# http://www.mituniversity.edu.in/assets/logo.png

# MINI - PROJECT REPORT ON

## **“*Pothole Detector*”**

# BY

#### *Gurveen Singh 2173081*

#### *Kshitij Gunjalkar 2173104*

#### *Saad Ahmed 2173088*

#### *Sanskar Choubey 2173175*

##### Under the Guidance of

### *Prof. Reena Pagare*

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

**MIT SCHOOL of Engineering**

## Loni Kalbhor, Pune

2018 - 2019

**M.I.T. SCHOOL OF ENGINEERING**

DEPARTMENT OF COMPUTER ENGINEERING

## LONI – KALBHOR PUNE

***CERTIFICATE***



This is to certify that the Mini- Project report entitled

**“Pothole Detector”**

submitted by

#### *Gurveen Singh 2173081*

#### *Kshitij Gunjalkar 2173104*

#### *Saad Ahmed 2173088*

#### *Sanskar Choubey 2173175*

is a record of bonafide work carried out by them, under my guidance, in partial fulfillment of the requirement for the Second Year of Engineering(Computer ) at M.I.T. School of Engineering, Pune under MIT Art, Design & Technology University.

Date: 26/10/2018 Place: MITSOE

# Prof. Reena Pagare Dr. RajneeshKaur Sachdeo

# Guide, Dean Engineering,

**Department of CSE Head , Department of CSE**

**M.I.T. School of Engineering MIT School of Engineering**

**Loni Kalbhor, Pune Loni-Kalbhor, Pune**

## ***ACKNOWLEDGEMENT***

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along the completion of our project. All that we have done is only due to such supervision and assistance and we would not forget to thank them.

*We* respect and thank Ms.  RajneeshKaur Sachdeo , for providing us an opportunity to do the project work in MITSOE and giving us all support and guidance which made us complete the project duly. We are extremely thankful to her for providing such a nice support and guidance.

*We* owe our deep gratitude to our project guide Prof Reena Pagare, who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system.

*We heartily thank all other faculties and non-faculty members who shared there valuable advises and suggestions for betterment and modification of our project.*

*Gurveen Singh*

*Kshitij Gunjalkar*

*Saad Ahmed*

*Sanskar Choubey*

***LIST OF FIGURES***

|  |  |  |
| --- | --- | --- |
| Sr.NO. | Contents | Page No. |
| 1. | Figure 1: Arduino Uno r3 | 8 |
| 2. | Figure 2 : Ultrasonic sensor | 9 |
| 3. | Figure 3: GPS Neo 6m | 10 |
| 4. | Figure 4 : Example of IOT | 13 |
| 5. | Figure 5 : Flowchart | 15 |
| 6. | Figure 6: Serial Monitor Output | 17 |
| 7. | Figure 7 : My SQL | 18 |
| 8. | Figure 8 : Excel Sheet | 19 |
| 9. | Figure 9 : E- mail | 20 |

# List of Contents

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Contents** | **Page No.** |
| **1.** | **Title** | **1** |
| **2.** | **Certificate** | **2** |
| **3.** | **Acknowledgement** | **3** |
| **4.** | **List of figures** | **4** |
| **5.** | **Abstract** | **5** |
| **6.** | **Introduction** | **6** |
| **7.** | **Problem Definition** | **11** |
| **8.** | **Features of Project** | **11** |
| **9.** | ***Technology*** | **12** |
| **10** | **Flowchart** | **15** |
| **11.** | **Output** | **17** |
| **12.** | **Code** | **21** |
| **13.** | **Reference** | **29** |
| **14.** | **Annexure A** | **30** |

**Abstract**

Many on-going projects in the field of vehicular networks are working in the direction of providing driver with relevant information about roads and traffic moments. One of the major problems in developing countries is maintenance of roads. Well maintained roads contribute a major portion to the country’s economy. And also, many people are killed by unwanted potholes on road. So, Identification of pavement distress such as potholes not only helps drivers to avoid accidents or vehicle damages but also helps authorities to maintain roads. The implementation of this service will provide users with valuable information about the conditions of roadways that will limit potential damage to the user’s vehicle or Maintenance Authority.

