SMART CITY

PROJECT REVIEW-3

Submitted in partial fulfillment of the requirements for the degree of

Bachelor of Technology
in
COMPUTER SCIENCE

1. AKANCHA AGARWAL – 17BCB0016
2. DIKSHA SINGHANIA – 17BCE0032
3. NISHTHA DUBEY – 17BCE0082
4. JANANI J. – 17BCE0207
5. MRINALINI SINGH – 17BCE0396
6. DIVYA GAHLOT – 17BCE0509
7. ISHITA GUPTA – 17BCE0710
8. SANDHYA ANANTHAN – 17BCI0099

CSE3999_TECHNICAL-ANSWERS-FOR-REAL-WORLD-PROBLEMS

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

DECLARATION

We hereby declare that the thesis entitled “SMART CITY" submitted by me, for the award of the degree of *Bachelor of Technology in Computer Science* to VIT is are cord of bonafide work carried out by us under the supervision of Dr. Pounambal.

We further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or Diploma in this institute or any other institute or university.

Place: Vellore
Date: 04-06-2020

ACKNOWLEDGEMENTS

This is to certify that the thesis entitled "SMART CITY" submitted by AKANCHA-17BCB0016, DIKSHA SINGHANIA – 17BCE0032, NISHTHA – 17BCE0082, JANANI J. – 17BCE0207, MRINALINI SINGH – 17BCE0396, DIVYA GAHLOT – 17BCE0509, ISHITA – 17BCE0710 and SANDHYA ANANTHAN – 17BCI0099, VIT, for the award of the degree of Bachelor of Technology Program, is are cord of bonafide work carried out by him/her under my supervision during the period, 01.11..2019 to 08.06.2020, as per the VIT code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfills the requirements and regulations of the University and in my opinion meets the necessary standards for submission.

Place: Vellore
Date: **04-06-2020**

EXECUTIVE SUMMARY

In our project, we aim to demonstrate some of the challenges and opportunities offered in the field of study of smart cities through the use of simulations, models and deliverables. We will be building a smart city model which will comprise of Smart street lights to save electricity, Smart Garbage bins for efficient waste disposal, Smart Door Locks that will ensure safety in homes and an Efficient Car Parking System that will facilitate the traffic control system.

We will also be implementing an air pollution control system which will keep a check on the ever-increasing air pollution, CCTV system to provide a secure neighborhood, Traffic control System to ease navigation for commuters and a smart gardening system for the farmers. We shall make extensive use of various technologies such as IoT, Image Processing, Android App Development etc. throughout the project undertaking to explore different aspects and come up with an efficient model to demonstrate the effectiveness of the proposed system.

CONTENTS

LIST OF ABBREVIATIONS

S.No.	Abbreviation	Full Word
1.	IOT	Internet of Things
2.	ML	Machine Learning
3.	KV	KiloVolts
4.	LPWAN	Low-power wide-area network
5.	CCTV	Closed circuit television
6.	LED	Light Emitting Diode

1. INTRODUCTION

1.1 OBJECTIVE

In our project, we aim to demonstrate some of the challenges and opportunities offered in the field of study of smart cities through the use of simulations, models and deliverables. We will be building a smart city model which will comprise of Smart street lights to save electricity, Smart Garbage bins for efficient waste disposal, Smart Door Locks that will ensure safety in homes and an Efficient Car Parking System that will facilitate the traffic control system.

We will also be implementing an air pollution control system which will keep a check on the ever-increasing air pollution, CCTV system to provide a secure neighborhood, Traffic control System to ease navigation for commuters and a smart gardening system for the farmers. We shall make extensive use of various technologies such as IoT, Image Processing, Android App Development etc. throughout the project undertaking to explore different aspects and come up with an efficient model to demonstrate the effectiveness of the proposed system.

1.2 MOTIVATION

The sole motivation behind this project was to learn how smart cities could be made. Smart Cities are on the rise. There are many cities worldwide that are planning projects or strategies intended to implement smart solutions. Though there are many success stories, there are no valid reference models yet that can be used to establish a clear path toward a Smart City because, among other reasons, cities are living entities with a vast number of variables and many areas of activity. Cities, like all other organizations, evolve and modernize based on a set of projects. Having a model would allow not only for improved planning, but also for a more suitable and uniform determination of each city's "smart" ability, and thus to establish a ranking of the potential of the projects proposed for their development. In this research paper we analyze some of the current proposals for Smart Cities and we put forth a model based on so-called motivating factors, which identify those projects or areas of greatest interest that have made it possible for many of the world's Smart Cities to be created and to thrive.

1.3 BACKGROUND

The world is in transition and two of the principal factors in this transition are the accelerating advances in computer and communications technologies and the migration of our civilization into cities. More than half of the world's people are urban-dwellers today and the rate of migration from rural to urban is increasing. Some 500 million people are in the process of urbanizing at this moment. Cities also consume 75% of the world's energy and produce 85% of the world's greenhouse gases. How civic leaders cope with the influx of people and deal with issues such as infrastructure, new ultra broadband communications and information technologies, health care, security and commerce -- is a major concern today. At the same time, opportunities abound as some leaders transform their communities through the application of the ever evolving computer and communications technologies. Copenhagen has smart energy, Kyoto has smart government, Stockholm has smart traffic, Zurich has smart healthcare, New York has smart crime fighting and these are just a few examples.

In the end, the IoT technology stack for smart city applications today is relatively easy and cheap for many use cases such as smart waste management or smart parking. Urban environments typically have good wireless coverage for cases with moving parts, you have the cloud, there are several point solutions and products which are designed for smart city projects and in several cities across the globe there is low-power wide-area network connectivity available (LPWAN) which suffices for many applications. In fact, when drafting its list of IoT applications, IoT Analytics found that 59 percent (!) of all LPWAN projects today are part of smart city initiatives while LPWAN in general is only present in 10 percent of the IoT projects the company identified.

Choices regarding connectivity, data exchange, IoT platforms and so forth will evolve as we start moving to real smart cities. It’s best if these future choices are already being taken into account for current smart city deployments.

2. PROJECT DESCRIPTION AND GOALS

2.1 PROJECT DESCRIPTION

The smart city concept represents a compelling platform for IT-enabled service innovation. It offers a view of the city where service providers use information technologies to engage with citizens to create more effective urban organizations and systems that can improve the quality of life. The emerging Internet of Things (IoT) model is foundational to the development of smart cities. Integrated cloud-oriented architecture of networks, software, sensors, human interfaces, and data analytics are essential for value creation. IoT smart-connected products and the services they provision will become essential for the future development of smart cities. Our project uses the smart city concept and proposes a strategy development model for the implementation of IoT systems in a smart city context.

The Internet of Things is an infrastructure that includes physical devices, modern vehicles, buildings, and even essential electrical devices which we use on a consistent basis inter-connected to each other over the internet so that they can accumulate and exchange data amongst themselves.. The concept of IoT aims to present the Internet even more pervasive and even more immersive. The IoT will improve the development of various applications that make use of the massive amount and diversity of data produced by objects to implement further services to companies, citizens, and public administrations.

The important concept of smart cities is the waste management which is very much trending and helpful these days. This project manages the garbage collection done by Municipal Corporation with the help of an IOT based embedded device attached to dustbin of each area, this device continuously updates the status of dustbins in each area. This device continuously detects the level of dustbin using sensors and as the dustbin gets full it will update its status of getting full and will go to waiting state and remain in this state till dustbin gets empty. Once the dustbin is cleaned by the employees the device will comes out of waiting state and will update its status of getting cleaned. Thus a record is maintained regarding dustbin status for each area in the website in tabular form using IOT technology along with embedded system which will efficiently manage the garbage collection will resolve the major environmental issue of inefficient garbage collection leads to a clean and healthy environment.

Traffic congestion caused by vehicle is an alarming problem at a global scale and it has been growing exponentially. Car parking problem is a major contributor and has been, still a major problem with increasing vehicle size in the luxurious segment and confined parking spaces in urban cities. Searching for a parking space is a routine (and often frustrating) activity for many people in cities around the world. This search burns about one million barrels of the world’s oil every day. As the global population continues to urbanize, without a well-planned, convenience-driven retreat from the car these problems will worsen. Smart Parking systems typically obtains information about available parking spaces in a particular geographic area and process is real-time to place vehicles at available positions .It involves using low-cost sensors, real-time data collection, and mobile-phone-enabled automated payment systems that allow people to reserve parking in advance or very accurately predict where they will likely find a spot. When deployed as a system, smart parking thus reduces car emissions in urban centers by reducing the need for people to needlessly circle city blocks searching for parking. It also permits cities to carefully manage their parking supply.

Using physical keys to lock or unlock the door is the most natural way and everyone is acquainted with it. The physical key is a well-tested and well-known technology, but it also has its flaws. There can only be one unique key for a lock. For different locks you have different keys. Furthermore, carrying a large number of keys is a burden and increases the chance of keys getting stolen, misplaced or lost. We will be developing an exceptionally improvised system by which we can lock and unlock main door remotely without requirement of any physical effort or need of a key. When an unknown person presses the calling bell, immediately the door camera capture his/her image and send the same to owner smart app using internet. Owner can lock or unlock the door remotely through the secure app installed in his / her smart phone. The app has additional feature to take exterior image / video for monitoring the activity going on outside the door. The system has the capability of sending the notification if the door - lock goes offline. Our proposed internet of things or iot based system facilitate the user hazard free, simple, robust and secure solution for home security and eradicate manual effort towards door lock – unlock issues.

Development in the civilization includes everything in the human society and one of the most important civilization indexes is the good transportation network. This includes streets, roads and highways that has to be adequately illuminated so that a sufficient visibility is guaranteed to the safety of the people travelling on the streets. Currently, in the whole world, enormous electric energy is consumed by the street lamps, which are automatically turn on when it becomes dark and automatically turn off when it becomes bright. This is the huge waste of energy in the whole world and should be changed. The Smart Street Lighting system is proposed in the view to overcome the drawback of the existing lighting system used. Our smart street light system consists of a LED light, a motion sensor and a short-distance communication network. The lights turn on before pedestrians and vehicles come and turn off or reduce power when there is no one. It will be difficult for pedestrians and drivers of vehicles to distinguish our smart street lamps and the conventional street lights, since our street lamps all turn on before they come.

Vehicles have become an integral part of every one's life. Situations and circumstances demand the usage of vehicles in this fast paced urban life. As a coin has two sides, this has its own effects, one of the main side effects being air pollution. Every vehicle will have emission but the problem occurs when it is beyond the standardized values. The primary reason for this breach of emission level being the incomplete combustion of fuel supplied to engine, which is due to the improper maintenance of vehicles. This emission from vehicles cannot be completely avoided but, it definitely can be controlled. . In order to monitor quality of air, a Wireless sensor network (WSN) based new framework is proposed which is based on data acquisition and transmission. The parameters of the environment to be monitored are chosen as temperature, humidity, volume of CO, volume of CO2, detection of leakage of any gas - smoke, alcohol, LPG. Thus the proposed model will be extremely useful in assessing the quality of air to ensure good health and wellbeing of the masses by monitoring the various elements present in air.

Smart gardening is an efficient way to manage the plants at our garden using the technology of IOT. Various sensors such as moisture sensor, humidity sensor, temperature sensor and ultrasonic sensor are used to check the moisture content of the soil and intensity of sunlight falling on plant at regular time intervals. After the data is processed, we can get to know about the adequate amount of water and other nutrients that a plant needs in order to maintain a healthy garden.

Traffic signal management is one of the major problematic issues in the current situation. Such scenarios, every signal are getting 60 seconds of timing on the road at a regular interval, even when traffic on that particular road is dense. As per the proposed model in this article, optimization of the timing interval of the traffic signal purely depends on the number of vehicles on that particular roadside. The major advantage of this system is that it can able to decrease the more waiting time for the drivers to cross road signal. In this model, the clustering algorithm models are used which are based on KNN algorithm. Using this algorithm new model will be liable to determine expected required timing as per provided inputs to the signal which is vehicles count. The input of these systems is vehicles counts on each side of the road from crossing signal. And this input will be determined on much time is to be provided.

2.1 GOALS

I. To build a smart Garbage System that would

II. To build a smart parking lot system which would automatically tell passengers where to park in a parking lot.]

III. To Build a Smart Doorlock System that would help secure a household from theft and burglary and also ease execution of locks and keys.

IV. To develop a Smart Gardening system that would help individuals to monitor their crops

V. To develop a smart CCTV system that would help analyse surroundings more efficiently.

VI. To develop a smart streetlight system that would automatically save energy and contribute to the society.

VII. To develop a smart Air Pollution Monitor that would help keep a check on the pollution levels in a city.

VIII. To develop a smart Traffic Light System that would help users to swiftly move across streets.

IX. To unite all the above goals and develop a fully functional city.

3. TECHNICAL SPECIFICATIONS

SOFTWARES USED:

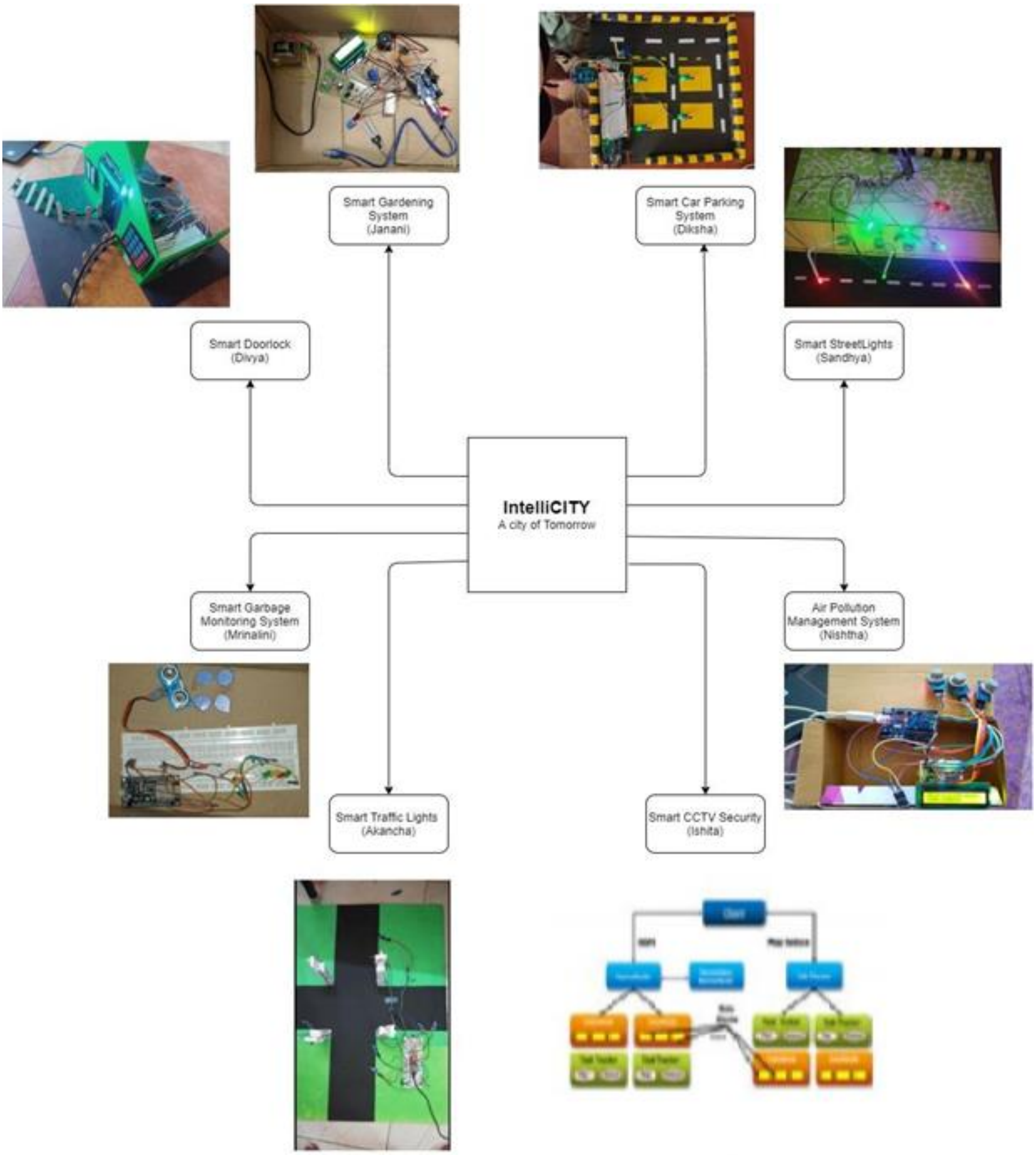
1. Arduino
2. Python IDE
3. Android Studio
4. Visual Studio Code
5. Git
6. RHEL
7. Hadoop
8. Python36
9. Virtual Box
10. Jupyter

