IOT BASED HEALTH MONITORING SYSTEM

Team Members:

- Shikhar Gupta
- Kamal Utla
- Sarthak Gupta
- Ishani Bais
- Shalu Pandey

Mentor: Avichal Gupta

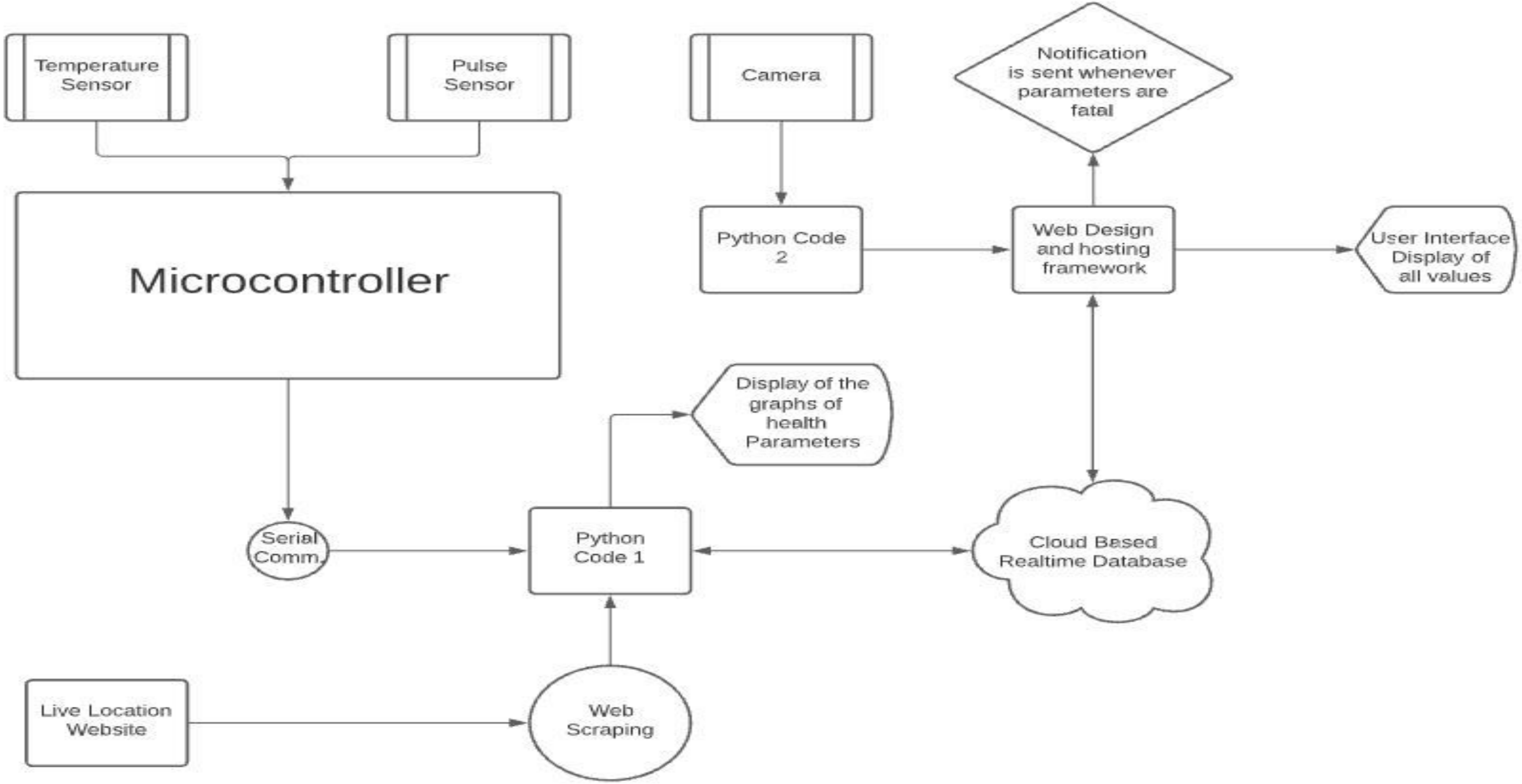
Key Features

- Live Temperature detection of body
- Live Pulse rate detection of body
- Stress level detection using AI
- Live graph of temperature and pulse rate
- Live location of user
- Sending live temperature and pulse to the user through SMS

COMPONENTS

- ARDUINO UNO
- BREADBOARD
- GSM MODULE
- JUMPER WIRE
- TEMPERATURE SENSOR(we are using vibrational sensor)
- PULSE SENSOR(we are using vibrational sensor)
- OLED DISPLAY(we are using python to show the realtime graph)
- WEB APPLICATION
- FIREBASE (REALTIME DATABASE)

