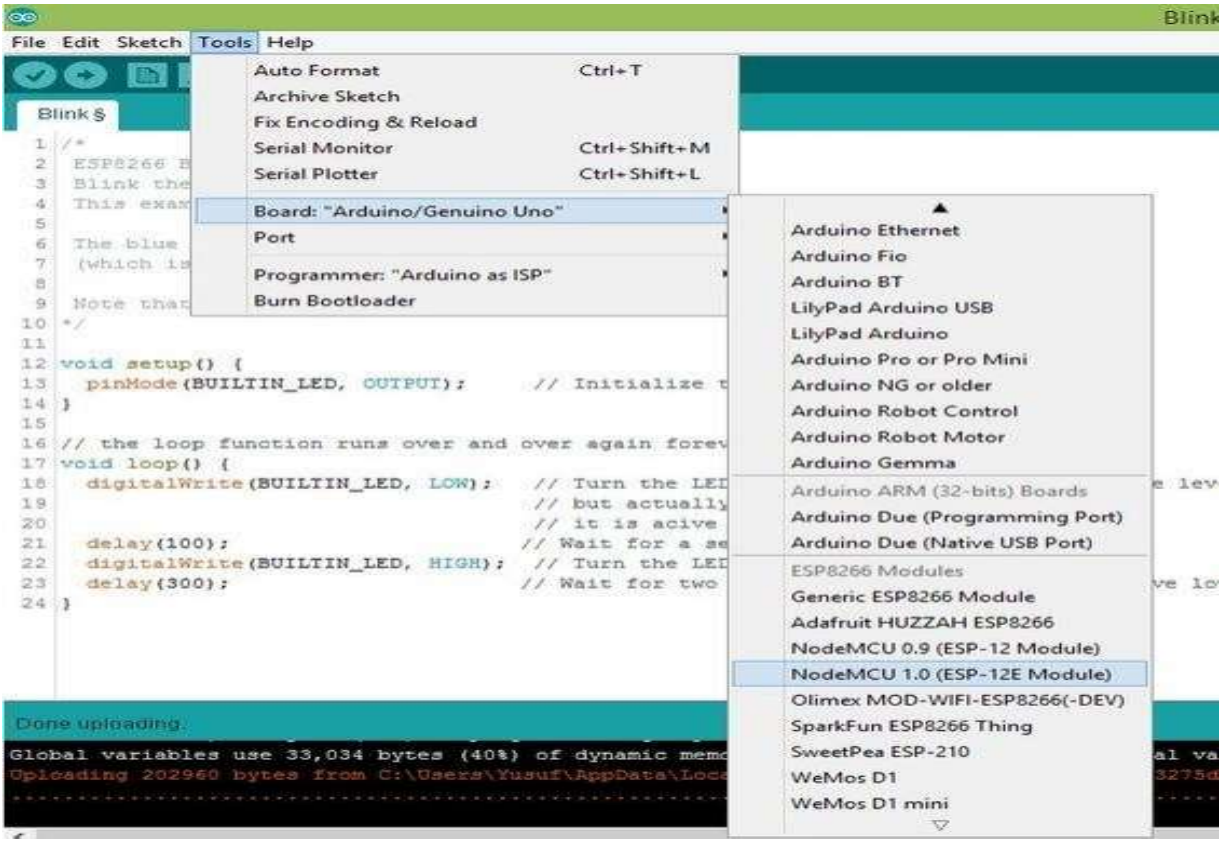
a. Go to Sketch > Include Library > Manage Libraries.

Search for BlynkSimpleEsp8266.h, FirebaseArduino.h, You can also search for the zip file of the libraries online then install using add a .zip Library link in include library



4. Board and Port Selection

After all libraries have been installed you will need to select the ESP8266 board and COM port.