HARDWARES USED:

1. Arduino
2. Webcam
3. Mobile Phone
4. Servo Motors and buzzers
5. Jumper Wires
6. Wifi module ESP8266
7. 16X2 LCD
8. 1K ohm RESISTOR
9. 220 ohm resistor
10. Keypad
11. Node MCU
12. HID lights
13. IR sensor
14. Light sensor
15. NodeMCU LiLon V3
16. Ultrasonic sensor
17. Step down Transformer
18. Relay
19. Temperature and Humidity sensor (DHT11)
20. Soil moisture sensor
21. LCD display

4. DESIGN, APPROACH AND DETAILS

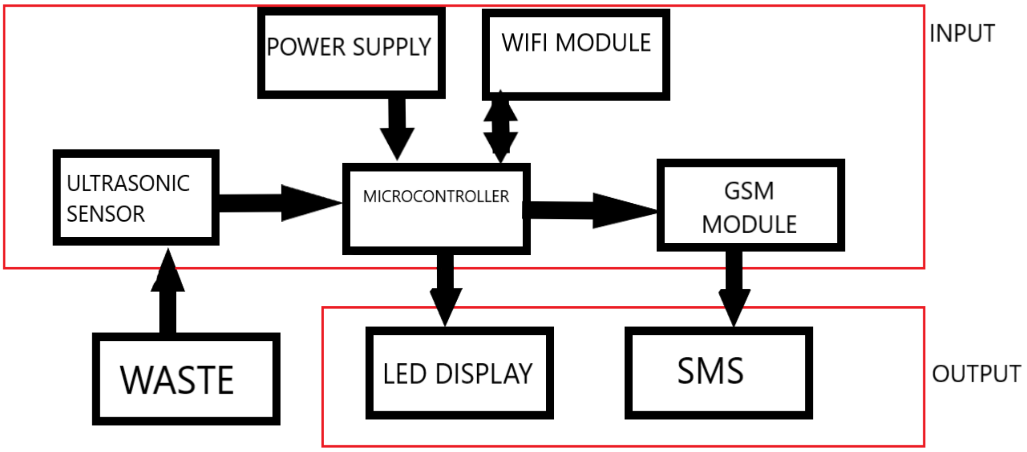
4.1 DESIGN


IntelliCITY
A city of Tomorrow
Smart Gardening System (Janani)
Smart Car Parking System (Diksha)
Smart Doorlock (Divya)
Smart StreetLights (Sandhya)
Smart Garbage Monitoring System (Mrinalini)
Air Pollution Management System (Nishtha)
Smart Traffic Lights (Akancha)
Smart CCTV Security (Ishita)

2.1 APPROACH AND DETAILS

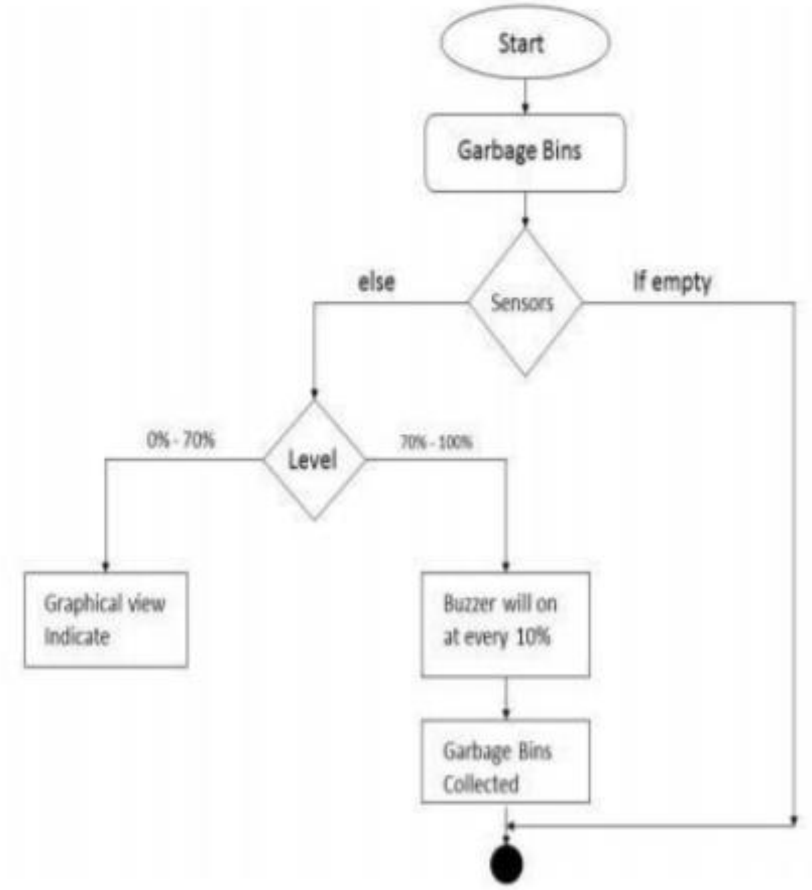
1. Garbage Monitoring System

ARCHITECTURE



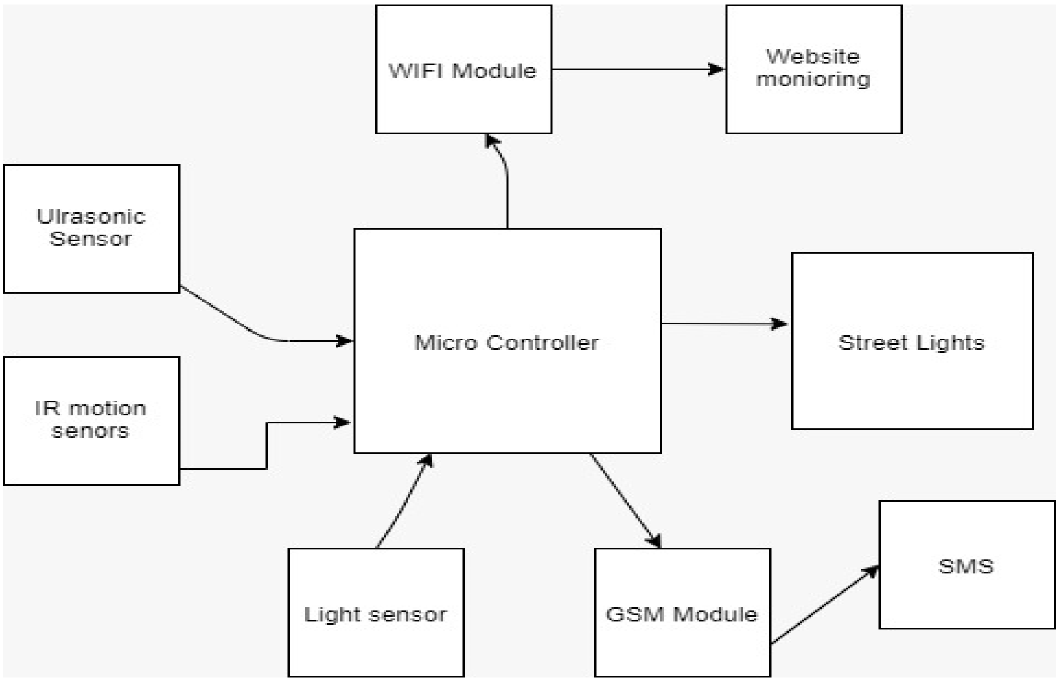
SMART GARBAGE MONITORING SYSTEM

FLOWCHART

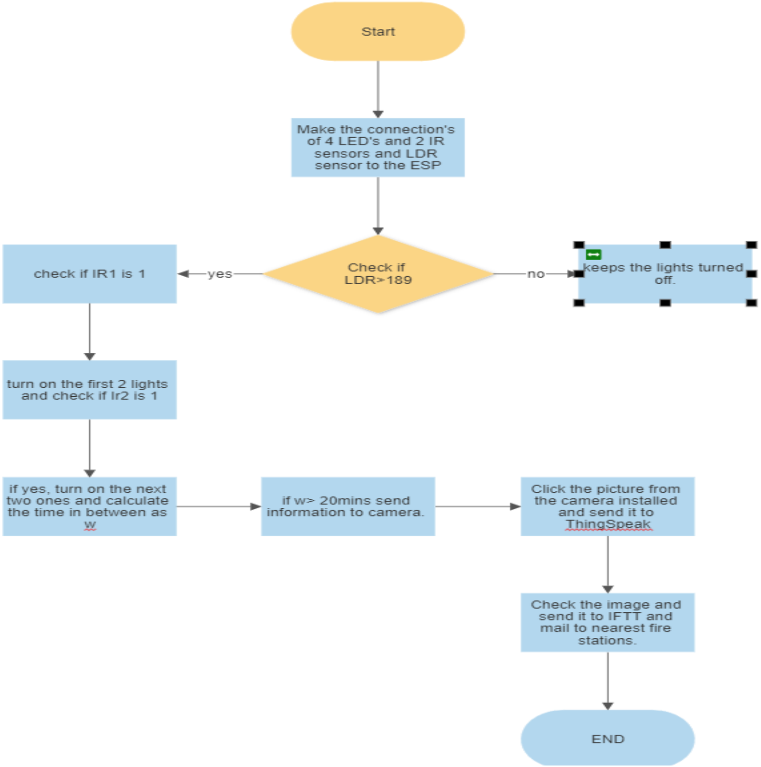


2. Smart Streetlights

ARCHITECTURE

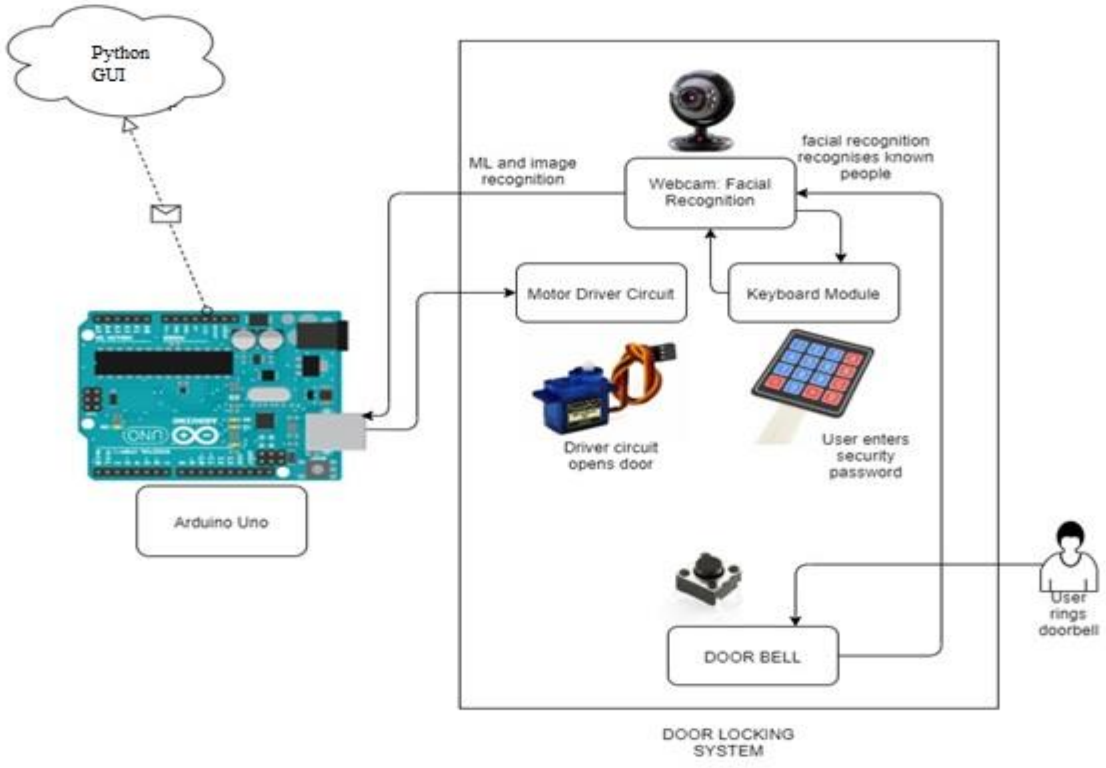


FLOWCHART

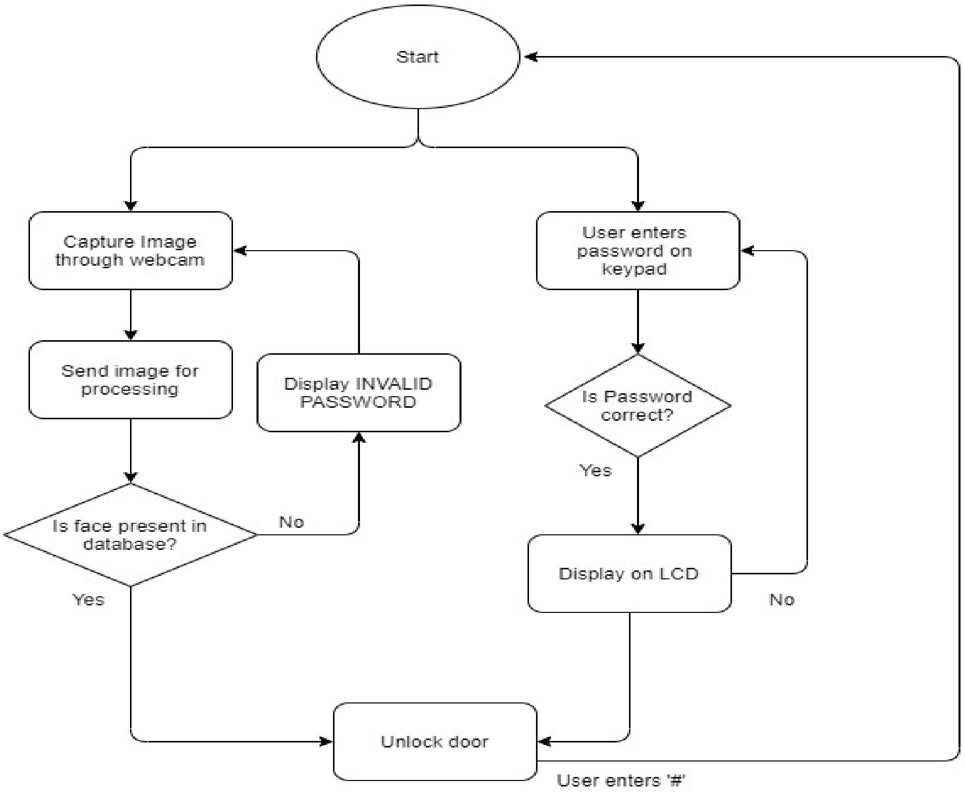

Start
Make the connection's of 4 LED's and 2 IR sensors and LDR sensor to the ESP
Check if LDR>189
yes
no
check if IR1 is 1
keeps the lights turned off.
turn on the first 2 lights and check if Ir2 is 1
if yes, turn on the next two ones and calculate the time in between as w
if w> 20mins send information to camera.
Click the picture from the camera installed and send it to ThingSpeak
Check the image and send it to IFTT and mail to nearest fire stations.
END