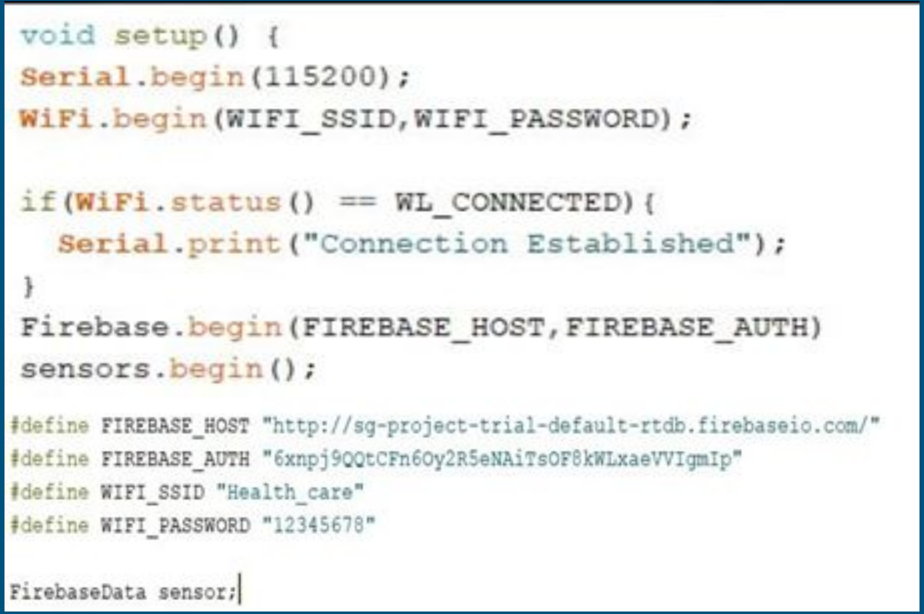
FLOWCHART



Temperature Sensor
Pulse Sensor
Camera
Notification is sent whenever parameters are fatal
Microcontroller
Python Code 2
Web Design and hosting framework
User Interface Display of all values
Display of the graphs of health Parameters
Serial Comm.
Python Code 1
Cloud Based Realtime Database
Live Location Website
Web Scraping

ROADMAP

The Arduino is connected to Firebase. When the sensors read the value of temperature, pulse rate, location (longitude and latitude), the data is pushed in firebase

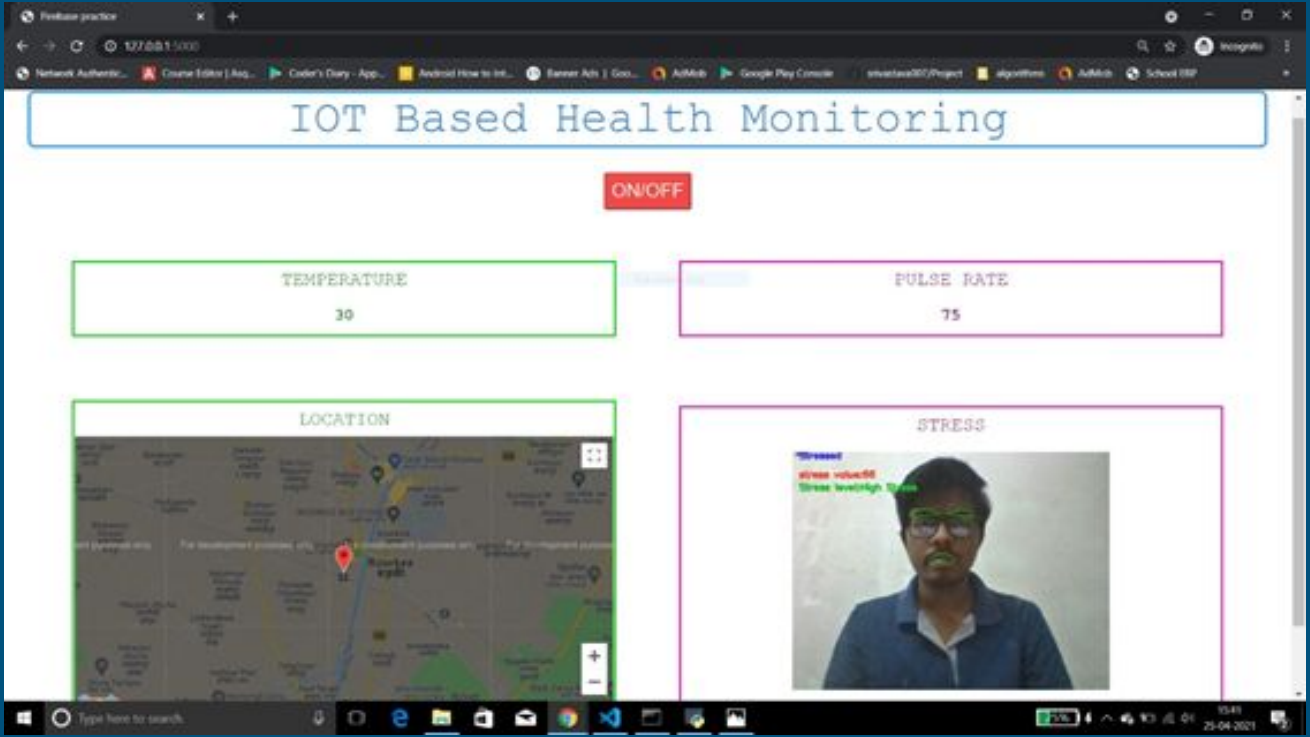


```
void setup() {
Serial.begin(115200);
WiFi.begin(WIFI_SSID,WIFI_PASSWORD);

if(WiFi.status() == WL_CONNECTED){
  Serial.print("Connection Established");
}
Firebase.begin(FIREBASE_HOST,FIREBASE_AUTH)
sensors.begin();
#define FIREBASE_HOST "http://sg-project-trial-default-rtdb.firebaseio.com/"
#define FIREBASE_AUTH "6xnpj9QQtCFn6Oy2R5eNAiTsOF8kWLxaeVVIgmIp"
#define WIFI_SSID "Health_care"
#define WIFI_PASSWORD "12345678"

FirebaseData sensor;
```