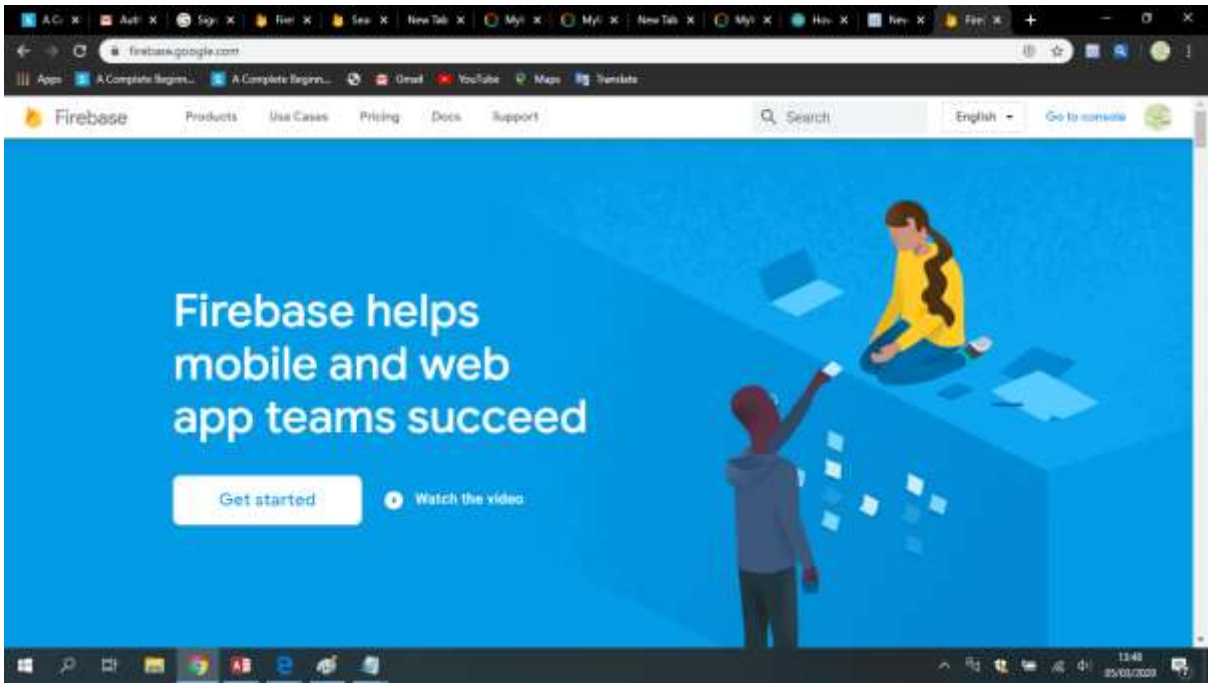
5. Uploading and compilation of code

After all has been set up the next step is to compile and upload the code to the board