The pothole detector project proposes a cost effective solution to identify potholes on roads and provide timely alerts to concerned authority to avoid accidents or vehicle damages, and also send Email to concerned authorities. Ultrasonic sensor is used to identify potholes and also to measure their depth. The GPS is being used to trace the location of the detected pothole The sensed-data includes pothole depth and location of the potholes. The following data will be Emailed to the concerned authority in regular interval of time by the operator. After receiving the data the authorities would be able to plan the route of maintenance in more accurate, exact and efficient way helping them to save time, money and other resources like fuel and which will cut the price of maintenance.

The proposed system will help in development and maintenance of roads in more accurate way.

# Introduction

India is a fastest developing country after china. Although India is doing exceptionally well in certain field, they majorly lack in road ways. Roads are the dominant means of transportation in India today. However, roads here are narrow and congested with poor quality and is not maintained properly. This road condition is a boosting factor for traffic congestion and number of road accidents. Potholes are formed due to heavy movements of vehicles and heavy rains and water logging. According to the survey of “road accidents in India 2015” road accidents increased by 2.5 per cent from 4,89,400 in 2014 to 5.01,423 in 2015. Over 11,000 people killed by potholes last year. To overcome this pathetic condition many technologies have been implemented and integrated. Internet of Things (IoT) is an ever-growing network of physical objects (which also includes people). The concept is basically Information gathering is the first step in IOT.

To deal with this problem we here are proposing an ultrasonic based pothole detection system. In this model we have implemented IOT. We, in this project are using ultrasonic HC-SR04 model, GPS Neo 6M model and the brain of all this Arduino UNO.

**Arduino UNO:**

[8]

The Arduino UNO is an open-source microcontroller board based on the [Microchip](https://en.wikipedia.org/wiki/Microchip_Technology" \o "Microchip Technology) [ATmega328P](https://en.wikipedia.org/wiki/ATmega328P" \o "ATmega328P) microcontroller and developed by [Arduino.cc](https://en.wikipedia.org/wiki/Arduino" \o "Arduino).[[2]](https://en.wikipedia.org/wiki/Arduino_Uno" \l "cite_note-2)[[3]](https://en.wikipedia.org/wiki/Arduino_Uno" \l "cite_note-What_is_Arduino?-3) The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits.[[1]](https://en.wikipedia.org/wiki/Arduino_Uno" \l "cite_note-Makerspace-1) The board has 14 Digital pins, 6 Analog pins, and programmable with the [Arduino IDE](https://en.wikipedia.org/wiki/Arduino" \l "Software" \o "Arduino) (Integrated Development Environment) via a type B USB cable.[[4]](https://en.wikipedia.org/wiki/Arduino_Uno" \l "cite_note-priceton-4) It can be powered by a USB cable or by an external 9 volt battery, though it accepts voltages between 7 and 20 volts. It is also similar to the Arduino Nano and Leonardo.[[5]](https://en.wikipedia.org/wiki/Arduino_Uno" \l "cite_note-5)[[6]](https://en.wikipedia.org/wiki/Arduino_Uno" \l "cite_note-6) The hardware reference design is distributed under a [Creative Commons](https://en.wikipedia.org/wiki/Creative_Commons" \o "Creative Commons) Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available. "Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0.[[1]](https://en.wikipedia.org/wiki/Arduino_Uno" \l "cite_note-Makerspace-1) The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases.[[4]](https://en.wikipedia.org/wiki/Arduino_Uno" \l "cite_note-priceton-4) The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform.[[3]](https://en.wikipedia.org/wiki/Arduino_Uno" \l "cite_note-What_is_Arduino?-3) The ATmega328 on the Arduino Uno comes preprogrammed with a bootloader that allows uploading new code to it without the use of an external hardware programmer.[[3]](https://en.wikipedia.org/wiki/Arduino_Uno" \l "cite_note-What_is_Arduino?-3) It communicates using the original STK500 protocol.[[1]](https://en.wikipedia.org/wiki/Arduino_Uno" \l "cite_note-Makerspace-1) The Uno also differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.[[7]](https://en.wikipedia.org/wiki/Arduino_Uno" \l "cite_note-website-7)