3. Smart DoorLock System

ARCHITECTURE

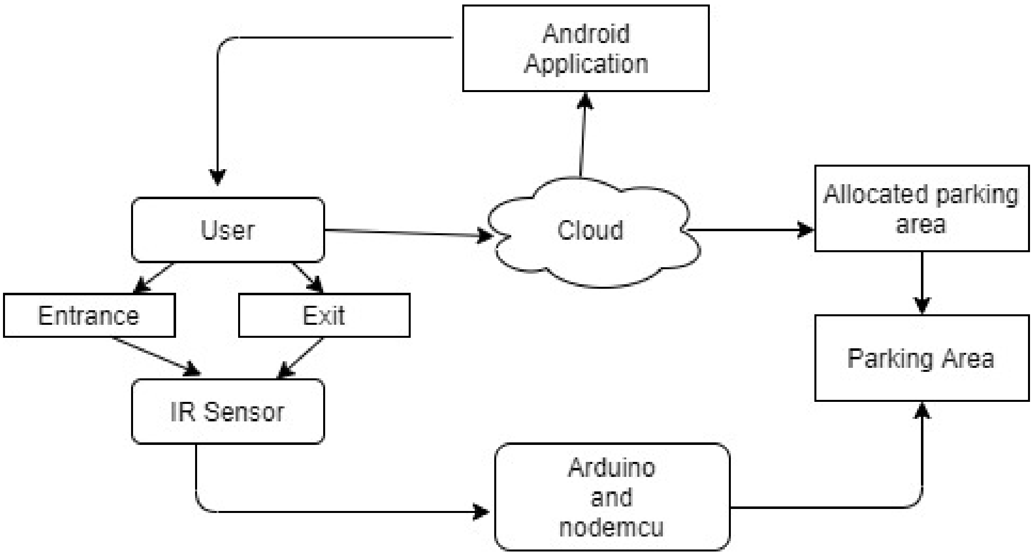


FLOWCHART

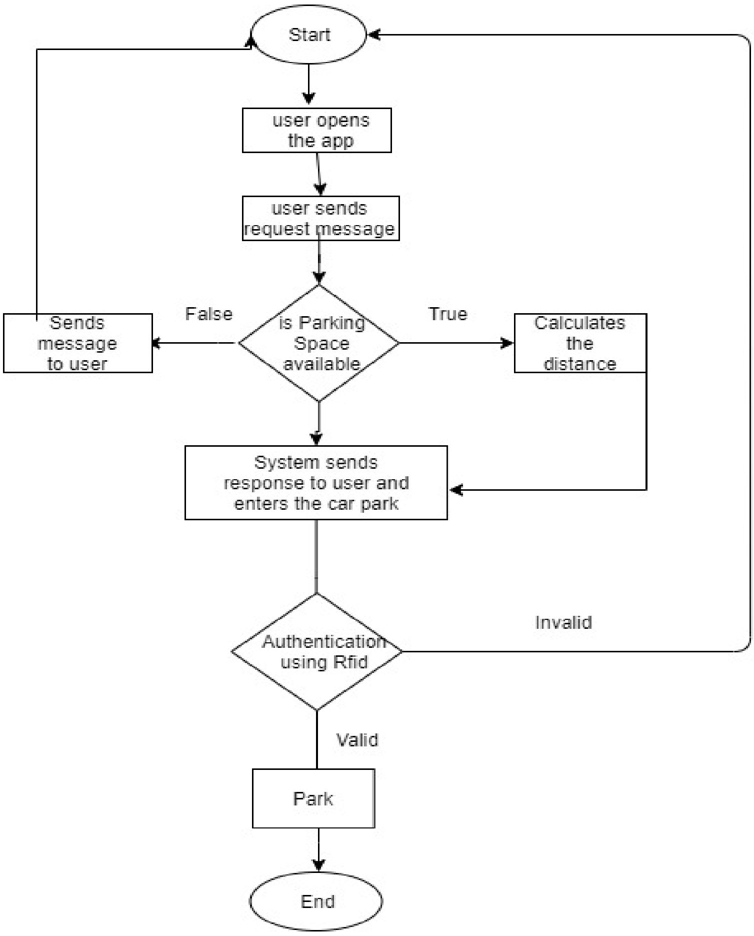

Start
Capture Image through webcam
Send image for processing
Display INVALID PASSWORD
Is face present in database?
No
Yes
User enters password on keypad
Is Password correct?
Yes
Display on LCD
No
Unlock door
User enters '#'

4. Smart Car-Parking System

ARCHITECTURE

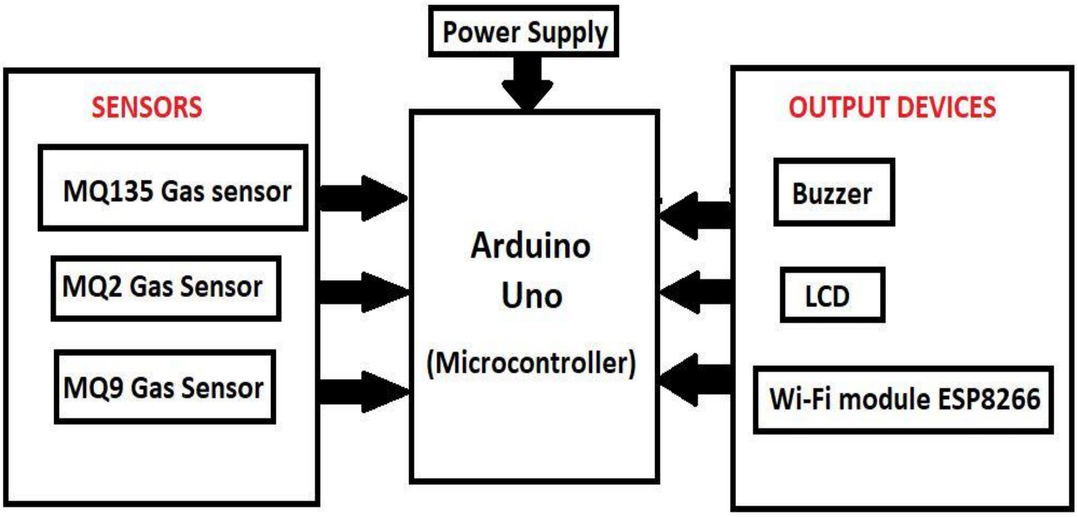


FLOWCHART

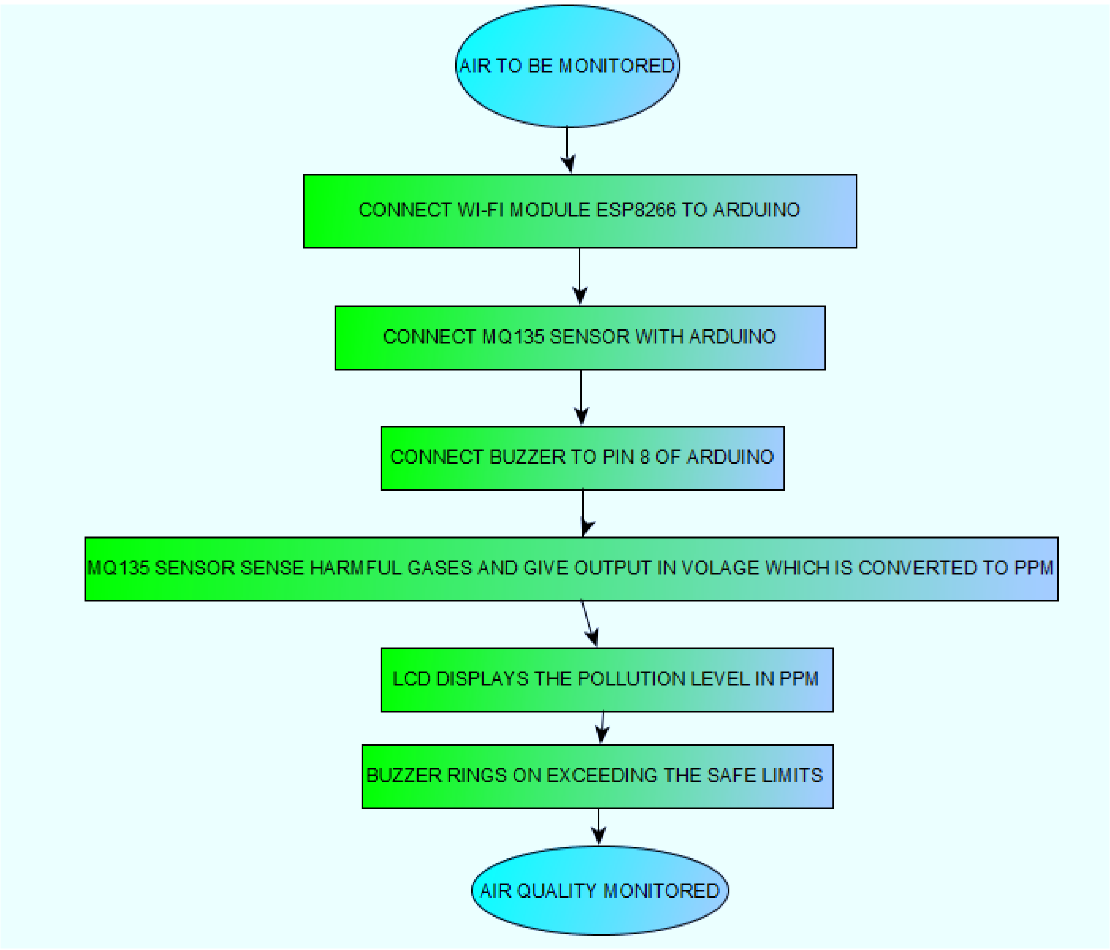

Start
user opens the app
user sends request message
False
is Parking Space available
True
Sends message to user
Calculates the distance
System sends response to user and enters the car park
Authentication using Rfid
Invalid
Valid
Park
End

5. Air Pollution System

ARCHITECTURE

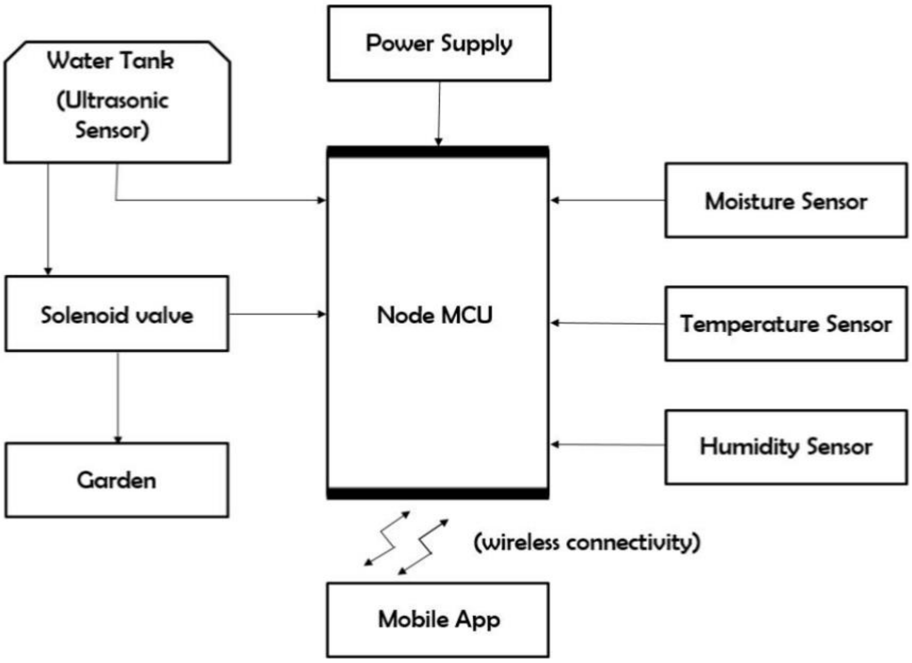


FLOWCHART

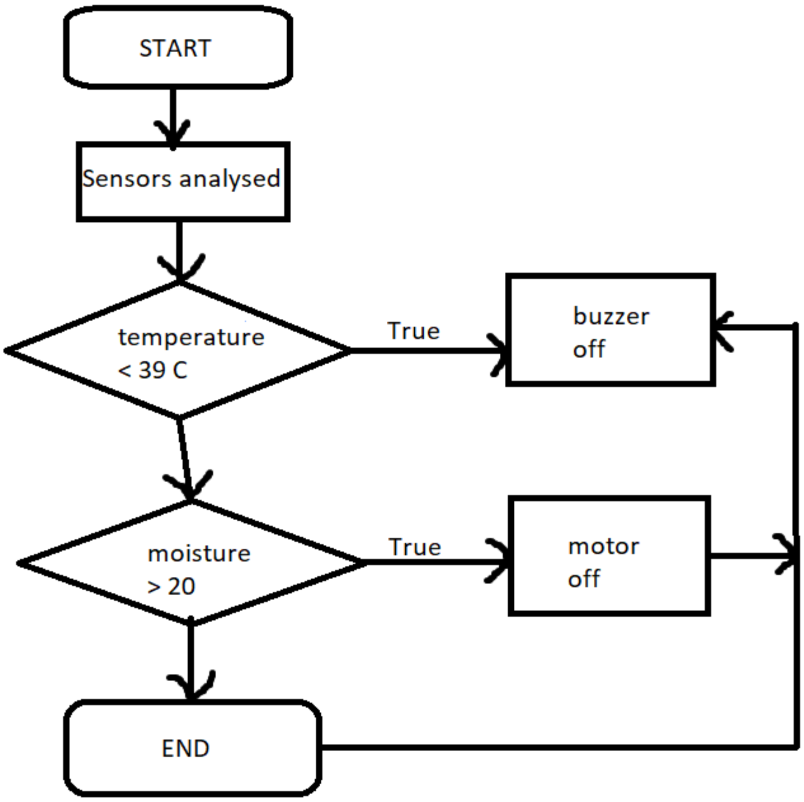

AIR TO BE MONITORED
CONNECT WI-FI MODULE ESP8266 TO ARDUINO
CONNECT MQ135 SENSOR WITH ARDUINO
CONNECT BUZZER TO PIN 8 OF ARDUINO
MQ135 SENSOR SENSE HARMFUL GASES AND GIVE OUTPUT IN VOLAGE WHICH IS CONVERTED TO PPM
LCD DISPLAYS THE POLLUTION LEVEL IN PPM
BUZZER RINGS ON EXCEEDING THE SAFE LIMITS
AIR QUALITY MONITORED