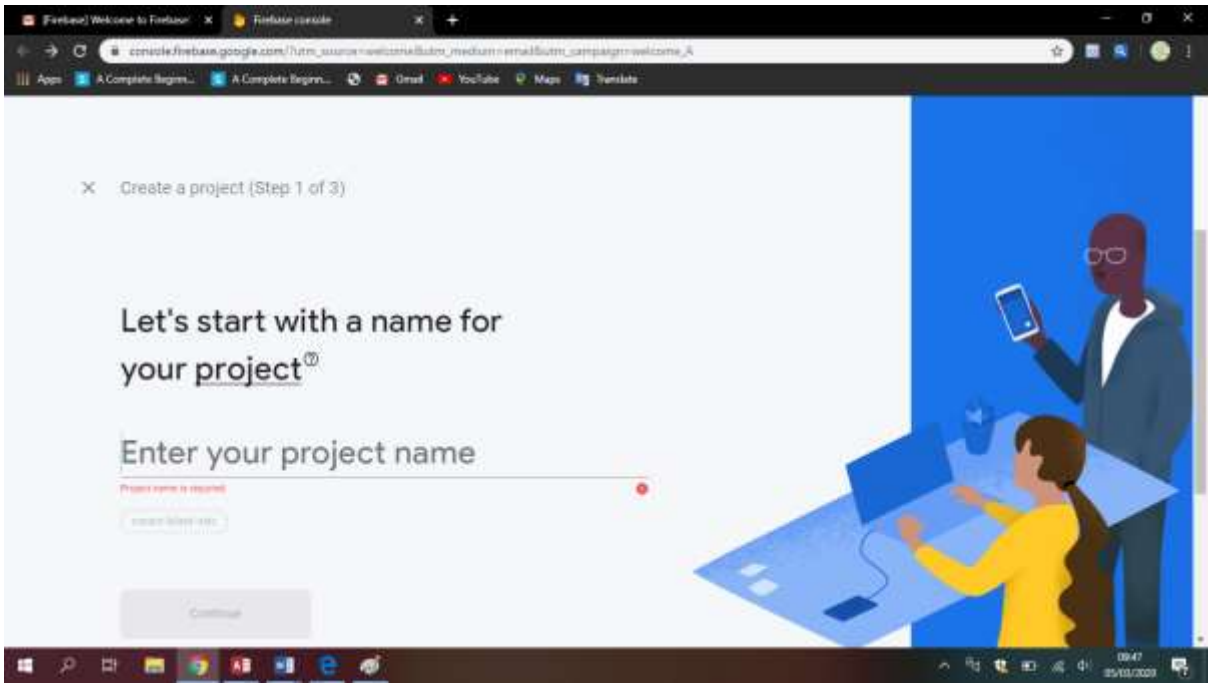
FIREBASE SET UP

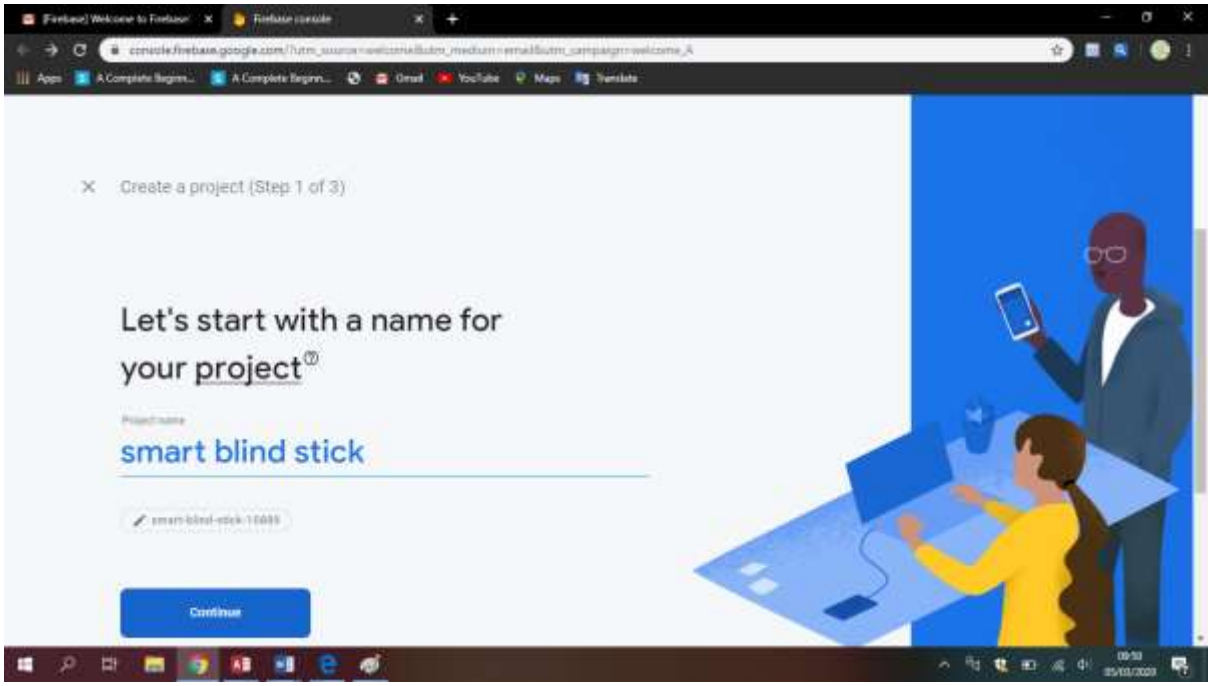
a. Create an account/login

Go to the Firebase website and sign up for an account if you don't already have one. You can log in with a Google account for easy access.