****

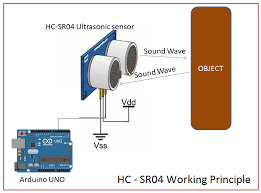
**Fig.No. 1 Arduino Uno r3**

**Ultrasonic sensor:**

[9]

The sensor used in the design model is the ultrasonic sensor. The HC-SR04 is an active ultrasonic sensor and contains a transmitter and a receiver. It is used to measure distance at which, objects are placed in front of it. The ultrasonic sensor transmits high frequency sound waves and waits for the reflected wave to hit the receiver. The distance is calculated based on the time taken by the ultrasonic pulse to travel a particular distance. There are different types of ultrasonic sensors with different transmission ranges and angles of detection.

The HC-SR04 sensor work at frequency of 40 KHz and can measure distances of the objects in the range 2 to 400 cm with a 15° angle of detection.

****

**Fig.NO. 2 Ultrasonic Sensor**

**GPS Neo 6M:**

[10]

The NEO-6M GPS module is a well-performing complete GPS receiver with a built-in 25 x 25 x 4mm ceramic antenna, which provides a strong satellite search capability. With the power and signal indicators, you can monitor the status of the module. Thanks to the data backup battery, the module can save the data when the main power is shut down accidentally. Its 3mm mounting holes can ensure easy assembly on your aircraft, which thus can fly steadily at a fixed position, return to Home automatically, and automatic waypoint flying, etc. Or you can apply it on your smart robot car for automatic returning or heading to a certain destination, making it a real "smart" bot!

****

**Fig.No. 3 GPS NEO 6M**

**Working:**

After compiling and updating the code from Arduino IDE, the runtime output will be shown in the serial monitor of Arduino. The readings will be shown in the format of depth of pothole, latitude and longitude. After the program is shown in runtime the data will be stored in a MySQL file which will then be converted into a excel sheet with the help of CSV file. Then this excel format sheet will be emailed to the concerned authority in two clicks in regular interval of time by the operator of the project.

# Problem Definition

Using Arduino Uno board, Ultrasonic sensor and GPS to detect pothole on the road and sending information(gps location of pothole)

to road authority.

# Features of Project

* Execution and handling of the project is easy.
* No wastage of time as we get exact location of the pothole.
* The maintenance team gets the data in regular interval of time.
* Less cost, the overall cost of the project is very less and that cost can be freed in very less span of time.
* The data is stored in an excel sheet which is easy to share and easy to keep an eye on it.
* Saving of resources like fuel, the concerned authority can directly go to the location of pothole which saves the time and fuel.

# Technology

The platform used in this project is Internet of Things i.e. IoT. The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers ([UIDs](https://internetofthingsagenda.techtarget.com/definition/unique-identifier-UID)) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. Increasingly, organizations in a variety of industries are using IoT to operate more efficiently, better understand customers to deliver enhanced customer service, improve decision-making and increase the value of the business.

### How IoT works:

### An IoT ecosystem consists of web-enabled smart devices that use embedded processors, sensors and communication hardware to collect, send and act on data they acquire from their environments. [IoT devices](https://internetofthingsagenda.techtarget.com/definition/IoT-device) share the sensor data they collect by connecting to an IoT gateway or other edge device where data is either sent to the cloud to be analyzed or analyzed locally. Sometimes, these devices communicate with other related devices and act on the information they get from one another. The devices do most of the work without human intervention, although people can interact with the devices -- for instance, to set them up, give them instructions or access the data.