6. Smart Gardening System

ARCHITECTURE

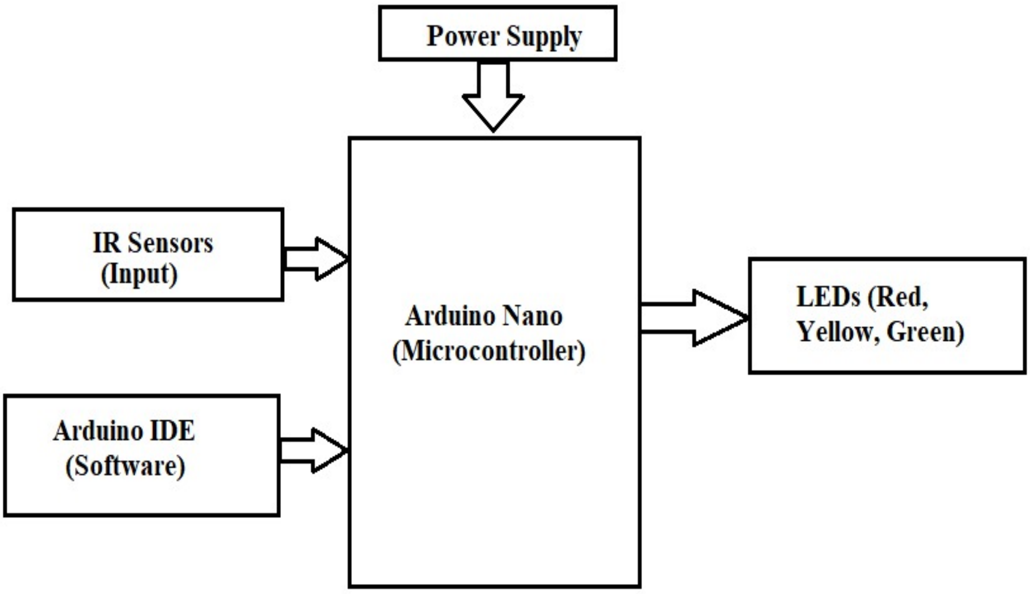


FLOWCHART

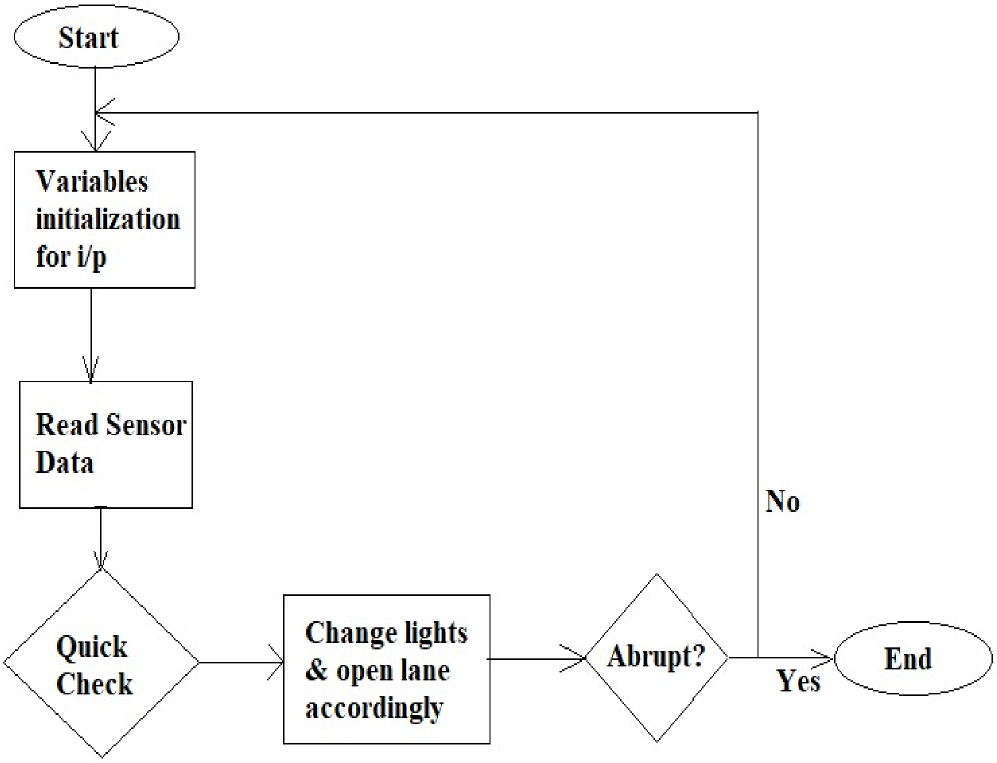


7. Smart Traffic Lights

ARCHITECTURE

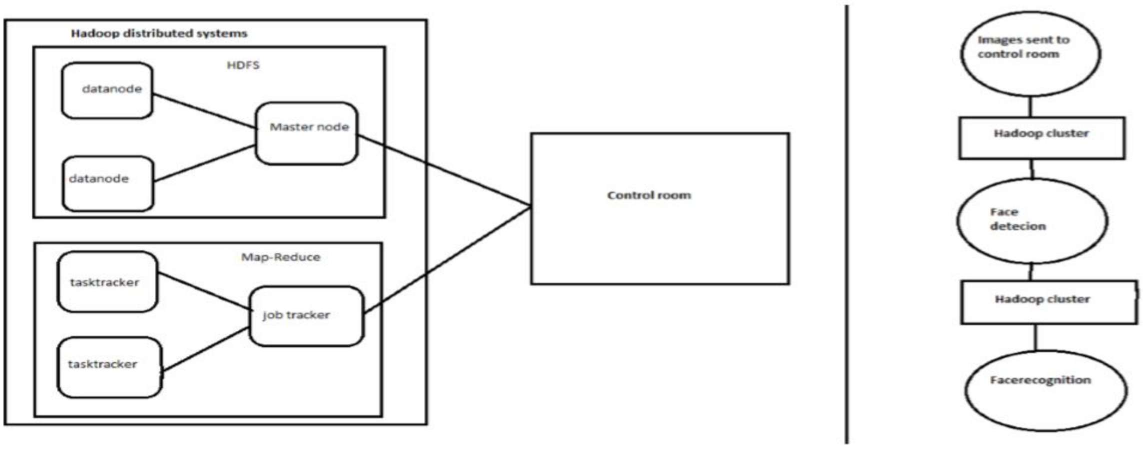


FLOWCHART

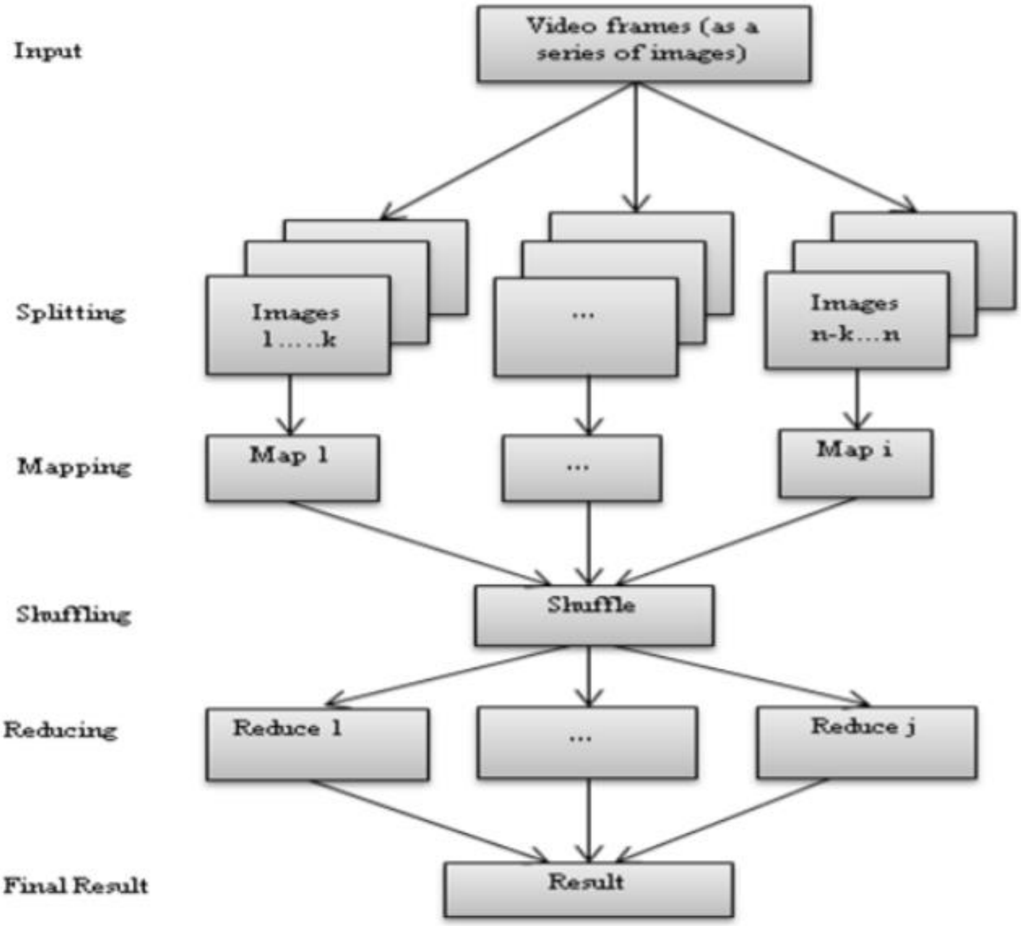

Start
Variables initialization for i/p
Read Sensor Data
Quick Check
Change lights & open lane accordingly
Abrupt?
No
Yes
End

8. Smart CCTV System

ARCHITECTURE



FLOWCHART



4.2Codes and Standards.

Some Code Snippets are given as follows:

2.1 APPROACH AND DETAILS

1. Garbage Monitoring System

```
#include <FirebaseArduino.h>
#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
const char* ssid = "TheAlgo";
const char* password = "project1234";
#define FIREBASE_HOST "friendlychat-6bd7a.firebaseio.com"
#defineFIREBASE_AUTH"WehKXhUO22JG8msnzGJaMXcBZPEke9JxC2BNMvFS"
#define pin1 D0
#define pin2 D1
#define pin3 D2
#define pin4 D3
#define pin5 D4
#define TRIGGER D7
#define ECHO D8
long maximum = 40;
long block = maximum/4;
bool started = true;
int id; int led = 0;
bool blinkAll = false;
bool blinkOn = false;
HTTPClient http;
void setup() {
Serial.begin(115200);
 Serial.println("started");

 pinMode(pin1, OUTPUT);
pinMode(pin2, OUTPUT); pinMode(pin3,
OUTPUT); pinMode(pin4, OUTPUT);
pinMode(pin5, OUTPUT);

 pinMode(TRIGGER,OUTPUT);
 pinMode(ECHO,INPUT);

 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());
 Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
Firebase.setFloat("bin001-max", maximum); if
(Firebase.failed()) {
 Serial.print("setting /number failed:");
Serial.println(Firebase.error());
 return;
 }
}
int getReading(){
 long duration, distance;
digitalWrite(TRIGGER, LOW);
 delayMicroseconds(2);

 digitalWrite(TRIGGER, HIGH);
 delayMicroseconds(10);
 digitalWrite(TRIGGER, LOW);
 duration = pulseIn(ECHO, HIGH);
distance = (duration/2) / 29.1;
 Serial.print(distance);
 Serial.println(" cm");
 return distance;
}
void displayLED(byte data){
digitalWrite(pin1, (data & 0b0001)>>0);
digitalWrite(pin2, (data & 0b0010)>>1);
digitalWrite(pin3, (data & 0b0100)>>2);
digitalWrite(pin4, (data & 0b1000)>>3);
}
long lastSent = 0;
long sendFirebaseInterval = 500;
void loop() { int dist =
getReading();
 if(dist > block)blinkAll = false; if(dist >
maximum)displayLED(0b0001); else
if(dist > block*3)displayLED(0b0010); else
if(dist > block*2)displayLED(0b0100); else
if(dist > block*1)displayLED(0b1000);
 else blinkAll = true;
 if(blinkAll){
blinkOn = !blinkOn;
 displayLED(blinkOn ? 0b1111 : 0b0000);
 }
 long current = millis();
 if((current - lastSent) >= sendFirebaseInterval){
 Firebase.setFloat("bin001-dist", dist);
 }
 delay(100);
}
```

2. Smart Streetlights

```
#include <ESP8266WiFi.h>;
#include <WiFiClient.h>;
#include <ThingSpeak.h>;
const char* ssid = "sandhya";
const char* password = "password1234";
WiFiClient client;
unsigned long myChannelNumber = 987641;
const char * myWriteAPIKey = "IMYD4FHSUTUS3W33";
const char * myReadAPIKey = "960PH1D9XXLC4ZD4";
int led_1;
int led_2;
int led_3;
int led_4;
int ir1 = D0;
int led1 = D5;
int led2 = D6;
int ir3 = D2;
int led3 = D7;
int led4 = D8;
int ldr = A0;
int val =0;
void setup() {
  Serial.begin(9600);
  //delay(10);
  pinMode(ir1,INPUT);
  pinMode(led1,OUTPUT);
  //pinMode(ir2,INPUT);
  pinMode(led2,OUTPUT);
  pinMode(ir3,INPUT);
  pinMode(led3,OUTPUT);
  pinMode(led4,OUTPUT);
  WiFi.begin(ssid, password);
  ThingSpeak.begin(client);
}
void loop() {
  int s1 = digitalRead(ir1);
  //int s2 = digitalRead(ir2);
  s1=not(s1);
  int s3 = digitalRead(ir3);
  s3 = not(s3);
  val = analogRead(ldr);
  Serial.print(s1);
  Serial.print(":");
  //Serial.print(s2);
  //Serial.print(":");
  Serial.print(s3);
  Serial.print(":");
  Serial.println(val);
    digitalWrite(led1,LOW);
    digitalWrite(led2,LOW);
    digitalWrite(led3,LOW);
    digitalWrite(led4,LOW);
  if(val>180)
  {
    if(s1==1){
    digitalWrite(led1,HIGH);
    digitalWrite(led2,HIGH);
    }
    else{
    digitalWrite(led1,LOW);
    digitalWrite(led2,LOW);
    }
    if(s3==1){
    digitalWrite(led3,HIGH);
    digitalWrite(led4,HIGH);
    }
    else{
    digitalWrite(led3,LOW);
    digitalWrite(led4,LOW);
    }
  }
  else
  {
    digitalWrite(led1,LOW);
    digitalWrite(led2,LOW);
    digitalWrite(led3,LOW);
    digitalWrite(led4,LOW);
  }
```

3. Smart DoorLock System

TRAINING.py

```
import cv2,os
import numpy as np
from PIL import Image
# recognizer = cv2.face.LBPHFaceRecognizer_create()
recognizer=cv2.face.createFisherFaceRecognizer_create()
detector= cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
def getImagesAndLabels(path):
    #get the path of all the files in the folder
    imagePaths=[os.path.join(path,f) for f in os.listdir(path)]
    #create empth face list
    faceSamples=[]
    #create empty ID list
    Ids=[]
    #now looping through all the image paths and loading the Ids and the images
    for imagePath in imagePaths:
        #loading the image and converting it to gray scale
        pilImage=Image.open(imagePath).convert('L')
        #Now we are converting the PIL image into numpy array
        imageNp=np.array(pilImage,'uint8')
        #getting the Id from the image
        Id = int(os.path.split(imagePath)[-1].split(".")[1])
        # extract the face from the training image sample
        faces=detector.detectMultiScale(imageNp)
        #If a face is there then append that in the list as well as Id of it
        for (x,y,w,h) in faces:
            faceSamples.append(imageNp[y:y+h,x:x+w])
            Ids.append(Id)
    return faceSamples,Ids
faces,Ids = getImagesAndLabels('TrainingImage')
recognizer.train(faces, np.array(Ids))
recognizer.save('TrainingImageLabel/trainner.yml')
```

TESTING.py

```
import cv2
import numpy as np
recognizer = cv2.face.LBPHFaceRecognizer_create()
recognizer.read('TrainingImageLabel/trainner.yml')
cascadePath = "haarcascade_frontalface_default.xml"
faceCascade = cv2.CascadeClassifier(cascadePath)
font = cv2.FONT_HERSHEY_SIMPLEX
cam = cv2.VideoCapture(0)
while True:
    ret, im =cam.read()
    gray=cv2.cvtColor(im,cv2.COLOR_BGR2GRAY)
    faces=faceCascade.detectMultiScale(gray, 1.2,5)
    for(x,y,w,h) in faces:
        Id, conf = recognizer.predict(gray[y:y+h,x:x+w])
        # # else:
        # #     Id="Unknown"
        # cv2.rectangle(im, (x-22,y-90), (x+w+22, y-22), (0,255,0), -1)
        cv2.rectangle(im, (x, y), (x + w, y + h), (0, 260, 0), 7)
        cv2.putText(im, str(Id), (x,y-40),font, 2, (255,255,255), 3)
        # cv2.putText(im, str(Id), (x + h, y), font, 1, (0, 260, 0), 2)
    cv2.imshow('im',im)
    if cv2.waitKey(10) & 0xFF==ord('q'):
        break
cam.release()
cv2.destroyAllWindows()
```