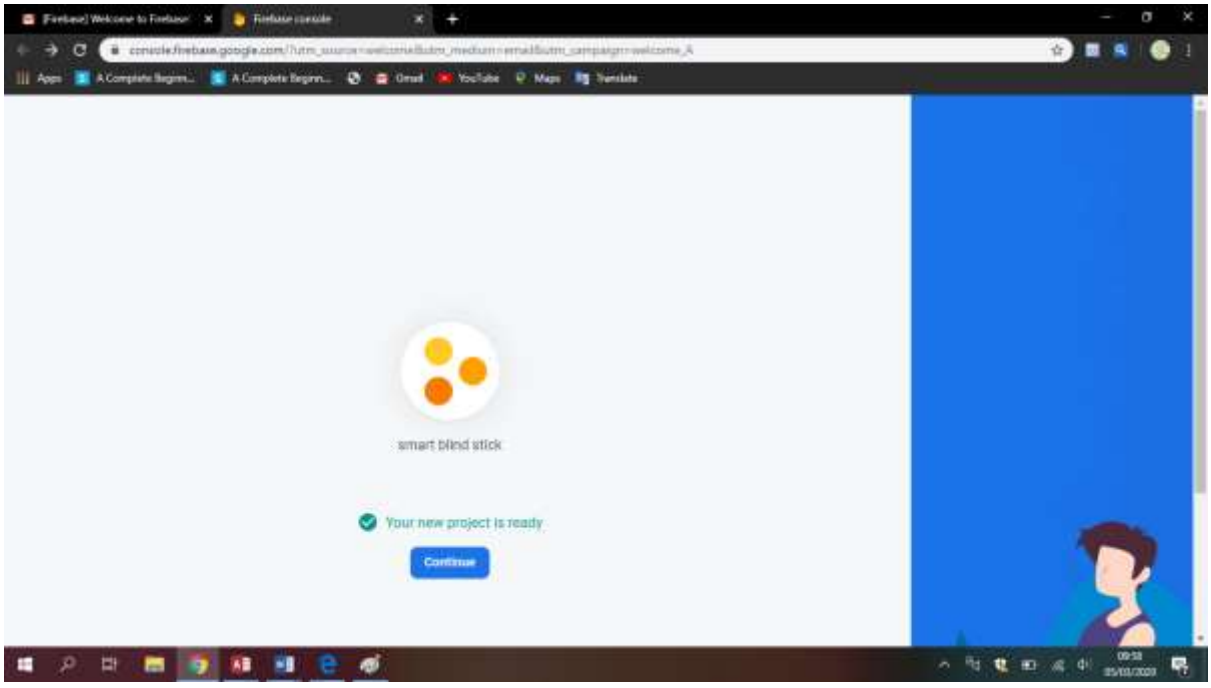


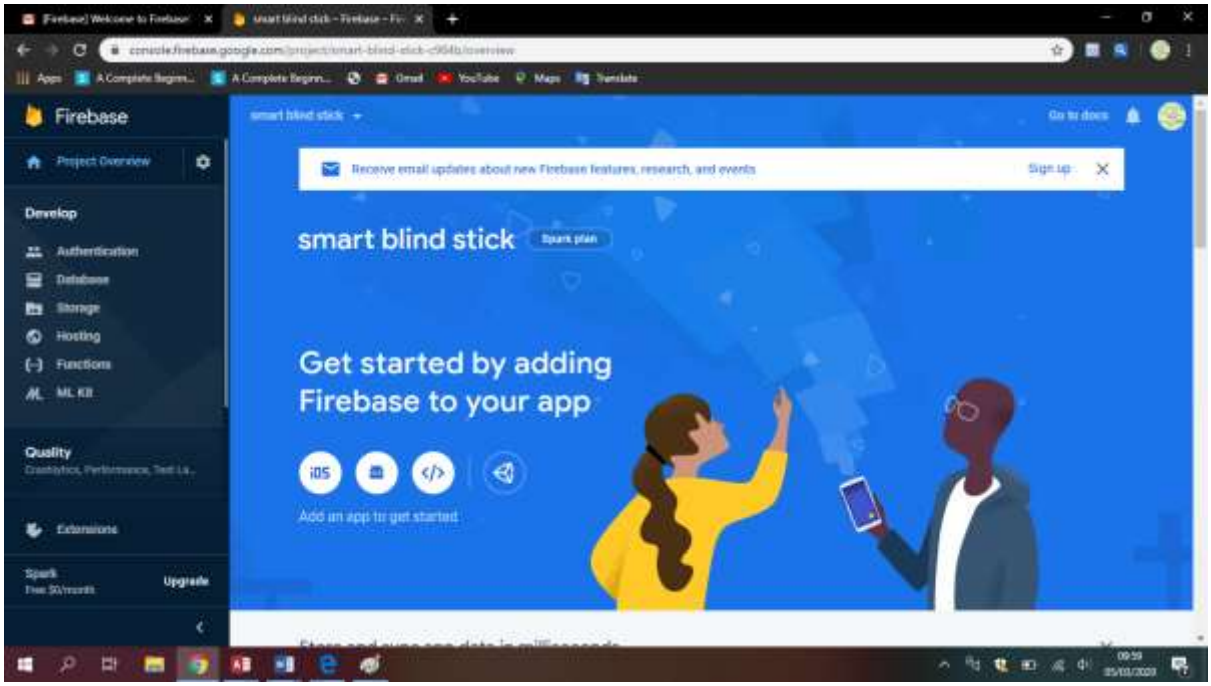
b. Create a new project





When you log in, you should be directed to the Firebase console. You can manage all of your projects here. Go ahead and create a new one by clicking the blue **CREATE NEW**