The connectivity, networking and communication protocols used with these web-enabled devices largely depend on the specific IoT applications deployed.



**Fig.No. 4**

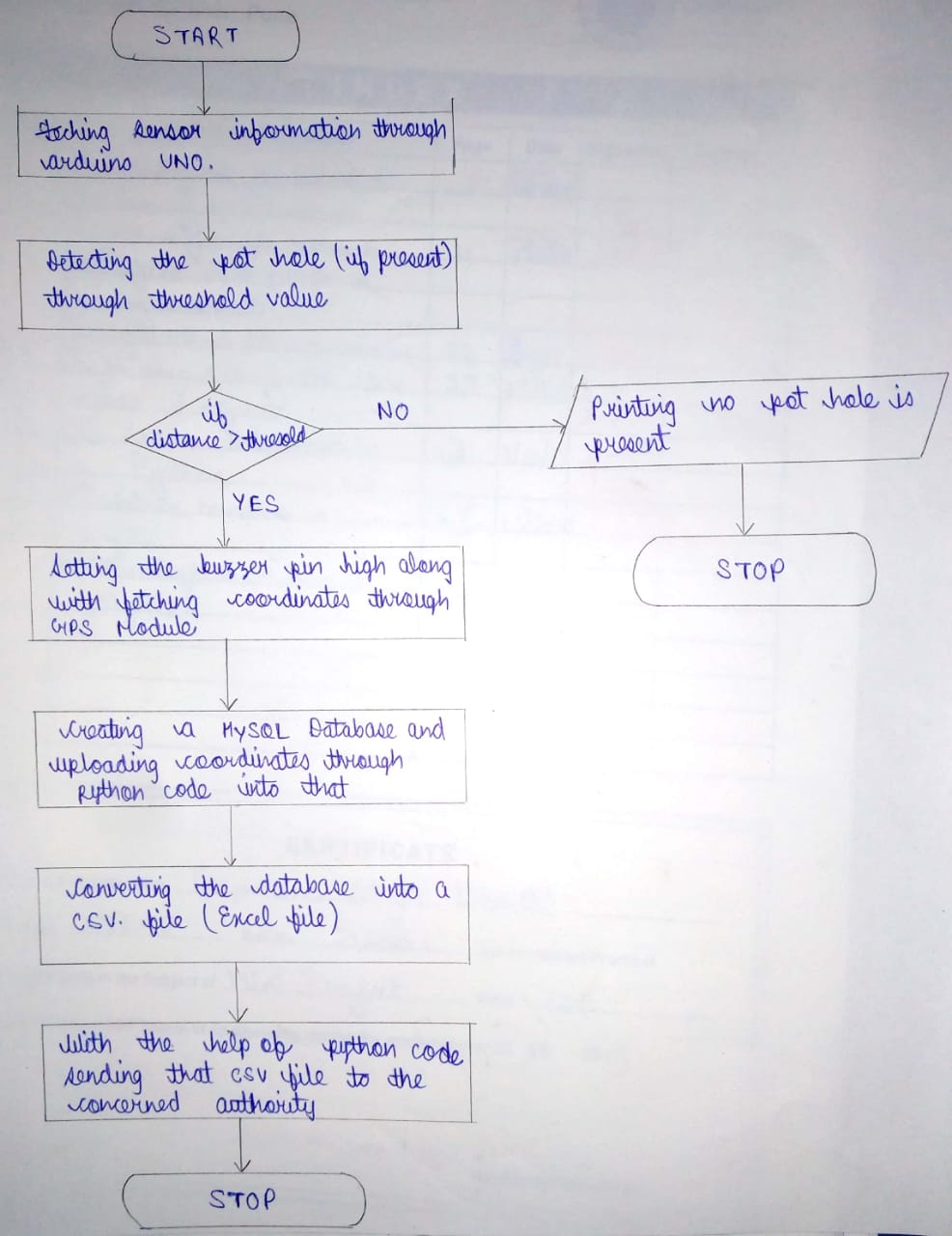
### Benefits of IoT

The internet of things offers a number of benefits to organizations, enabling them to:

* monitor their overall business processes;
* improve the customer experience;
* save time and money;
* enhance employee productivity;
* integrate and adapt business models;
* make better business decisions; and
* generate more revenue.