4. Smart Car-Parking System

```
#include <Firebase.h>
#include <FirebaseArduino.h>
#include <FirebaseCloudMessaging.h>
#include <FirebaseError.h>
#include <FirebaseHttpClient.h>
#include <FirebaseObject.h>
#include <ESP8266WiFi.h>
#include <FirebaseArduino.h>
#define FIREBASE_HOST "trial1-2fb27.firebaseio.com"
#define FIREBASE_AUTH "ilfI8E06mRrK4DoOvKaXOPwYwEEhWzMuP9yjPTn4"
#define WIFI_SSID "diksha"
#define WIFI_PASSWORD "udontneedtoknow"
const int ProxSensor0=D7;
int inputVal0 = 0;
const int ProxSensor1=D6;
int inputVal1 = 0;
const int ProxSensor2=D5;
int inputVal2 = 0;
const int ProxSensor3=D2 ;
int inputVal3 = 0;
void setup() {
  Serial.begin(9600);
    // connect to wifi.
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("connecting");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  Serial.println();
  Serial.print("connected: ");
  Serial.println(WiFi.localIP());
  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
}
int n = 0;
void loop() {
  inputVal0 = digitalRead(ProxSensor0);
  inputVal1 =  digitalRead(ProxSensor1);
  inputVal2 =  digitalRead(ProxSensor2);
  inputVal3 =  digitalRead(ProxSensor3);
  Serial.println(inputVal0);
  Serial.println(inputVal1);
  Serial.println(inputVal2);
  Serial.println(inputVal3);
  // set value
  Firebase.setInt("ir1", inputVal0);
  Firebase.setInt("ir2",inputVal1);
  Firebase.setInt("ir3",inputVal2);
  Firebase.setInt("ir4",inputVal3);
  // handle error
  delay(1000);
  // get value
  Serial.print("ir1: ");
  Serial.println(Firebase.getInt("ir1"));
  Serial.print("ir2: ");
  Serial.println(Firebase.getInt("ir2"));
  Serial.print("ir3: ");
  Serial.println(Firebase.getInt("ir3"));
  Serial.print("ir4: ");
  Serial.println(Firebase.getInt("ir4"));
  delay(1000);
  if (Firebase.failed()) {
      Serial.print("setting /message failed:");
      Serial.println(Firebase.error());
      return;
  }
  delay(1000);
   // append a new value to /logs
  // handle error
  if (Firebase.failed()) {
      Serial.print("pushing /logs failed:");
      Serial.println(Firebase.error());
      return;
  }
  delay(1000);
}
```

5. Air Pollution System

```
#include <LiquidCrystal.h>
#include "ESP8266.h"
#define SSID "Redmi"     // "SSID-WiFiname"
#define PASS "nishtha"       // "password"
#define IP "184.106.153.149"// thingspeak.com ip
String msg = "GET /update?key=Z4W7Y091E1GQSNWG";
 //change it with your api key like "GET /update?key=Your Api Key"
#define QUALITY_PIN A0 //AIR QUALITY
#define METHANE_PIN A1 //MQ2 Methane
#define CO_PIN A2 //MQ9 Carbon Monoxide
ESP8266 ESP;
LiquidCrystal lcd(7, 6, 5, 4, 3, 2);
int qualityLevel;
int coLevel;
int methaneLevel;
void updateLevel(){
  String cmd = msg ;
  cmd += "&field1=";
  cmd += qualityLevel;
  cmd += "&field2=";
  cmd += coLevel;
  cmd += "&field3=";
  cmd += methaneLevel;
  cmd += "\r\n";
  if(ESP.get(IP, 80, cmd)){
//    ESP.serialDebug();
  }
}
void setup() {
  Serial.begin(115200);
  lcd.begin(16, 2);
  lcd.print("Initializing!");
  ESP.preInit();
  while(ESP.connectWiFi(SSID, PASS)){
    Serial.print("Trying..");
    delay(500);
  }
  lcd.setCursor(0, 1);
  lcd.print("** ALL OK **");
  delay(2000);
}
void loop() {
  qualityLevel = analogRead(QUALITY_PIN);
  qualityLevel = map(qualityLevel, 0, 1023, 0, 100);
  qualityLevel=100-qualityLevel;
  coLevel = analogRead(CO_PIN);
  coLevel = (100-map(coLevel, 0, 1023, 0, 100));

//  float sensor_volt = coLevel / 1024 * 5.0;
//  sensor_volt = (5.0-sensor_volt)/sensor_volt; // omit * RL
//  coLevel = sensor_volt / 9.8;
 // The ratio of RS/R0 is 9.8 in a clear air from Graph (Found using WebPlotDigitizer)
  methaneLevel = analogRead(METHANE_PIN);
  methaneLevel = map(methaneLevel, 0, 1023, 0, 100)*10;
  Serial.println("Read sensor: ");
  Serial.print("Quality : ");
  Serial.print(qualityLevel);
  Serial.print("% CO : ");
  Serial.print(coLevel);
  Serial.print("ppm CH4 : ");
  Serial.print(methaneLevel);

  Serial.println("ppm");
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Qty:");
  lcd.print(qualityLevel);
  lcd.print("%|CO:");
  lcd.print(coLevel);
  lcd.print("ppm");
  lcd.setCursor(0, 1);
  lcd.print("CH4:");
  lcd.print(methaneLevel);
  lcd.print("ppm");
  if(qualityLevel<50){
      Serial.println("*Danger*");
  }
  updateLevel();
  delay(2000);
}
```

6. Smart Gardening System

```
#include <SoftwareSerial.h>  //including the software serial UART library which will make the digital pins as TX and RX
#include <LiquidCrystal.h>
#include "DHT.h"            //including the DHT22 library
#define DHTPIN 7            //Declaring pin 8 of arduino to communicate with DHT22
#define DHTTYPE DHT11       //Defining type of DHT sensor we are using (DHT22 or DHT11)
#define DEBUG true
DHT dht(DHTPIN, DHTTYPE);    //Declaring a variable named dht
LiquidCrystal lcd(13, 12, 11, 10, 9, 8);
SoftwareSerial esp8266(2,3); //Connect the TX pin of ESP8266 to pin 2 of Arduino and RX pin of ESP8266 to pin 3 of Arduino.
SoftwareSerial mySerial(1, 0);

int motor=A5;
//const int high = 5;
//const int low = 4;
int buz=6;
float temp_read,Temp_alert_val,Temp_shut_val;
float temperature, humidity;
float soil,soill;
int buttonState1=0;
int buttonState2=0;
void setup()
{
  Serial.begin(9600);
    lcd.begin(16,2);
   mySerial.begin(9600);
  esp8266.begin(115200); // Set the baud rate of serial communication
 // dht.begin();         //This will initiate receiving data from DHT22
// pinMode(high, INPUT);
//pinMode(low, INPUT);
   pinMode(motor,OUTPUT);
    pinMode(buz,OUTPUT);
  sendData("AT+RST\r\n",2000,DEBUG);            // Reset the module
  sendData("AT+CWMODE=2\r\n",1000,DEBUG);       // Configure ESP8266 as an access point
  sendData("AT+CIFSR\r\n",1000,DEBUG);          // Get the IP address of ESP8266
  sendData("AT+CIPMUX=1\r\n",1000,DEBUG);       // Configure ESP8266 for multiple connections
  sendData("AT+CIPSERVER=1,80\r\n",1000,DEBUG); // Start TCP server at port 80

lcd.setCursor(0,0); // Sets the location at which subsequent text written to the LCD will be displayed
lcd.print("Tem: ");
lcd.setCursor(11,0); // Sets the location at which subsequent text written to the LCD will be displayed
lcd.print("H: ");
lcd.setCursor(0,1); // Sets the location at which subsequent text written to the LCD will be displayed
lcd.print("Moist: ");

//lcd.setCursor(0,1); // Sets the location at which subsequent text written to the LCD will be displayed
//lcd.print("Animal: ");
}
```

7. Smart Traffic Lights

```
void readSensor()
{
  a1 = analogRead(A7);
  a2 = analogRead(A6);
  b1 = analogRead(A4);
  b2 = analogRead(A5);
  c1 = analogRead(A1);
  c2 = analogRead(A0);
  d1 = analogRead(A3);
  d2 = analogRead(A2);

  if (a1 < 400) a1 = 1; else a1 = 0; if (a2 < 400) a2 = 1; else a2 = 0;
if (b1 < 400) b1 = 1; else b1 = 0; if (b2 < 400) b2 = 1; else b2 = 0;
if (c1 < 400) c1 = 1; else c1 = 0; if (c2 < 400) c2 = 1; else c2 = 0;
if (d1 < 400) d1 = 1; else d1 = 0; if (d2 < 400) d2 = 1; else d2 = 0;

Serial.print(a1);
Serial.print("\t");
Serial.print(a2);
Serial.print("\t");
Serial.print(b1);
Serial.print("\t");
Serial.print(b2);
Serial.print("\t");
Serial.print(c1);
Serial.print("\t");
Serial.print(c2);
Serial.print("\t");
Serial.print(d1);
Serial.print("\t");
Serial.print(d2);
Serial.println("\t");

}

void roadAopen()
{
  digitalWrite(ledA3, LOW);

  digitalWrite(ledA1, HIGH);
  digitalWrite(ledB3, HIGH);
  digitalWrite(ledC3, HIGH);
  digitalWrite(ledD3, HIGH);
  delay(10000);
  digitalWrite(ledA1, LOW);
  digitalWrite(ledA2, HIGH);
  delay(1000);
  digitalWrite(ledA2, LOW);
  readSensor();
}

void roadBopen()
{
  digitalWrite(ledB3, LOW);

  digitalWrite(ledA3, HIGH);
  digitalWrite(ledB1, HIGH);
  digitalWrite(ledC3, HIGH);
  digitalWrite(ledD3, HIGH);
  delay(10000);
  digitalWrite(ledB1, LOW);
  digitalWrite(ledB2, HIGH);
  delay(1000);
  digitalWrite(ledB2, LOW);
  readSensor();
}
```

8. Smart CCTV System

```
import subprocess as s
import re
#if(ch==1):
print("Hadoop Setup")

IP=list()
pingname=list()


print("Name Node details-")
IP.append(input("Enter IP: "))
pingname.append(input("Enter the ping name: "))

print("Client Node details")
IP.append(input("Enter IP: "))
pingname.append(input("Enter the ping name: "))


print("Data Nodes details-")
num_dn=input("Enter the number of data nodes in the cluster you prefer to have:")
num_dn=int(num_dn)

for i in range(num_dn):
	print("Enter the IP and the ping name of data node {}-".format((i+1)))
	IP.append(input('Enter IP: '))
	pingname.append(input('Enter ping name: '))
	print(IP[i+2])
	print(pingname[i+2])
print("Basic draft ready")

len_IP=len(IP)


print("------------------")
print("SERVER")
print("------------------")
print("Let's set up Hadoop")
print("step-1 : Let's check network connectivity")

flag_net=True


for i in range(len_IP):

	output =s.getoutput("ping {} -c1".format(IP[i]))
```

```
        if (re.search('ttl',output)):
                print("{} OK".format(pingname[i]))
        else:
                print("Network problem with {}".format(pingname[i]))
                flag_net=False


if(flag_net==False):
        print("  Please rectify the above errors befor continuing ")
        s.getoutput('exit')
else:
        print("Congrats Step-1 working fine\n ")
```

4.3Constraints, alternatives and Trade-off

Infrastructure:

Smart Cities utilize sensor technology to gather and analyze information in an effort to improve the quality of life for residents. Sensors collect data on everything from rush hour stats to crime rates to overall air quality. Complicated and costly infrastructure is involved in installing and maintaining these sensors.

Educating and engaging the community:

For a Smart City to truly exist and thrive, it needs “smart” citizens who are engaged and actively taking advantage of new technologies. With any new city-wide tech project, part of the implementation process must involve educating the community on its benefits. This can be done through a series of in-person town hall-style meetings and email campaigns with voter registration, as well as an online education platform that keeps citizens engaged and up-to-date.

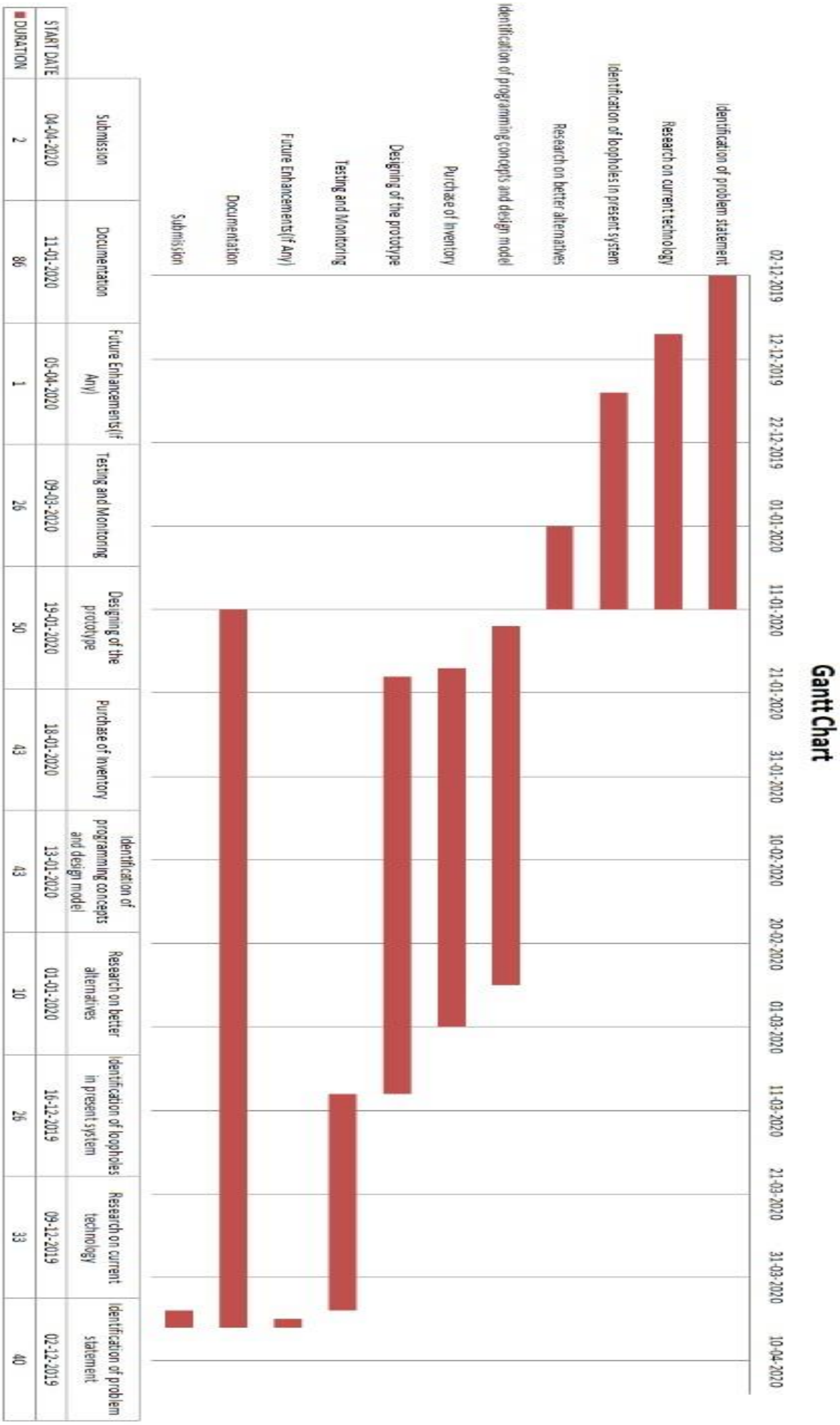
A wide area of Internet Connection:

To collect and send data from the sensors to the cloud or vice- versa or even to the user a high speed internet connection is needed in a wide area for proper working of the smart city.