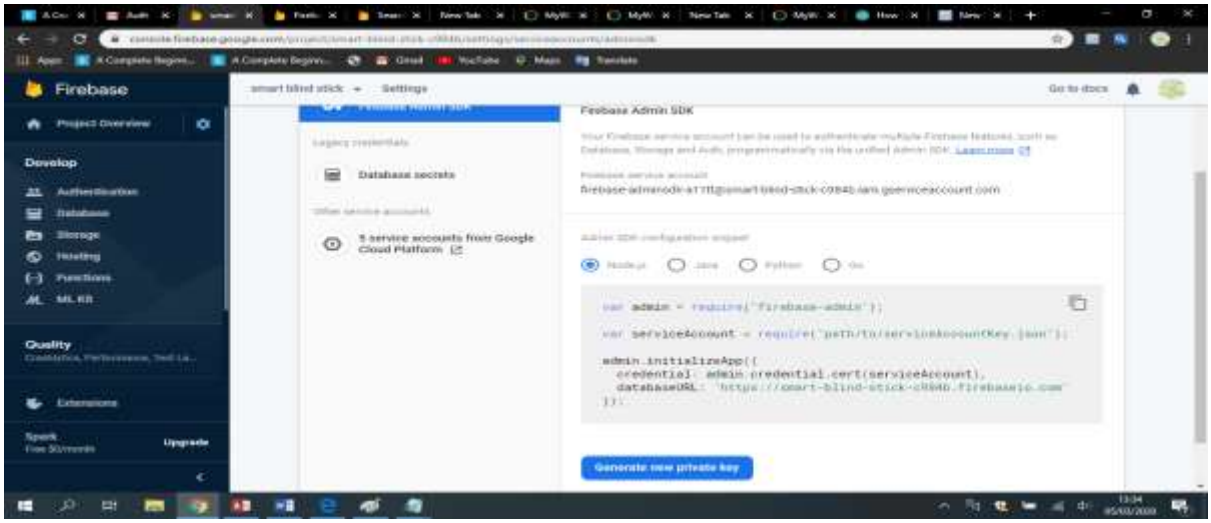


Once you create your project, you'll be redirected to your project's console overview. If you check the URL at the top of your browser, you'll see something like **https://console.firebase.google.com/u/1/project/team6-temp/database/team6-temp/data**, where the part of the URL after **/project/** matches your project's name.

c. Database link

To run the project you will need to get the database link and the secret key. For database link on database and copy the link without https and replace in the following line of code

"smart-blind-stick-c984b.firebaseio.com"

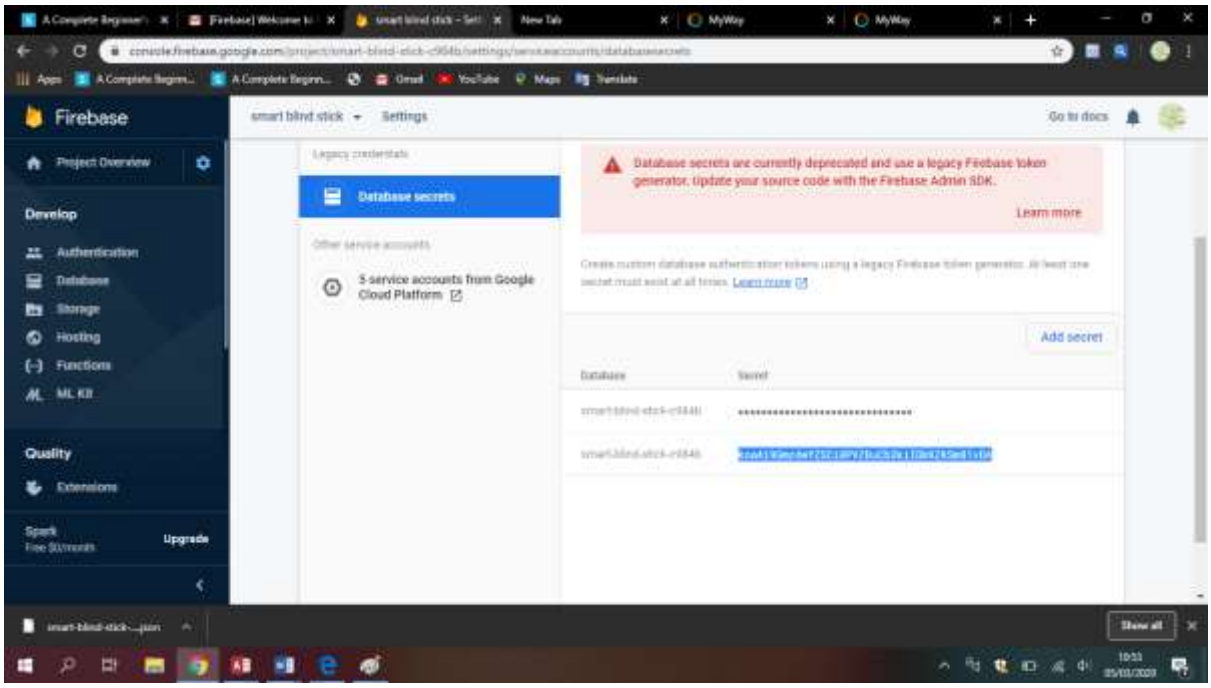


d. Database Secret

For the database secret click on Project overview/user and permissions. Then click on service accounts.