IoT encourages companies to rethink the ways they approach their businesses, industries and markets and gives them the tools to improve their business strategies.

# 5.Flowchart



**Fig.No. 5**

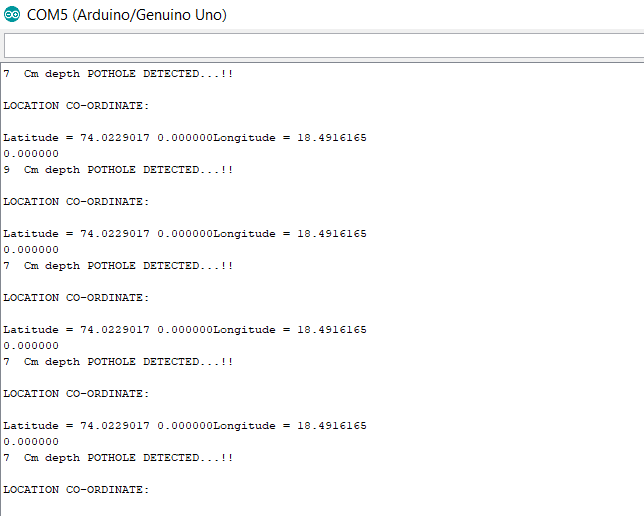
### Explanation:

First the ultrasonic and buzzer is interfaced to Arduino. Later, GPS is interfaced to Arduino. Then the code is compiled and uploaded to the Arduino board. The data is fetched through the sensors and the runtime output is shown on the serial monitor of Arduino IDE. If the distance is greater than the threshold value a pothole is detected and the latitude and longitude of that particular pothole is shown.

The data is stored in a My SQL database. The data is shown in a tabular format in my SQL. After storing the data in My SQL the file is converted into a csv file for converting the my sql file into excel sheet. After the file is converted into csv file, the command to email the concerned authority is executed and the concerned authority receives the email.

# 6.Outputs

**Serial Monitor:**

****

**Fig.No. 6**

The serial monitor shows the runtime output of the project. The following display is showing the location , depth of pothole.

**My SQL:**

The following screenshot shows us the data stored in my SQL database.

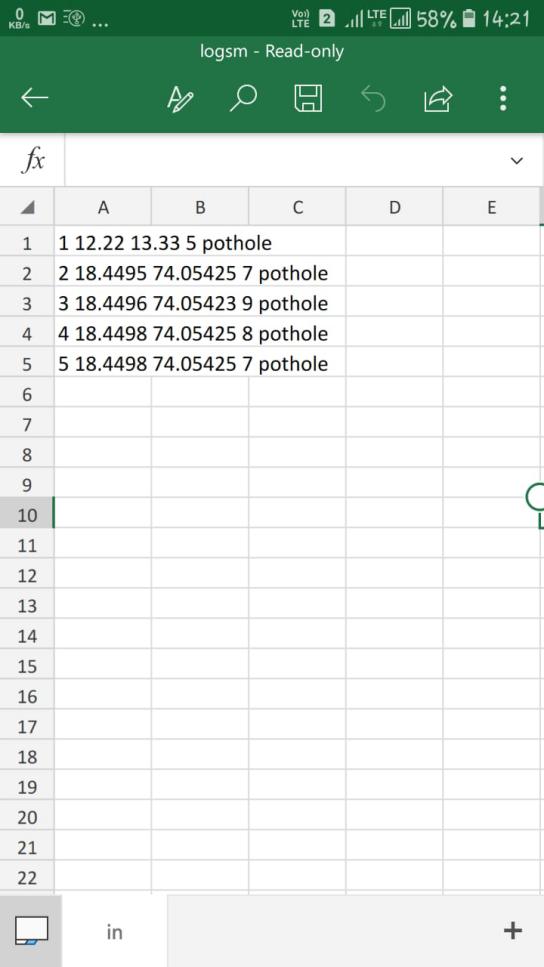
The data stored here is in tabular format. The data shown in table are

Depth of pothole, latitude and longitude.