Digital Security:

With the advent of technologies comes a lot of risk and here since everything is over the cloud. The government needs to ensure a digital security for its people so they can feel safe and not feel that their privacy is being violated at any time

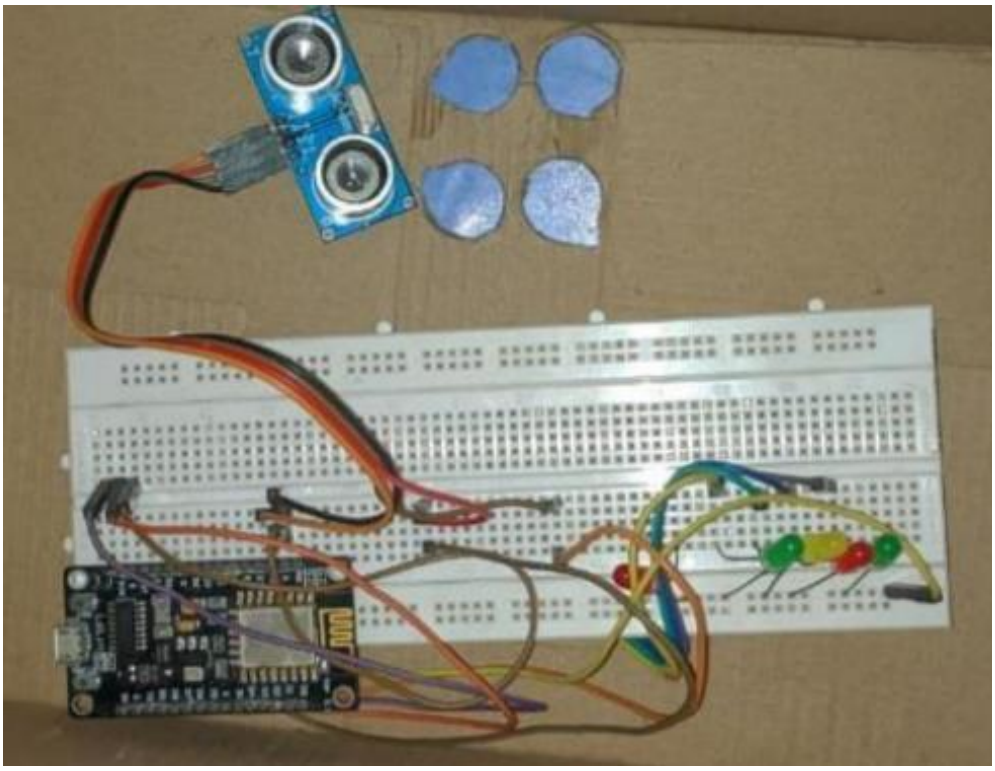
5. SCHEDULE TASKS AND MILESTONES


Gantt Chart

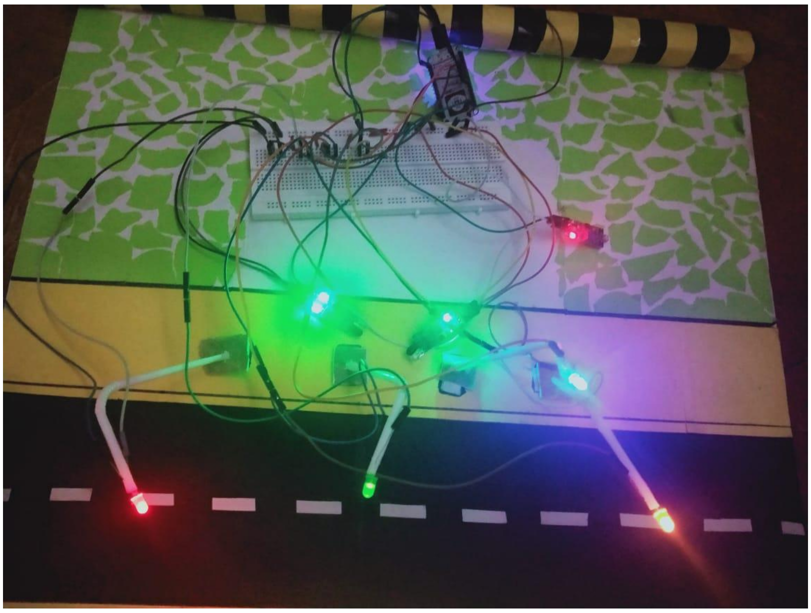
	Submission	Documentation	Future Enhancements(if Any)	Testing and Monitoring	Designing of the prototype	Purchase of Inventory	Identification of programming concepts and design model	Research on better alternatives	Identification of loopholes in present system	Research on current technology	Identification of problem statement
START DATE	04-04-2020	11-01-2020	05-04-2020	09-03-2020	19-01-2020	18-01-2020	13-01-2020	01-01-2020	16-12-2019	09-12-2019	02-12-2019
■ DURATION	2	86	1	26	50	43	43	10	26	33	40

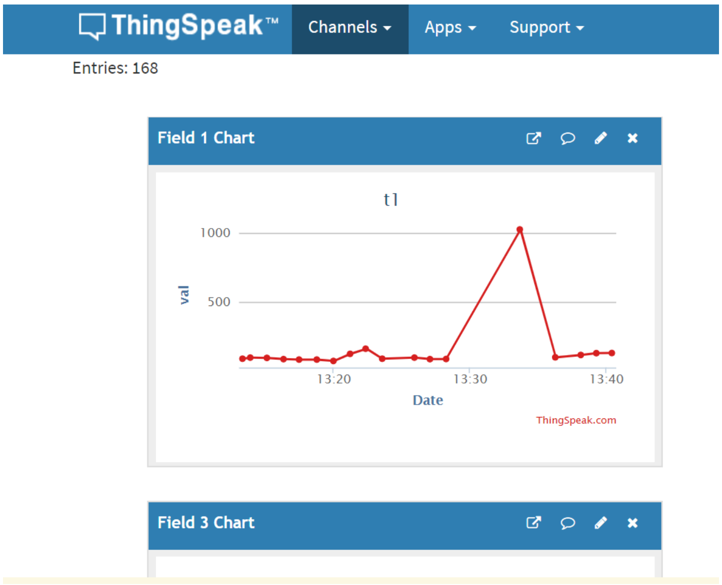
6. PROJECT DEMONSTRATION

1. Garbage Monitoring System

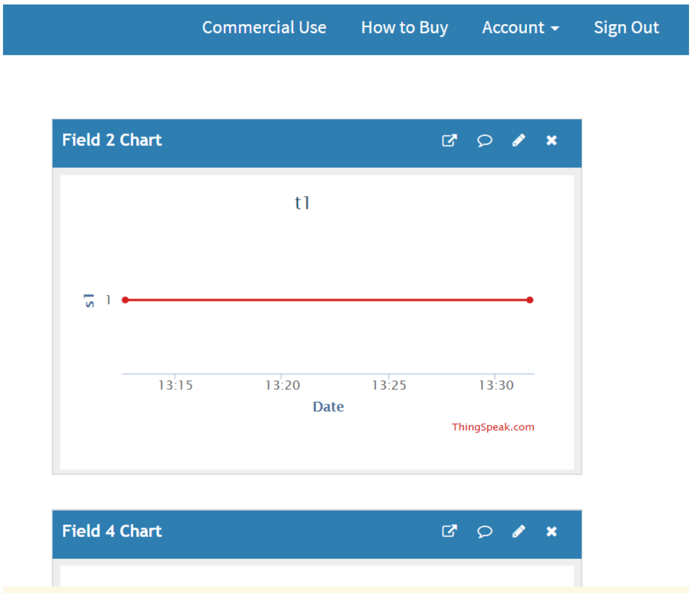


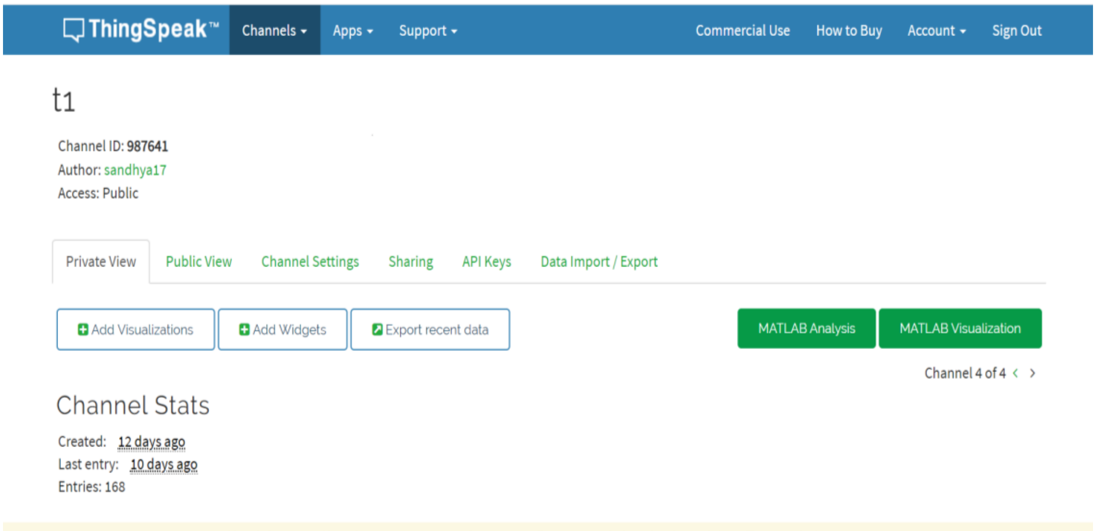
2. Smart Streetlights





Entries: 168





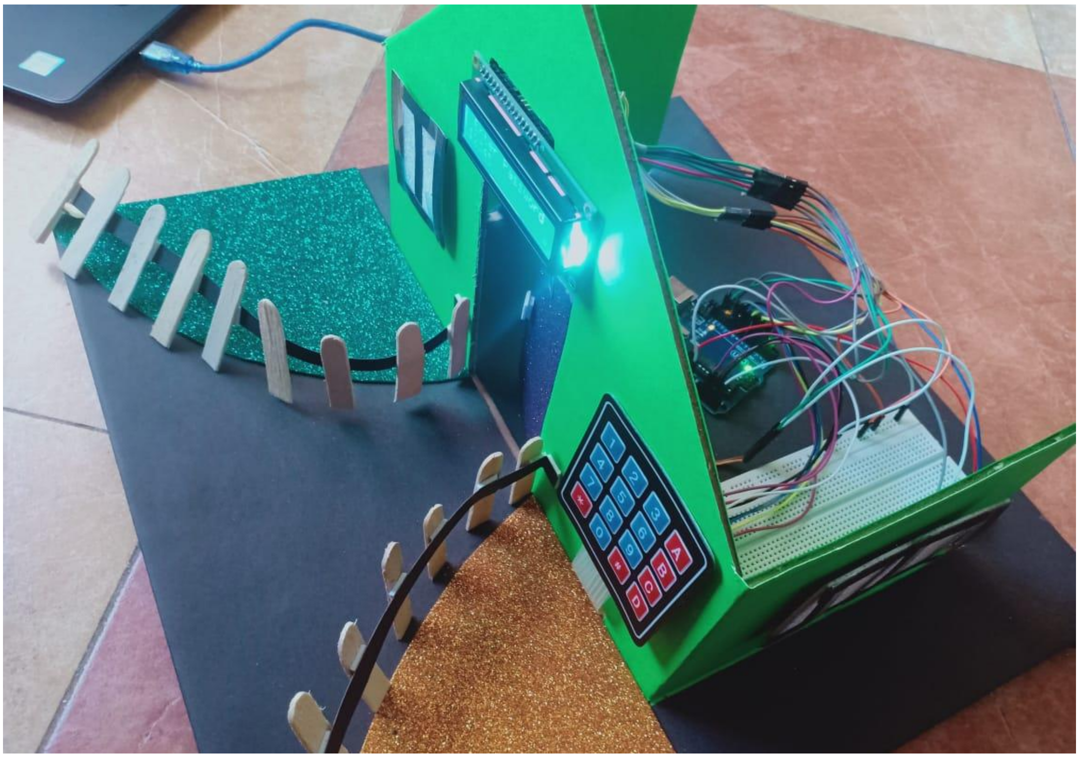
t1

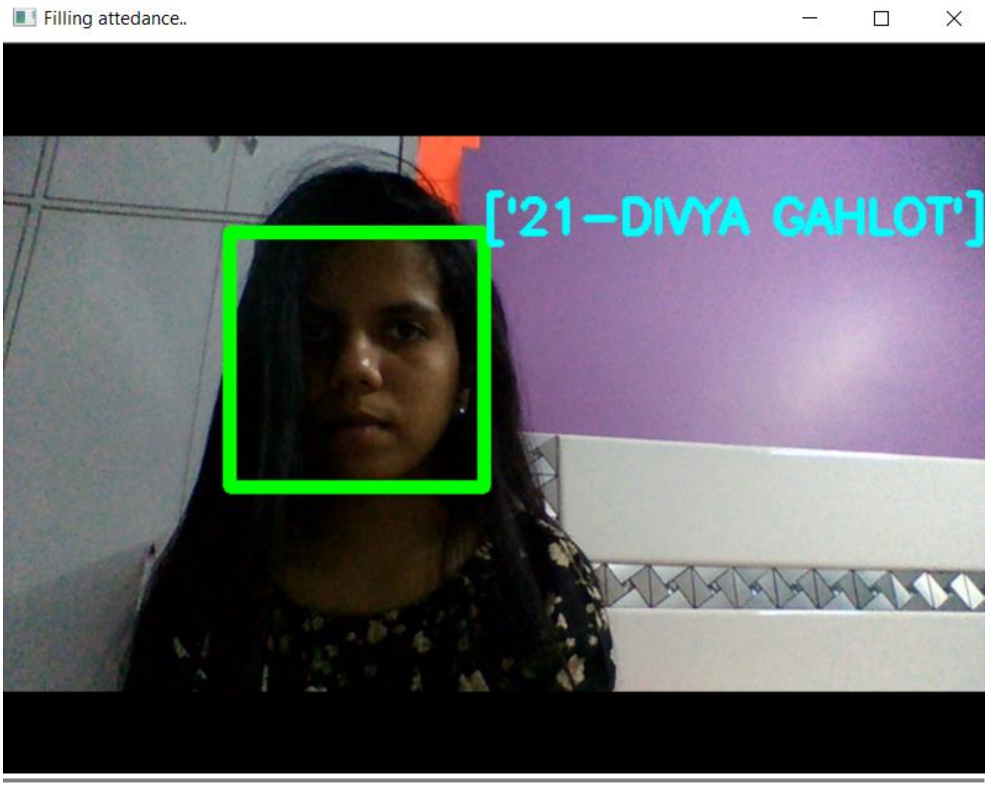
Channel ID: **987641**
Author: sandhya17
Access: Public