The web application receives data from firebase and displays it on the screen. The layout of the webpage is designed with the help of HTML and CSS. It displays four values, i.e., Temperature, Pulse Rate, Stress Levels and live location.



IOT Based Health Monitoring

ON/OFF

TEMPERATURE

30

PULSE RATE

75

LOCATION

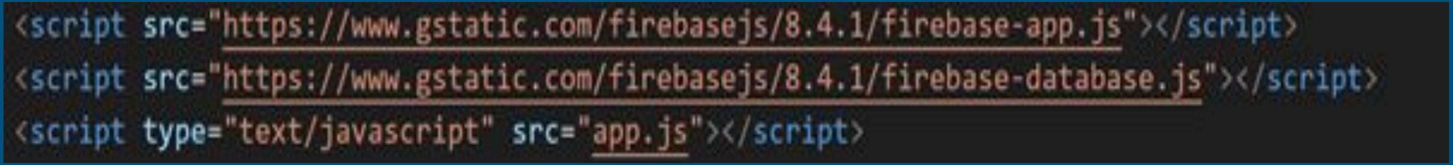
STRESS

Stressed

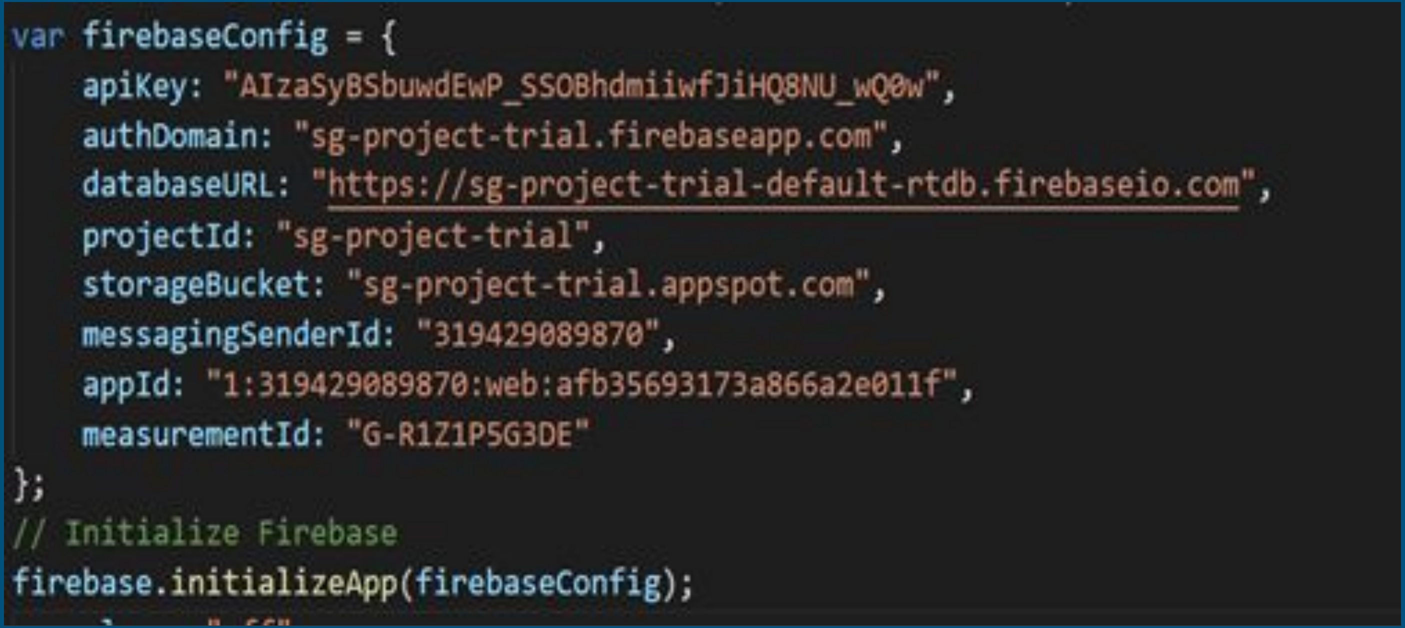
stress value:66

Stress level:High Stress

The Firebase SDK is initialized into the app by adding the given code in the JavaScript file. To read the database from firebase its reference is created in the JavaScript file.