****

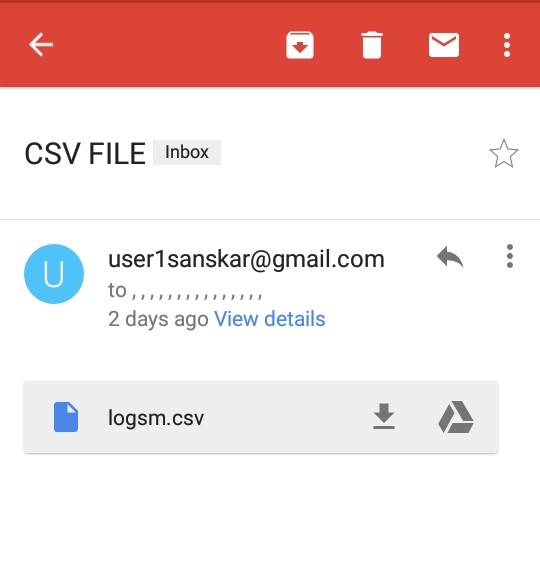
**Fig.No. 7**

**Excel Sheet:**



**Fig.No. 8**

**E-mail:**



**Fig.No. 9**

# Codes

**Interfacing of Arduino with GPS:**

#include <TinyGPS++.h>

#include <SoftwareSerial.h>

SoftwareSerial ss(4, 3);

int const trigPin = 10;

int const echoPin = 9;

int const buzzPin = 2;

int const RXPin = 4;

int const TXPin = 3;

TinyGPSPlus gps; // The TinyGPS++ object

void setup()

{

Serial.begin (9600);

pinMode(trigPin, OUTPUT); // trig pin will have pulses output

pinMode(echoPin, INPUT); // echo pin should be input to get pulse width

pinMode(buzzPin, OUTPUT); // buzz pin is output to control buzzering

}

void loop()

{

// Duration will be the input pulse width and distance will be the distance to the obstacle in centimeters

int duration, distance;

// Output pulse with 1ms width on trigPin

digitalWrite(trigPin, HIGH);

delay(1);

digitalWrite(trigPin, LOW);

// Measure the pulse input in echo pin

duration = pulseIn(echoPin, HIGH);

// Distance is half the duration devided by 29.1 (from datasheet)

distance = (duration / 2) / 29.1;

// if distance less than 0.5 meter and more than 0 (0 or less means over range)

if (distance > 6)

{

// Buzz

digitalWrite(buzzPin, HIGH);

Serial.print(distance);

Serial.print(" Cm depth POTHOLE DETECTED...!!\n");

gps.encode(ss.read());

Serial.print("\nLOCATION CO-ORDINATE: \n");

Serial.print("\nLatitude = ");

Serial.print(gps.location.lat(), 6);

Serial.print("Longitude = \n");

Serial.println(gps.location.lng(), 6);

}

else

{

// Don't buzz

digitalWrite(buzzPin, LOW);

}

// Waiting 60 ms won't hurt any one

delay(60);

}

**Interfacing of Arduino with My SQL:**

import serial

import MySQLdb

#for establishing connection to MySQLdb

dbConn=MySQLdb.connect(host="localhost",user="sanskar",passwd="Hello@123",db="pothole\_logs")