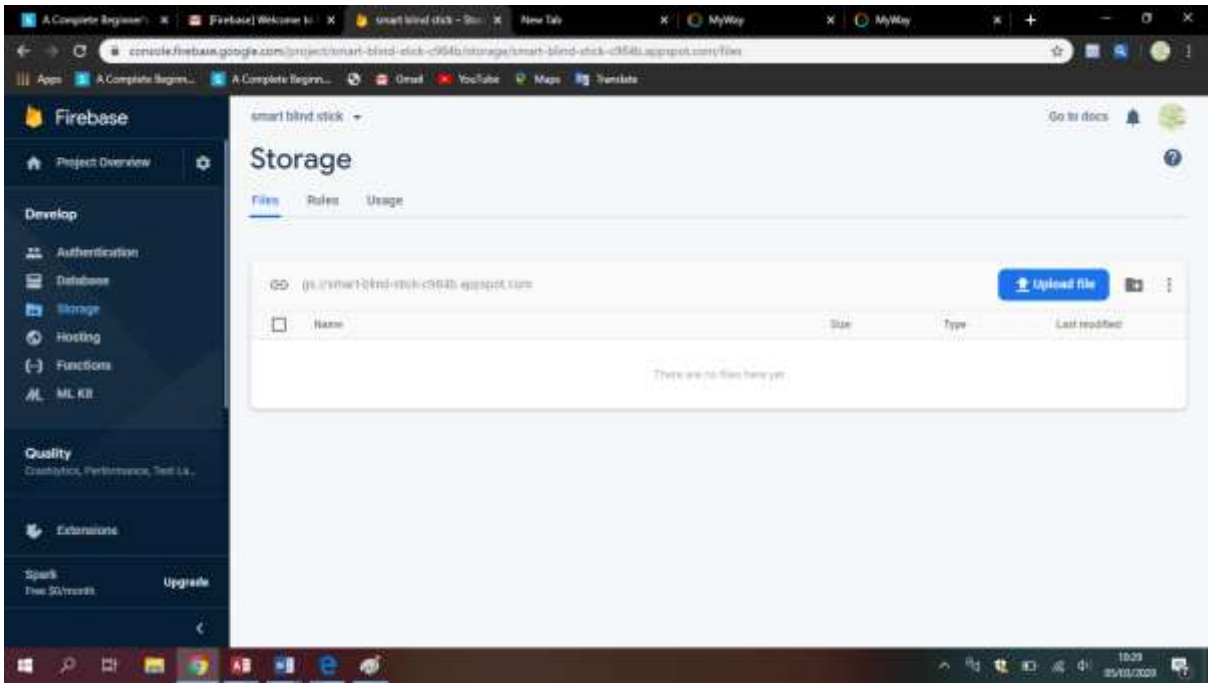
Click database secrete and copy to the following line of code

#define FIREBASE_AUTH "koa4i9Gmz4wYZSCi0PVZBuCb3kiIDkKZKSm81vD4"

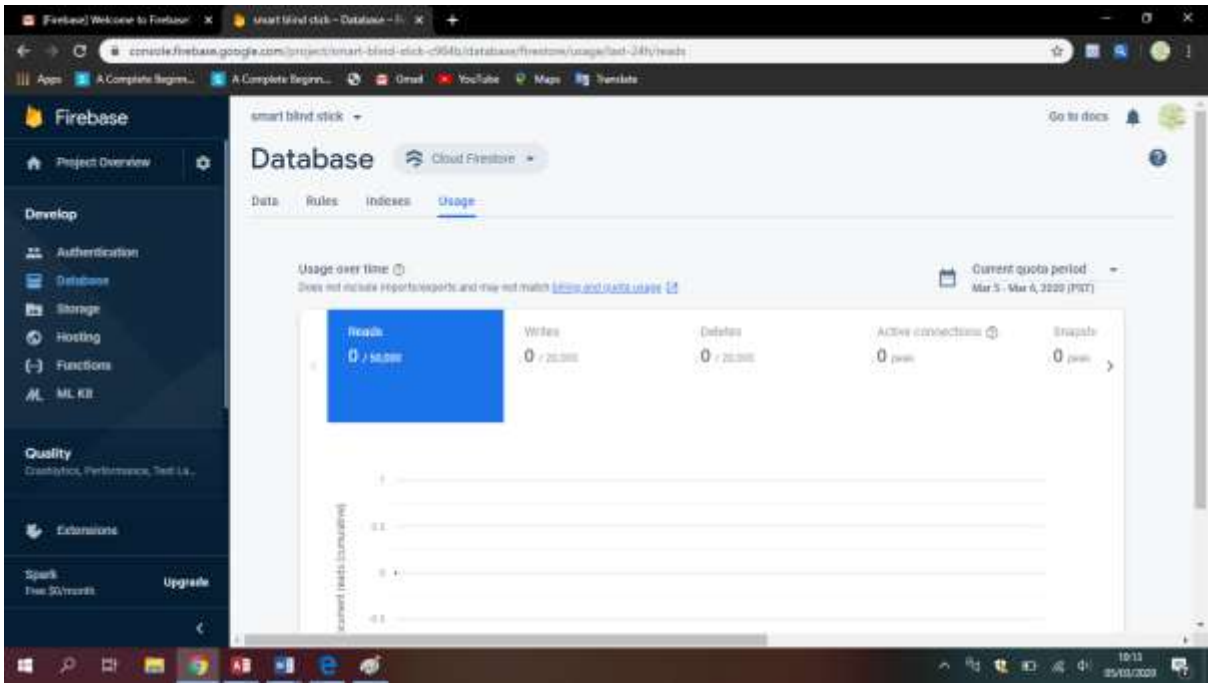


f. Storage View

To view the database click on database and select smart stick blind database



There is no file uploaded, because Ultrasonic sensor is not working, that's why there is no information in database view.