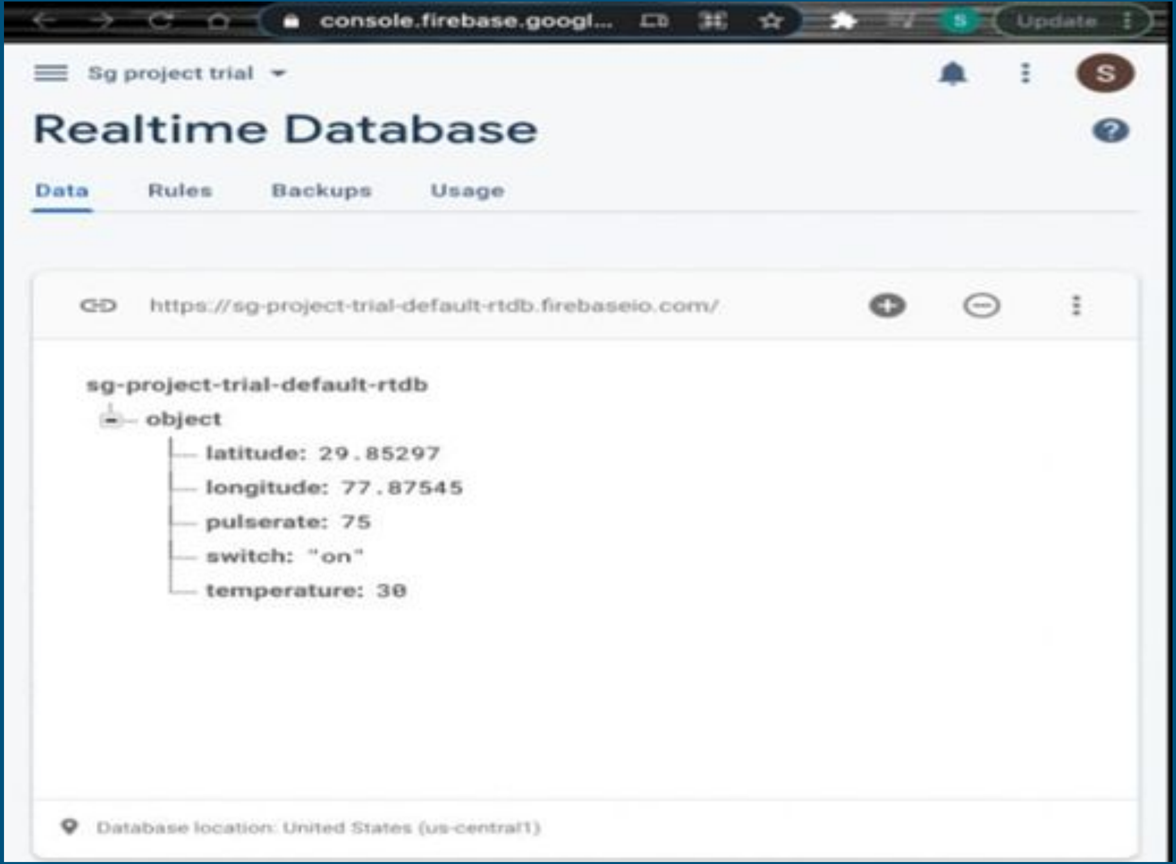


```
<script src="https://www.gstatic.com/firebasejs/8.4.1/firebase-app.js"></script>
<script src="https://www.gstatic.com/firebasejs/8.4.1/firebase-database.js"></script>
<script type="text/javascript" src="app.js"></script>
```



```
var firebaseConfig = {
    apiKey: "AIzaSyBSbuwdEwP_SSOBhdmiiwfJiHQ8NU_wQ0w",
    authDomain: "sg-project-trial.firebaseapp.com",
    databaseURL: "https://sg-project-trial-default-rtdb.firebaseio.com",
    projectId: "sg-project-trial",
    storageBucket: "sg-project-trial.appspot.com",
    messagingSenderId: "319429089870",
    appId: "1:319429089870:web:afb35693173a866a2e011f",
    measurementId: "G-R1Z1P5G3DE"
};
// Initialize Firebase
firebase.initializeApp(firebaseConfig);
```

The following nodes are created on firebase that are to be displayed on the web app- Pulse, Temperature, Latitude and Longitude and switch. With the help of the switch node, the user can switch on/off the device.