#code for opening a cursor to the database

cursor = dbConn.cursor()

device = '/dev/tty.usbmodemCOM5' # change according to your serial\_port

try:

print("Trying...",device)

arduino = serial.Serial(device,9600)

except:

print("Failed to connect on",device)

try:

data = arduino.readline() #for reading the data from the arduino

pieces = data.split("\t") #split the data by the tab

try:

cursor.execute("INSERT INTO logs(latitude, longitude, depth, status)VALUES(%s,%s,%s,%s)",(pieces[0],pieces[1],pieces[2],pieces[3]))

dbConn.commit() #commit the INSERT

cursor.close() #close the cursor

except MySQldb.IntegrityError:

print("failed to insert data")

finally:

cursor.close() #close just in case it Failed

except:

print("Failed to get data from Arduino!!!!")

**For Converting My SQl file .csv file:**

import MySQLdb as dbapi

import smtplib

import sys

import csv

import config

import os

from email.MIMEMultipart import MIMEMultipart

from email.MIMEBase import MIMEBase

from email import Encoders

def send\_email():

SUBJECT = "CSV FILE"

msg = MIMEMultipart()

msg['Subject'] = SUBJECT

msg['From'] = config.EMAIL\_ADDRESS

msg['To'] = ', '.join(config.SENT\_ADDRESS)

part = MIMEBase('application', "octet-stream")

part.set\_payload(open("logsm.csv", "rb").read())

Encoders.encode\_base64(part)

part.add\_header('content-disposition', 'attachment; filename="logsm.csv"')

msg.attach(part)

server = smtplib.SMTP("smtp.gmail.com:587")

server.ehlo()

server.starttls()

server.login(config.EMAIL\_ADDRESS, config.PASSWORD)

server.sendmail(config.EMAIL\_ADDRESS, config.SENT\_ADDRESS, msg.as\_string())

server.quit()

print('Email Sent Successfull')

dbServer = 'localhost'

dbPass= 'Hello@123'

dbSchema='pothole\_logs'

dbUser='sanskar'

dbQuery = 'SELECT \* FROM logs into outfile "//var//lib//mysql-files//logsm.csv" fields terminated by "\t" lines terminated by "\n";'

db = dbapi.connect(host=dbServer, user=dbUser, passwd=dbPass, db="pothole\_logs")

cur=db.cursor()

cur.execute(dbQuery)

result = cur.fetchall()

cur.close()

os.system('sudo cp /var/lib/mysql-files/logsm.csv /home/anonymous/Desktop/miniproject\(source\ code\)/')

send\_email()

**For Configuring .csv file:**

EMAIL\_ADDRESS="user1sanskar@gmail.com"

PASSWORD="User12345"

SENT\_ADDRESS="gunjalkarkshitij@gmail.com"

# References

[1] **IOT Based Pothole Detection and Notification System**

S. Gnanapriya, V.B. Padmashree, V. Bagyalakshmi and G.A. Pravallikha

Department of Information Technology, Easwari Engineering, College, Chennai, India

1. [https:///www.arduino.cc/en/guide/introduction](HTTPS://WWW.ARDUINO.CC/EN/GUIDE/INTODUCTION)
2. <https://timesofindia.indiatimes.com/auto/miscellaneous/arai-revises-ground-clearance-measurement-norms/articleshow/59599124.cms>
3. <https://store.arduino.cc/arduino-uno-rev3>
4. <https://www.hackster.io/Salmanfarisvp/ultrasonic-range-detector-with-arduino-46c96c>