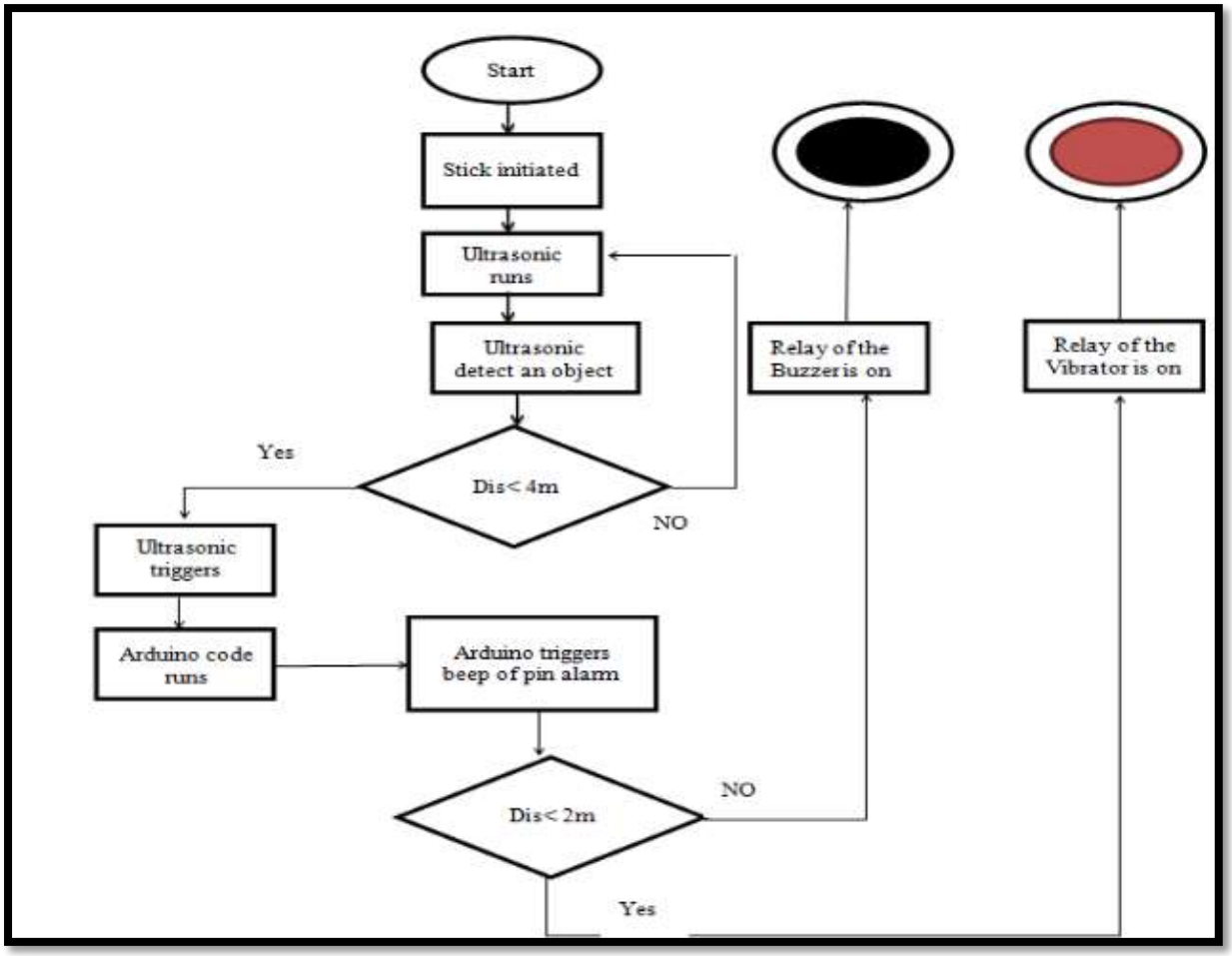
WORKING PRINCIPLE

The system will work as follows:

i).To assist blind or visually impaired people to safely-move among obstacles, holes, ponds and other hurdles faced by them in their daily life. The solution developed is a user-friendly navigational aid for them

ii). The advantage of the system lies in the fact that it can prove to be a very low cost solution to millions of blind people worldwide and the system will makes a time demanding system that monitors the environmental scenario of static and dynamic objects and provides necessary feedback forming navigation more precise, safe and secure.