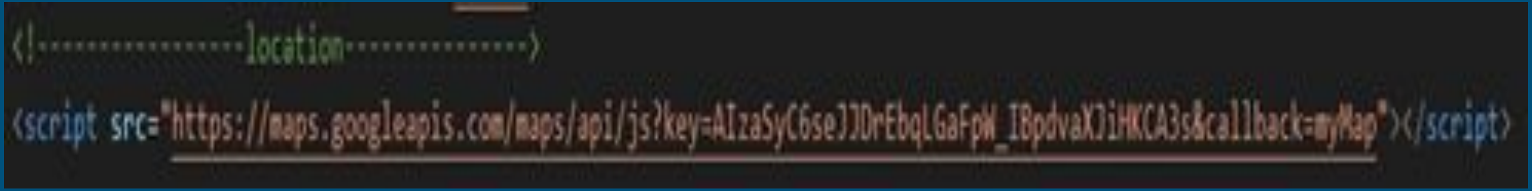
Realtime Database

https://sg-project-trial-default-rtdb.firebaseio.com/

```
sg-project-trial-default-rtdb
  ⊟─ object
       ├── latitude: 29.85297
       ├── longitude: 77.87545
       ├── pulserate: 75
       ├── switch: "on"
       └── temperature: 30
```

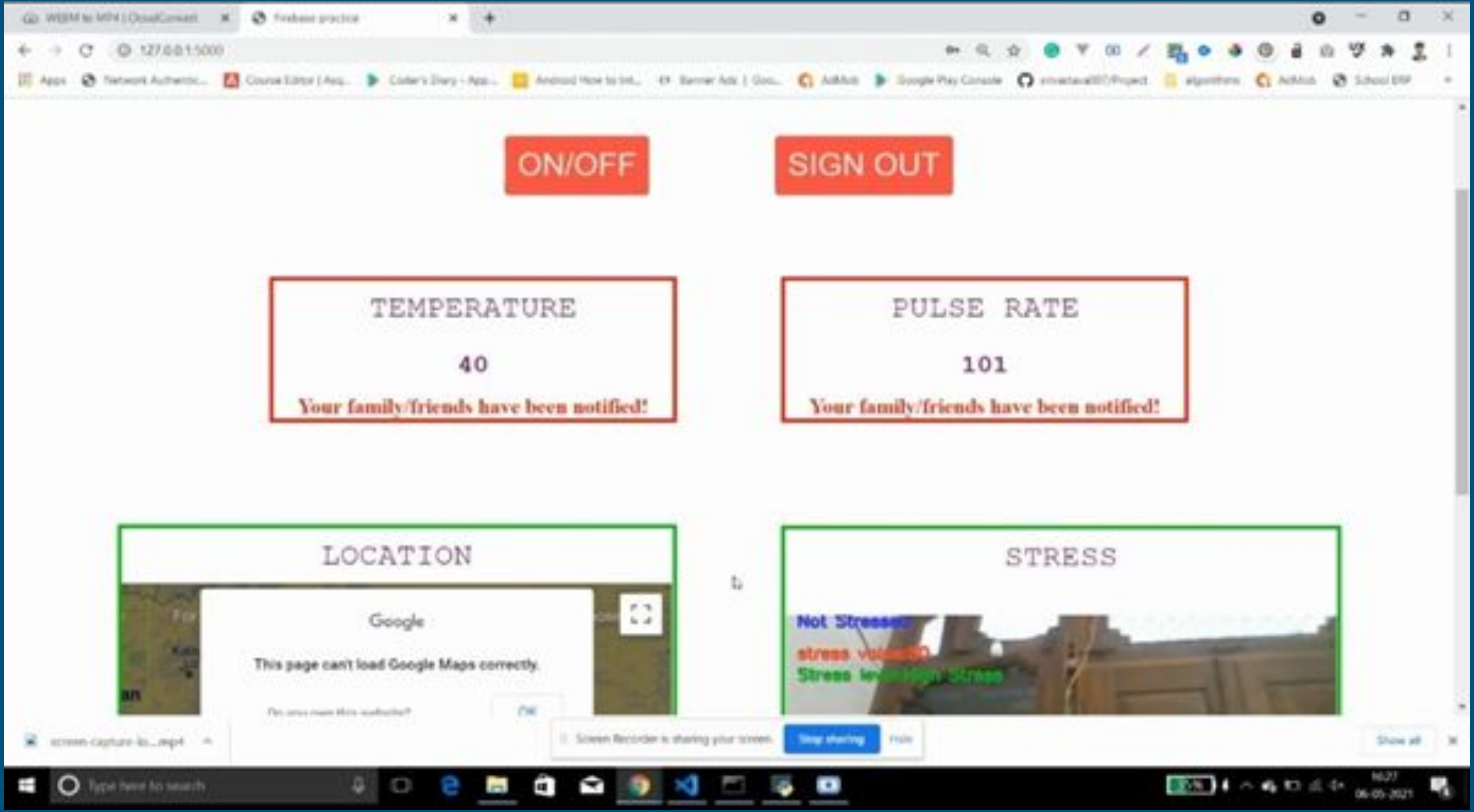
Database location: United States (us-central1)