[6 ]<https://idyl.io/arduino/how-to/interface-gps-module-arduino/>

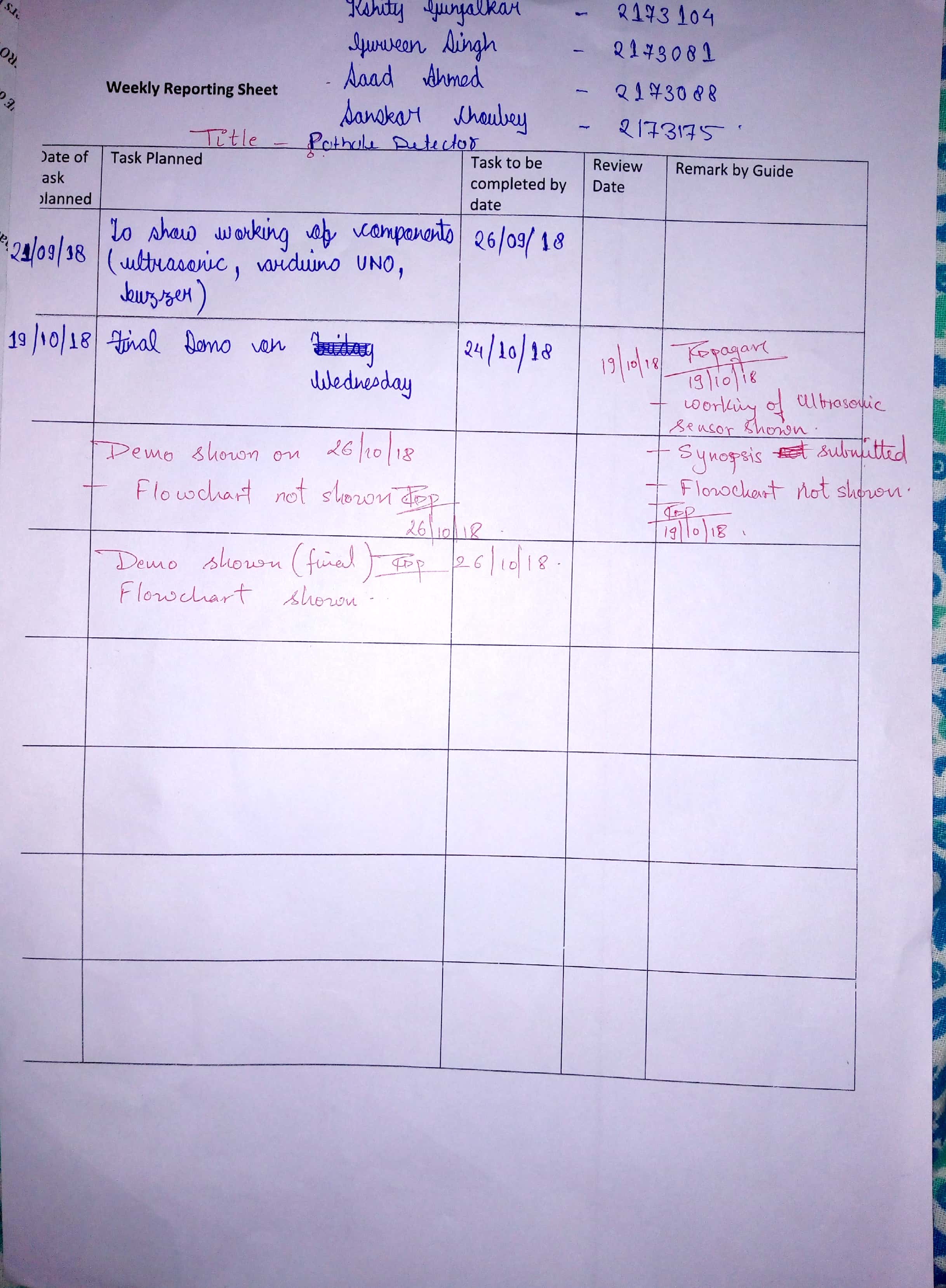
[7] <https://www.tutorialspoint.com/mysql/>

[8] <https://en.wikipedia.org/wiki/Arduino_Uno>

[9] <https://components101.com/ultrasonic-sensor-working-pinout-datasheet>

[10] <http://wiki.sunfounder.cc/index.php?title=Ublox_NEO-6M_GPS_Module>

**Annexure A**

****