This can be summarized as per the flow chart below:



IMPLEMENTATION

The system was implemented as per the instructions given above and the following are images from setup and output:

a. Serial output screens

It shows both outputs related to distance and Ultrasonic sensor operation

b. Firebase screen shot

The screen shot shows distance reading as sent to firebase

b. Image of the Circuit

This shows the image of the connected circuit

IMPLEMENTATION CHALLENGES

The device that alerts the user to obstacles in their path is the buzzer. The buzzer is housed in the cane, and is connected to the Arduino. The ultrasonic sensor analyses data from the firebase and it is this data that is sent to the vibration motor in the form of a corresponding NodeMCUesp8266 duty cycle. Depending on the number of pulses, the buzzer receives varying amounts of power, which causes the buzzer to spin at differing speeds. These speeds vary discretely instead of continuously, so that a given range of distances will correspond to one sound intensity. Additionally, each distance will also correspond to a certain delay between buzzers, with greater distances having greater delays.

POTENTIAL PROPOSAL AREAS

This project can be proposed in the following areas

- ✓ For ultrasonic blind walking stick applications
- ✓ For Smart home monitoring applications
- ✓ Distance monitoring using GPS applications

CONCLUSION

The Smart blind stick acts as a basic platform for the coming generation of more aiding devices to help the visually impaired to be more safe. It is effective and afford. It leads to good results in detecting the obstacles lying ahead of the user in a range of four meters, detecting stairs and water pits. This system offers a low-cost, reliable, portable, low power consumption and robust solution for navigation with obvious short response time. Though the system is hard-wired with sensors and other components, it's light in weight. Further aspects of this system can be improved via wireless connectivity between the system components, thus, increasing the range of the ultrasonic sensor and implementing a technology for determining the speed of approaching obstacles. While developing such an empowering solution, visually impaired and blind people in all developing countries were on top of our priorities.

FUTURE WORK

Future work of our project is the system can be supplemented with GPS and GSM module by this we can provide safety and also it helps to track the blind person.

REFERENCES

http://news.cnet.com/8301-17938_105-10302499-1.html/

https://firebase.google.com/

https://iot-guider.com/esp8266-nodemcu/learn-interfacing-ds3231-rtc-module-with-nodemcu/

https://nodemcu.readthedocs.io/en/master/modules/rtctime/

https://www.arduino.cc/en/main/software

CODES

```
#include "NewPing.h" // library for Ultrasonic sensoror
#define BLYNK_PRINT Serial
#include <BlynkSimpleEsp8266.h>  // library for blynk
#define FIREBASE_HOST "smart-blind-stick-c984b.firebaseio.com"//host name
#define FIREBASE_AUTH
"koa4i9Gmz4wYZSCi0PVZBuCb3kiIDkKZKSm81vD4"//secreat key of firebase
#define TRIGGER_PIN D1 // connect trigger pin to D1
#define ECHO_PIN D2 // NewPing setup of pins and maximum distance.
NewPing sonar(TRIGGER_PIN, ECHO_PIN);
char auth[] = "4va1Z0NsDqGjrgkM1U5KkfpunkxDKUUD";  // key of blynk
#define WIFI_SSID "Rugema"
#define WIFI_PASSWORD "ru123457"                    //password of wifi
ssid
BlynkTimer timer;
float duration, distance;
const int buzzer = A0;  // assign  buzzer to A0
//_____ method to send data to blynk_____
void sendSensor()
{
  Blynk.virtualWrite(V0, distance); // write the distance to virtual blynk
  delay (200);
  Blynk.virtualWrite(V1, h);// write the humidity from the moisture sensor
  delay (200);
 }
 //_____end of methos_____

void setup()
{

  Serial.begin(9600); //set baud rate of serial mpnitor
  delay(1000); //pause
  pinMode(buzzer, OUTPUT); // Set buzzer - pin A0 as an output
```

```
  tone(buzzer, 1000); // Send 1KHz sound signal.
   delay(1000);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);  //on the wifi
//try to connect with wifi
  Serial.print("Connecting to ");
  Serial.print(WIFI_SSID); // print on serial monitor ssid name
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print("."); // print dot while networking is not available
    delay(500);
  }
  Serial.println();
  Serial.print("Connected to ");
  Serial.println(WIFI_SSID);
  Serial.print("IP Address is : ");
  Serial.println(WiFi.localIP());        //print local IP address

  Blynk.begin(auth, WIFI_SSID, WIFI_PASSWORD);
  timer.setInterval(1000L, sendSensor);   //Start reading dht sensor
}

void loop()
{
  Blynk.run();
  timer.run();
  // Send ping, get distance in cm
  distance = sonar.ping_cm();
  h = dht.readHumidity();    // Reading temperature or humidity

  // Send results to Serial Monitor
  Serial.print("Distance = ");
  Serial.println(distance);
Serial.print("Humidity: ");  Serial.print(h);  // print on serial monitor humidity
  delay(2000);
```

}

TEAM 6 presenters