We have used Firebase Email Authentication feature so that the user can sign in through his/her account and used the application.



```
<!------------------location---------------->
<script src="https://maps.googleapis.com/maps/api/js?key=AIzaSyC6seJJDrEbqLGaFpW_IBpdvaXJiHKCA3s&callback=myMap"></script>
```

Also, we have added a feature wherein a text message will be sent to the user's family/friends indicating that the temperature/pulse rate levels are in critical range. This is done with the help of Twilio Messaging which is an API to send and receive SMS, MMS, messages globally.



TEMPERATURE

40

Your family/friends have been notified!

PULSE RATE

101

Your family/friends have been notified!

LOCATION

Google

This page can't load Google Maps correctly.

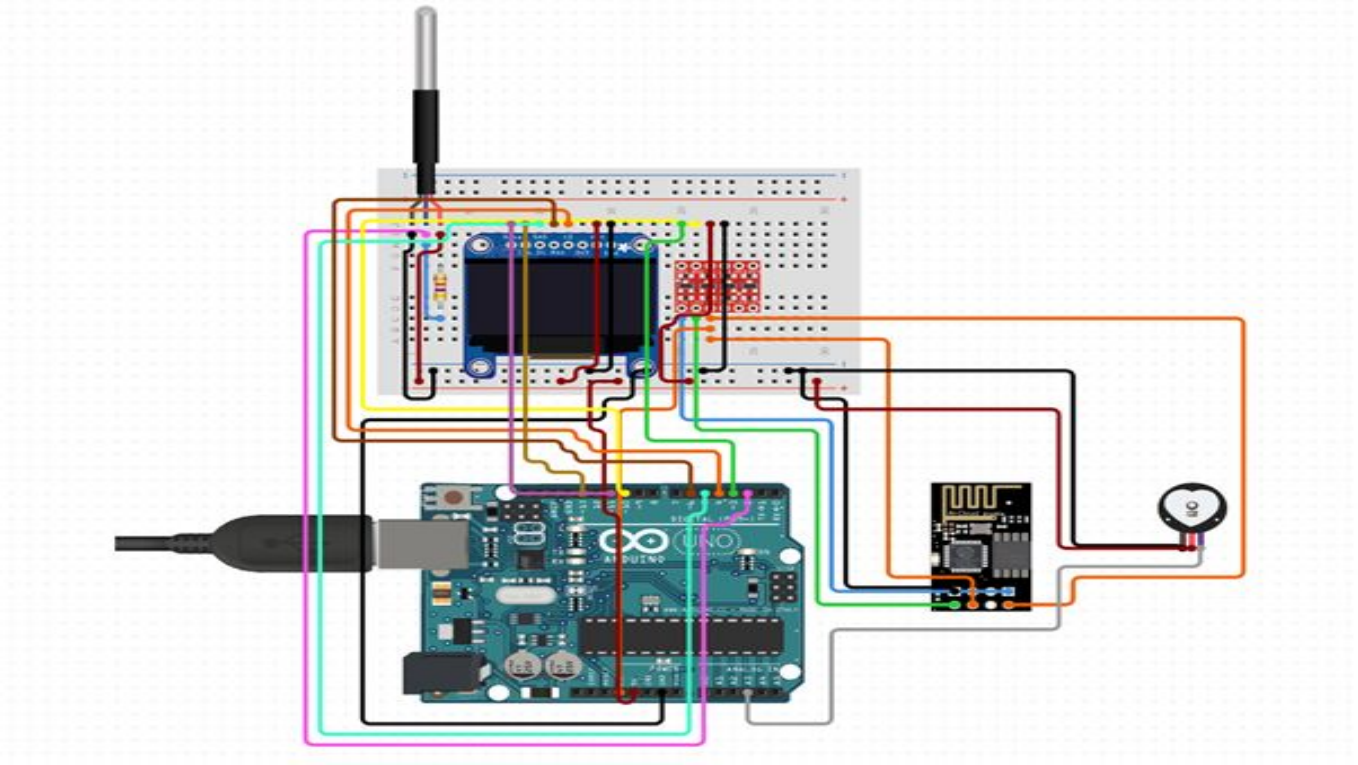
STRESS

Not Stressed

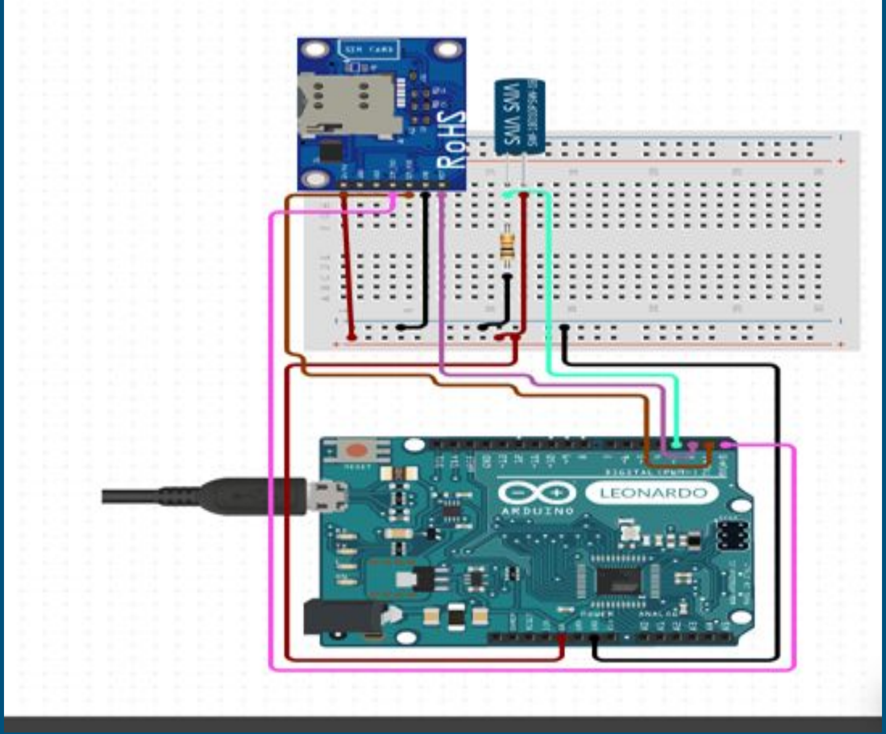
stress value:[illegible]

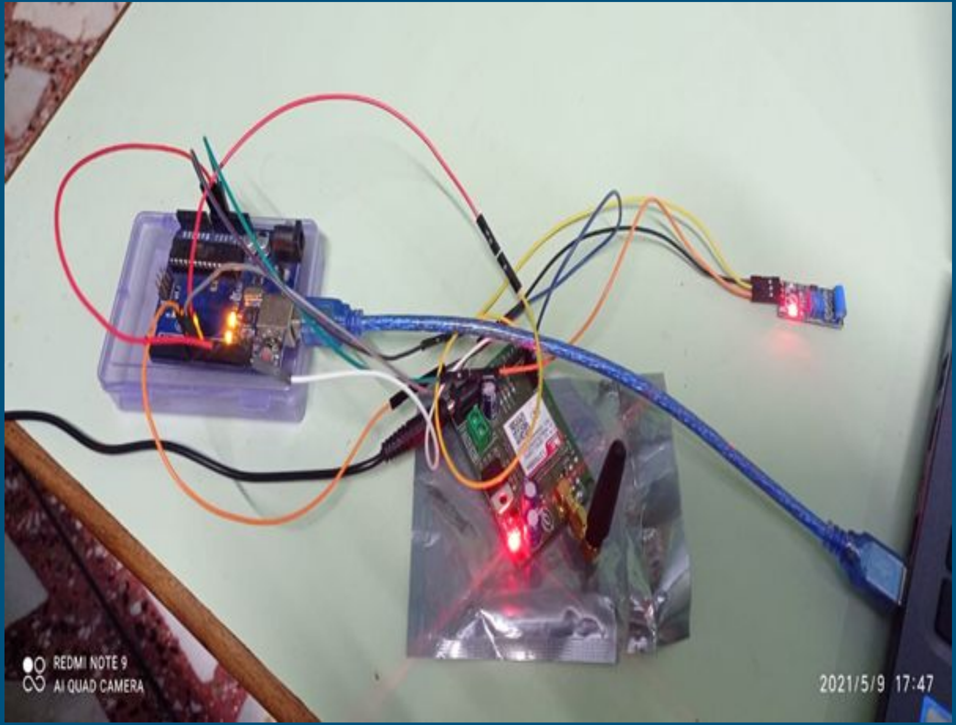
Stress level: High Stress

CIRCUIT DIAGRAM



Real Circuit diagram





Temperature,Pulse Detection and sending them to user through SMS Using GSM Module

The Arduino is connected to GSM module and temperature and pulse sensors. But due to lack of sensors we are using vibrational sensor instead of temperature and pulse sensors. Basically, vibrational sensors measure the intensity of vibrations. Values of the vibrational sensor are displayed on the website in place of pulse rate and temperature.

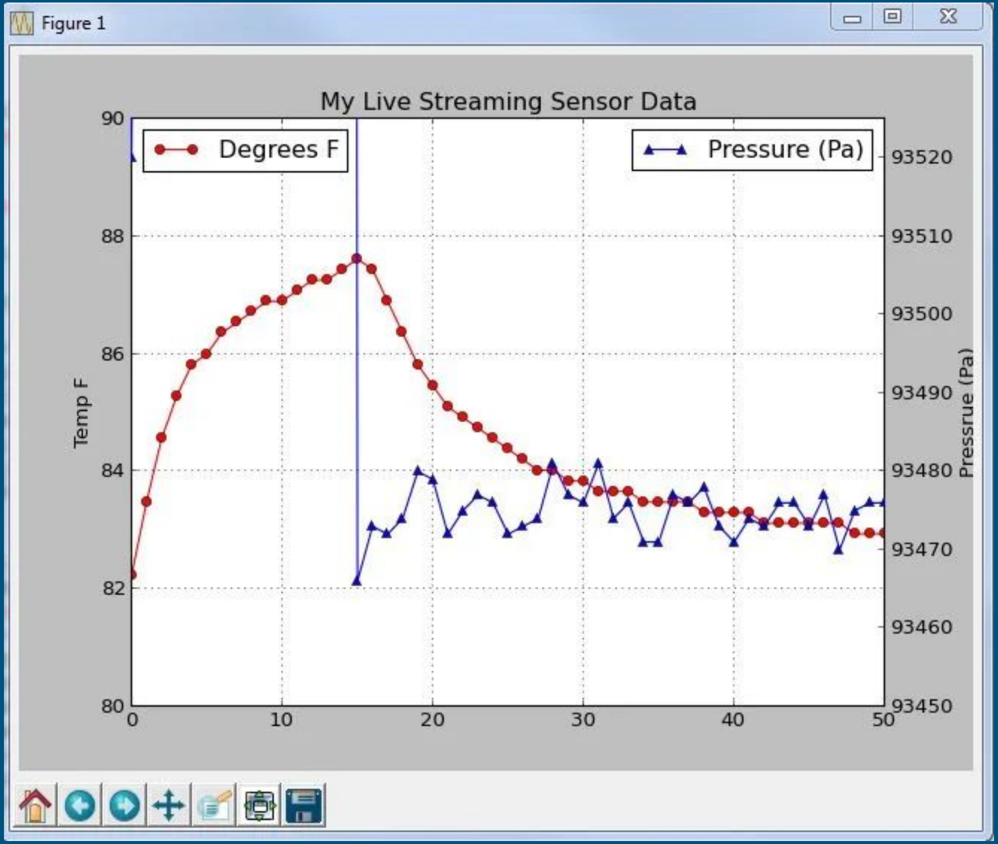
We are also using the GSM module to send the value of live temperature and live pulse rate to the user through SMS. The user must have to send the “on” command to the phone number of the sim which is inserted in the GSM module. As soon as the GSM module receives the “on” command it sends the values of temperature and pulse to the user through SMS. If user send the command other than “on” then also GSM will not respond, and user will receive no information.

We also add an additional feature so that the GSM module only responds to the command send by the user. We specified the mobile number of the user in the code. If anyone other than the user sends the message to the GSM module then the GSM module will not respond.

SENDING SENSORS VALUE TO FIREBASE AND LIVE LOCATION OF USER

We have used the pyserial Library to enable communication between the device and arduino, through the serial communication mode. The sensor values are sent from the arduino in a single line separated by commas. The data is then decoded, sliced and the unnecessary data (the result of some inconsistencies of the serial communication) is filtered. The values of sensor data is stored and continuously pushed to the real-time data base using the python firebase library. To fetch the live location of the user, we have made a web scraping bot, which automatically opens a website specified by us and fetches the latitude and longitude provided on the website, based on the user’s location; python selenium library and chrome driver are used to run the bot. The sensor values are pushed to the database only when we get the “on” command from the webapp, this has also been taken care of. Every time the switch is toggled the data is reset

LIVE PLOTTING OF GRAPH



Initially we are using OLED to show the real time graph of temperature and pulse. But due to lack of resources we cannot show the graph on OLED

So now we are using python to plot the real time graph of pulse rate and temperature.

We are using “matplotlib” and “drawnow” library of python to plot the realtime graph of pulse and temperature.

STRESS DETECTION FEATURE

The stress detection feature is made possible by using python along with some libraries like: Keras, OpenCV and Dlib. The trained model in the code detects the eyes, eyebrows and the lips from the video feed given to it. The distance between the lips, eyebrows and the aperture of eye is considered to determine the stress levels. The weights are assigned accordingly to all the parameters to get a final stress value. A particular threshold is assigned to the stress value, crossing which the stress level would be displayed as HIGH, in other cases it would be displayed LOW.

THANK YOU!