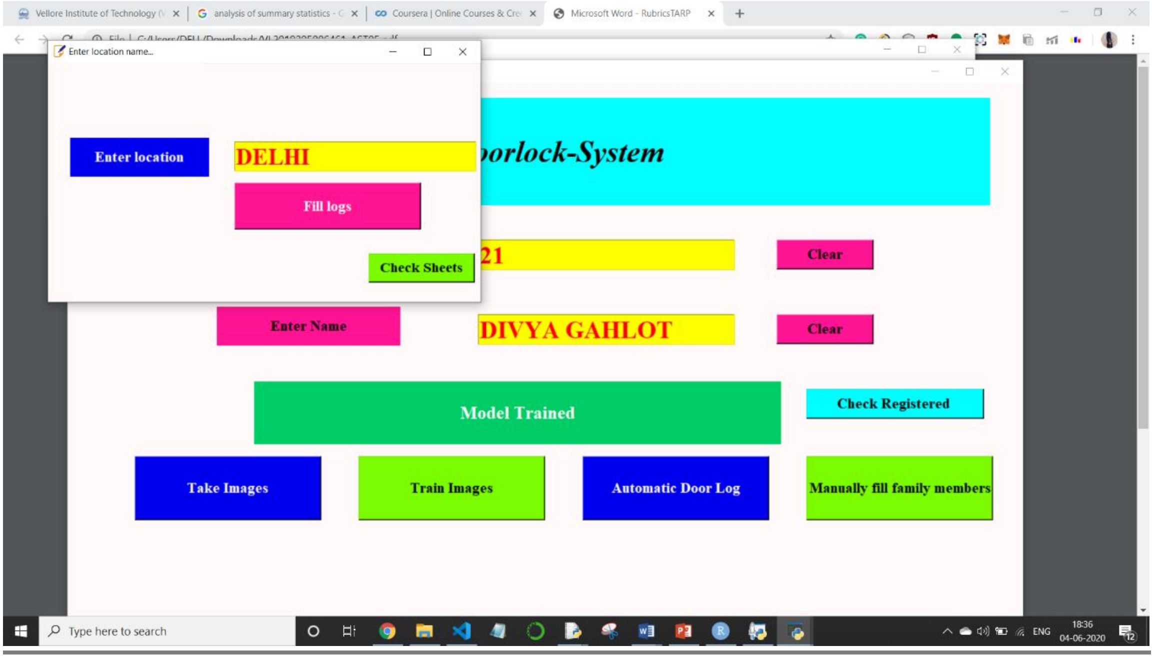
Channel Stats

Created: 12 days ago
Last entry: 10 days ago
Entries: 168

3. Smart DoorLock System



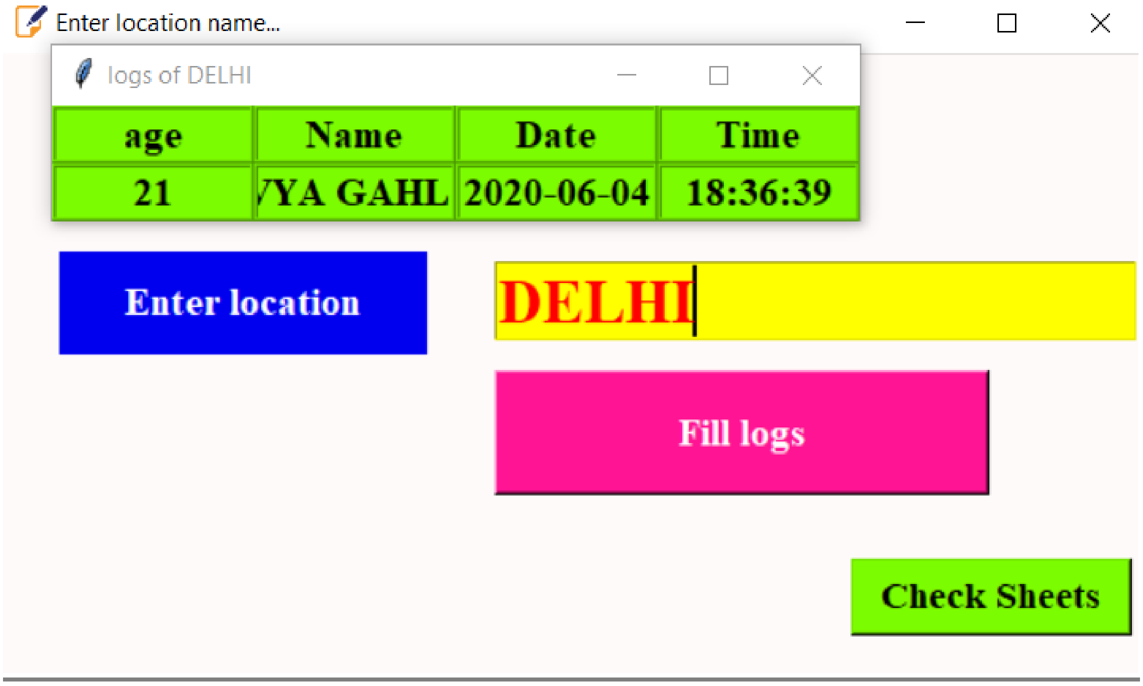




Enter location: DELHI

21

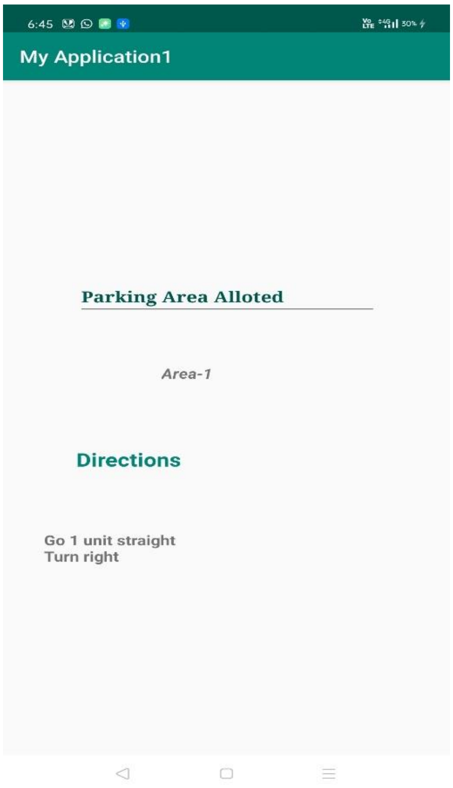
Enter Name: DIVYA GAHLOT



age	Name	Date	Time
21	VYA GAHL	2020-06-04	18:36:39

Enter location: DELHI

4. Smart Car-Parking System

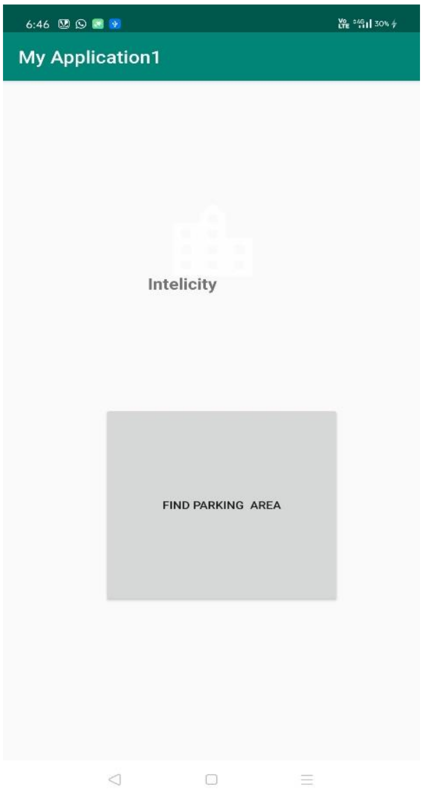


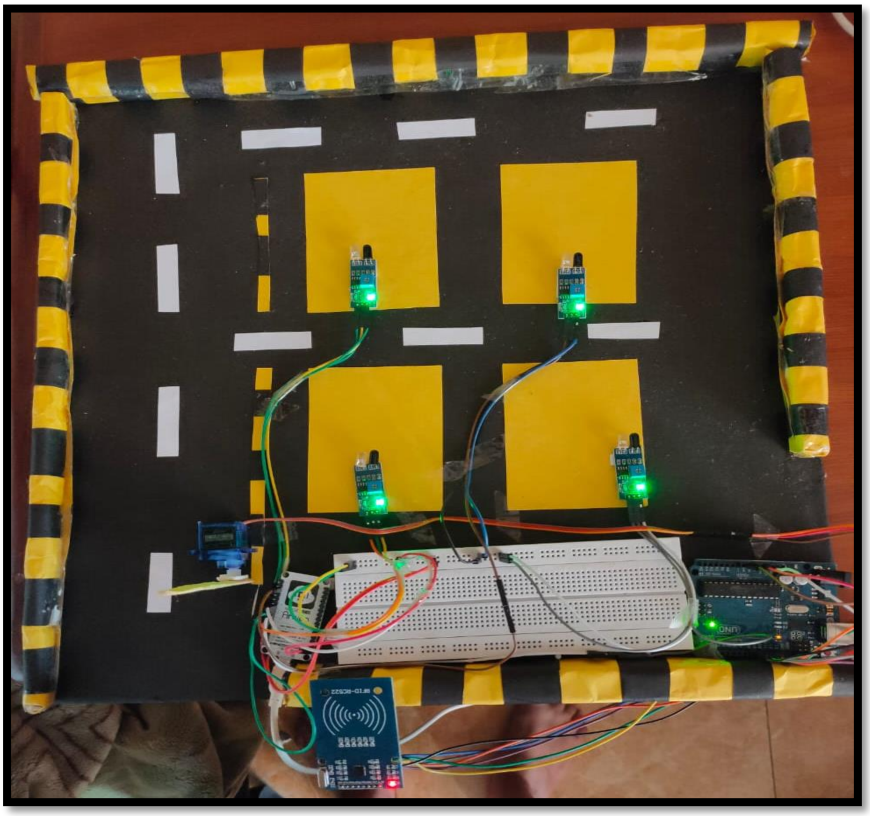
Parking Area Alloted

Area-1

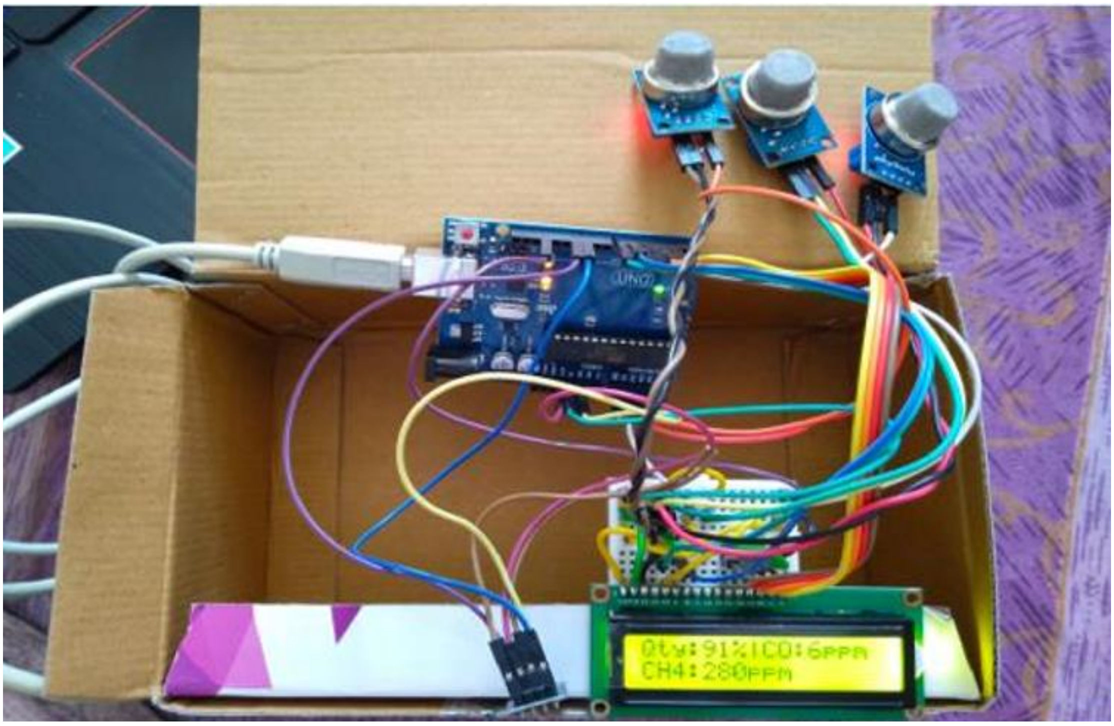
Directions

Go 1 unit straight
Turn right

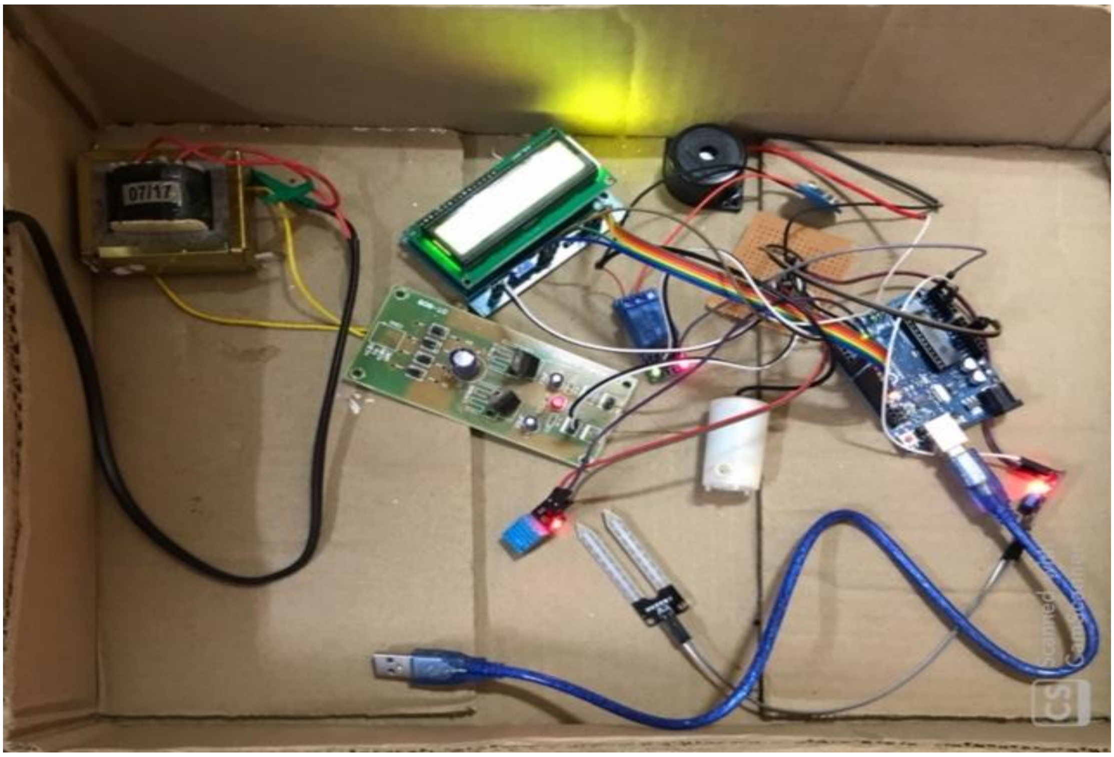




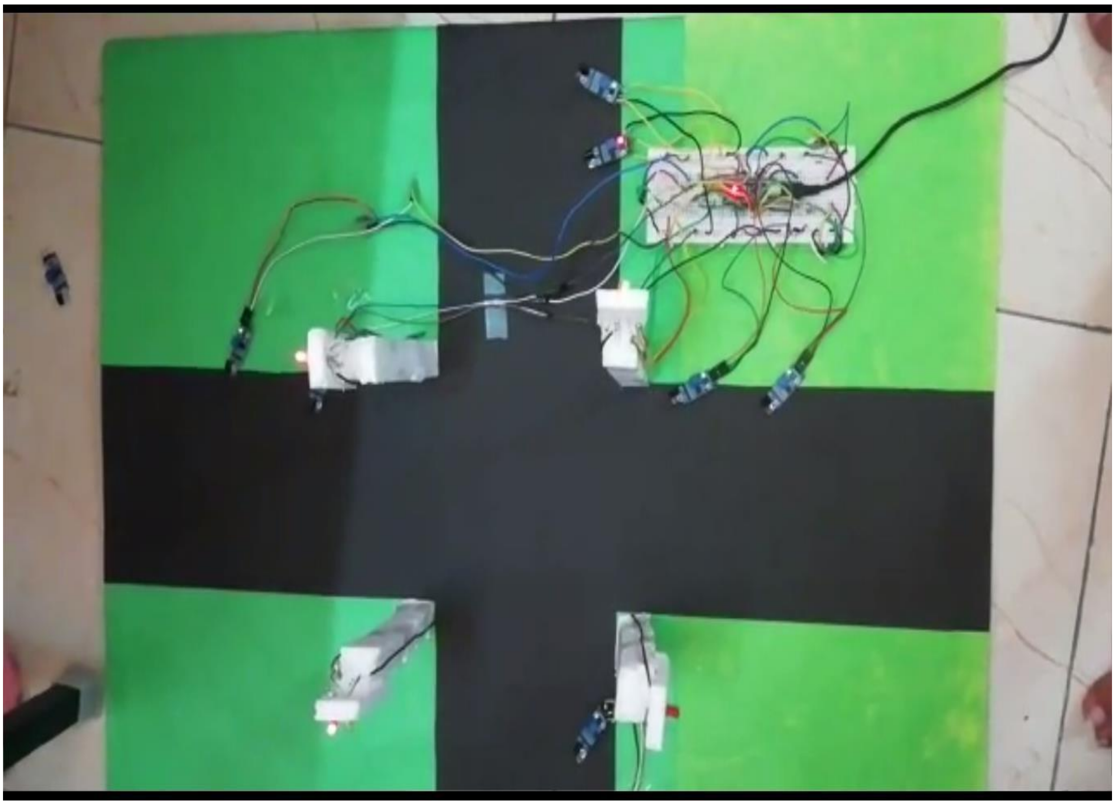
5. Air Pollution System



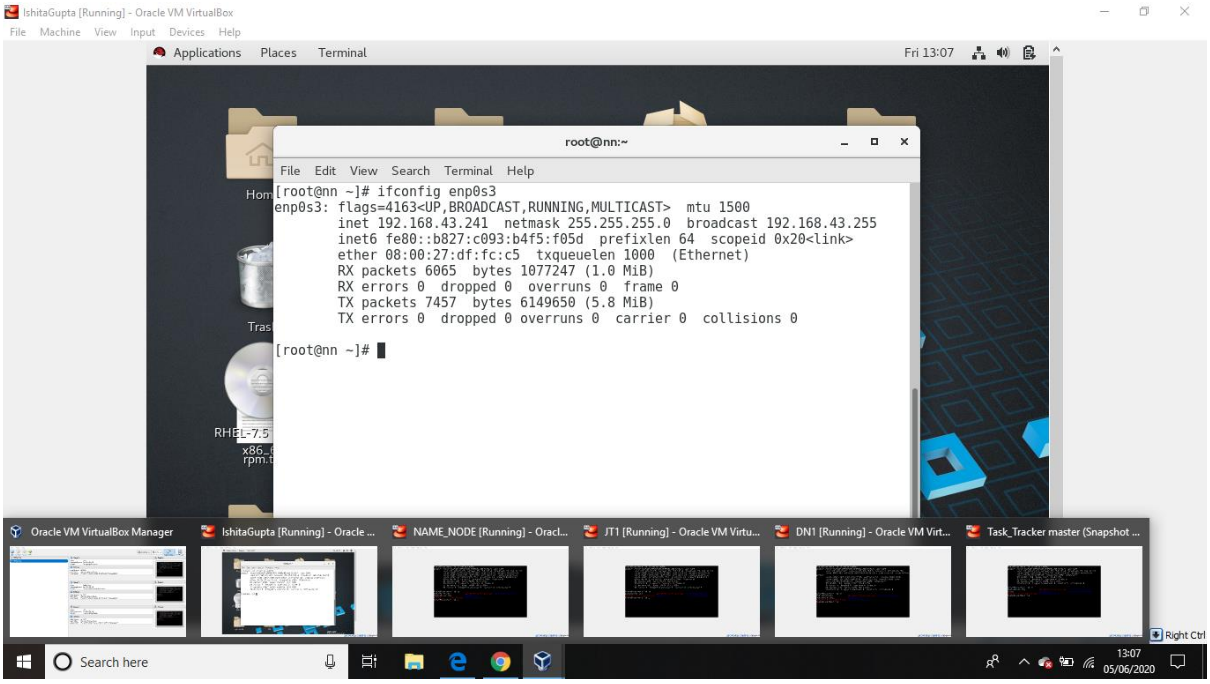
6. Smart Gardening System



7. Smart Traffic Lights



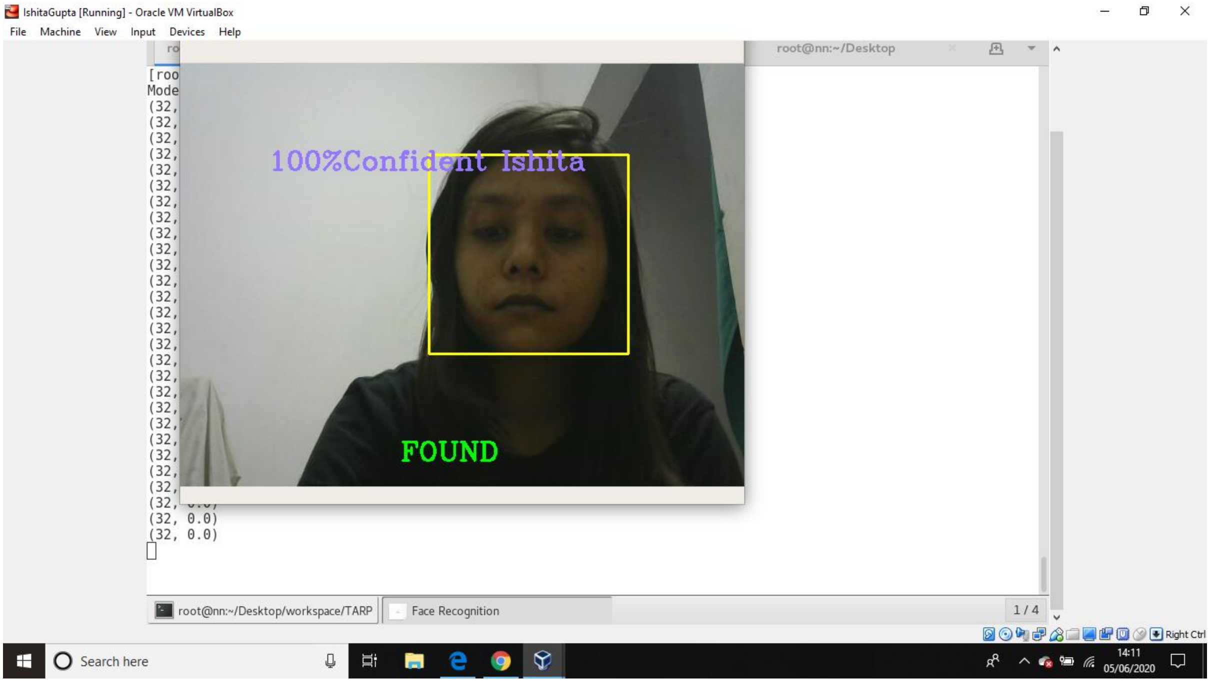
8. Smart CCTV System



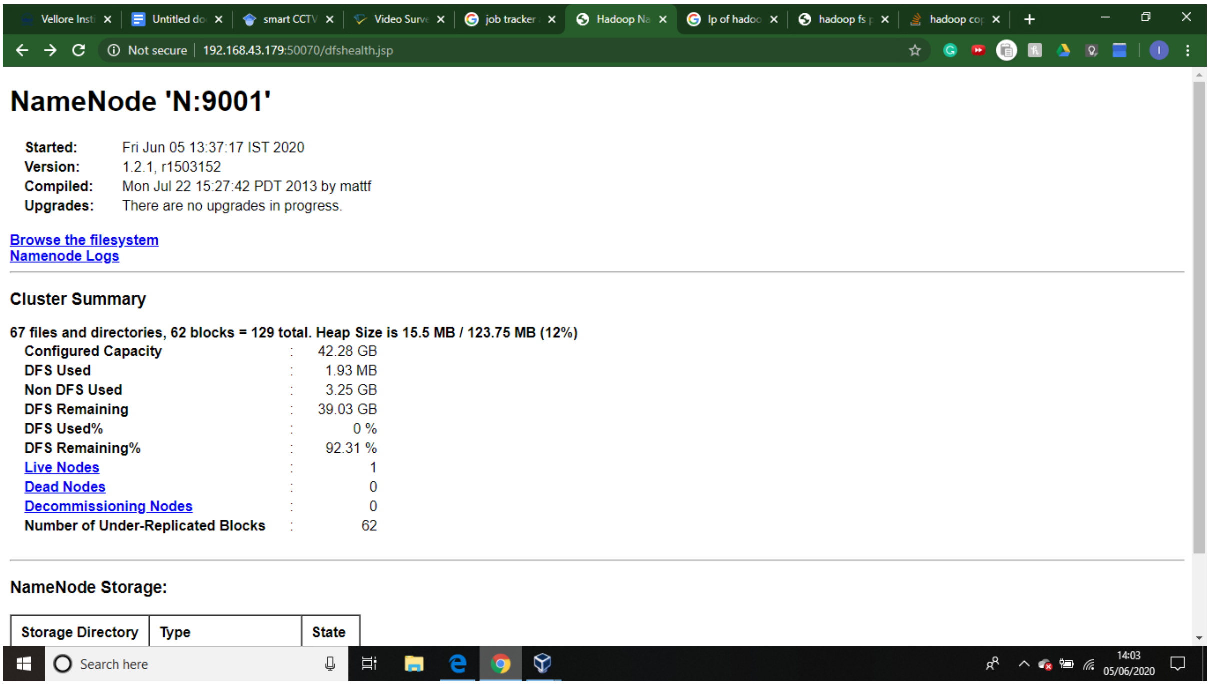
IshitaGupta [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places Terminal
Fri 13:07
root@nn:~
File Edit View Search Terminal Help
Oracle VM VirtualBox Manager
IshitaGupta [Running] - Oracle ...
NAME_NODE [Running] - Oracl...
JT1 [Running] - Oracle VM Virtu...
DN1 [Running] - Oracle VM Virt...
Task_Tracker master (Snapshot ...
Right Ctrl
Search here
13:07
05/06/2020

```
[root@nn ~]# ifconfig enp0s3
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.43.241  netmask 255.255.255.0  broadcast 192.168.43.255
        inet6 fe80::b827:c093:b4f5:f05d  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:df:fc:c5  txqueuelen 1000  (Ethernet)
        RX packets 6065  bytes 1077247 (1.0 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 7457  bytes 6149650 (5.8 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[root@nn ~]# 
```


IshitaGupta [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@nn:~/Desktop
100%Confident Ishita
FOUND
root@nn:~/Desktop/workspace/TARP
Face Recognition
1 / 4
Right Ctrl
Search here
14:11
05/06/2020

```
[roo
Mode
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32,
(32, 0.0)
(32, 0.0)
(32, 0.0)
```



NameNode 'N:9001'

Started:	Fri Jun 05 13:37:17 IST 2020
Version:	1.2.1, r1503152
Compiled:	Mon Jul 22 15:27:42 PDT 2013 by mattf
Upgrades:	There are no upgrades in progress.

Browse the filesystem
Namenode Logs

Cluster Summary

67 files and directories, 62 blocks = 129 total. Heap Size is 15.5 MB / 123.75 MB (12%)

Configured Capacity	:	42.28 GB
DFS Used	:	1.93 MB
Non DFS Used	:	3.25 GB
DFS Remaining	:	39.03 GB
DFS Used%	:	0 %
DFS Remaining%	:	92.31 %
Live Nodes	:	1
Dead Nodes	:	0
Decommissioning Nodes	:	0
Number of Under-Replicated Blocks	:	62

NameNode Storage:

Storage Directory	Type	State

7. COST ANALYSYS/ RESULT AND DISCUSSION

Individual Cost Components for each

1. Garbage Monitoring System

MRINALINI SINGH
17BCE0396

Sr. No.	Component	Price per quantity	Quantity	Total amount
1	Industrial Dustbin	Rs 3600	1	3600
2	Instant Adhesive	Rs 250	1	250
3	Ultrasonic sensor	Rs 80	1	80
4	Bread Board	Rs 120	1	120
5	Battery	Rs 25	1	25
6	LED	Rs 15	4	60
7	Jumper wires	Rs 2	20	40
8	LilOn NodeMCU V3	Rs 300	1	300
				4475 Rs

The total cost for this project implementation was around Rs 4475.If the project is to be implemented in real life, the cost estimate would be around Rs5000. If the city has n places where dustbins needs to be installed, the maximum cost will be around Rs. (n X 5000).
NOTE: The estimate is calculated by taking average of almost all favourable and harsh conditions into account.

2. Smart Streetlights

SANDHYA ANANTHAN
17BCI0099

Since this project has a base of streetlights (which are already existing), we just need to add the connections to an already existing model of streetlights.

Let us assume we are doing the analysis for 50 streetlights, each streetlight having 1 IR Sensor, connected to a joint Node MCU, which will cost around

Physical Cost:

(50*50(1 IR))+ (15* 300(Node MCU))+ 3000(wires)= 10,000 (for 50 street lights)

For extra equipment's like camera, and light sensor:

5*30(light censor)+ (10*1,000)=10,150 rupee

So, In total the cost for 50 street lights come up to 20,150(approximation).

This cost can be reduced while buying from wholesale or bulk.

3. Smart DoorLock System

DIVYA GAHLOT
17BCE0509

The Cost Structure can be idealized as follows. For the making of one setup of the Automatic Doorlock System, the following things were needed:

S.No	COMPONENT	COST
1.	Arduino Uno	330
2.	16*2 Liquid Crystal Display(LCD)	200
3.	4*4 Keyboard for Arduino	150
4.	Arduino IDE	-
5.	Servo Motor	20
6.	Buzzer for Bell	15
7.	Webcam for door	600
8.	Breadboard	60
9.	Jumper Wires	100
10.	Assembling Costs	100
11.	Packaging Costs	50
	TOTAL	1625

One entire setup for the Smart home security system costs Rs 1625 to make, package and deliver.
SCALING/EXPANDING
One Setup requires: Rs 1625
Considering supply of security system to a Colony with 500 households:
Total cost = 100 * 1625
= 1, 62, 500

4. Smart Car-Parking System

Sr no.	Component	Quantity	Price per piece	Total amount
1	Arduino board	1	330	330
2	Node MCU	1	180	180
3	Ir sensors	4	65	260
4	Rfid	1	200	200
5	Servo motor	1	20	20
6	Jumper wire (M to F)	15	2	30
7	Jumper Wire (M to M)	15	2	30
8	Bread Board	1	60	60
	TOTAL			Rs 1110

The cost estimation mentioned above is for what has been implemented, if the number of parking area increase, no of RFIDs needed will increase an so the cost.
So scaling up is we need to implement a parking space for 50 cars:
Then
Cost of RFID = 50* 200 = 10,000
Cost of ir sensors = 50* 65 = 3,250
So total cost = Rs 14,500

5. Air Pollution System

NISHTHA DUBEY
17BCE0082

Sr. No.	**Component**	**Prize per piece**	**Quantity**	**Total Amount**
1	Arduino Uno	Rs. 350	1	350
2	Wi-Fi module ESP8266	Rs. 380	1	380

3	16X2 LCD	Rs.220	1	220
4	1K ohm resistor	Rs.2	2	4
5	220ohm resistor	Rs.2	2	4
6	Male to female jumpers	Rs.4	10	40
7	Wires	Rs. 35	2 sets	70
8	Bread board	Rs.50	1	50
9	Buzzer	Rs.50	1	50
10	MQ135 Gas sensor	Rs.200	1	150
11	MQ2 Gas Sensor	Rs.150	1	150
12	MQ9 Gas Sensor	Rs.200	1	150
			TOTAL	**RS.1618**

The total cost for this project implementation was around Rs 1618. If the project is to be implemented in real life, the cost estimate would be around Rs 3000. If the city has n junctions, the maximum cost will be around Rs. (n X 3000).

NOTE: The estimate is calculated by taking average of almost all favourable and harsh conditions into account.

6. Smart Gardening System

Sr. No.	Component	Price per quantity	Quantity	Total amount
1	Arduino	Rs 350	1	350
2	Power supply	Rs 75	1	75
3	Step down Transformer	Rs 425	1	425
4	Motor	Rs 145	1	145
5	Relay	Rs 85	1	85
6	DHT11	Rs 98	1	98
7	Soil moisture sensor	Rs 70	1	70
8	16 x 2 LCD display	Rs 220	1	220

9	Buzzer	Rs 50	1	50
10	Jumper wires	Rs2	20	40
			Total	1,558

The total cost for this project implementation was around Rs 1558. If the project is to be implemented in real life, the cost estimate would be around Rs2000. If the city has n gardens where these models are needs to be installed, the maximum cost will be around Rs. (n X 2000).
NOTE: The estimate is calculated by taking average of almost all favourable and harsh conditions into account.

7. Smart Traffic Lights

Sr. No.	Component	Prize per piece	Quantity	Total Amount
1	Arduino Nano	Rs. 289	1	289
2	IR Sensor	Rs. 50	8	400
3	Red, yellow and green LEDs	Re.1	12	12
4	220 ohm resistor	Re.1	12	12
5	Male to female jumpers	Rs.4	24	96
6	Wires	Rs. 35	2 sets	70
7	Bread board	Rs.50	1	50

The total cost for this project implementation was around Rs 1000. If the project is to be implemented in real life, the cost estimate would be around Rs 3000. If the city has n junctions, the maximum cost will be around Rs. (n X 3000).
NOTE: The estimate is calculated by taking average of almost all favourable and harsh conditions into account.

8. Smart CCTV System

Cost for the production unit mostly consists of -

CCTV - A decent CCTV costs around 1000-1500 Rs. The CCTVs are installed everywhere mostly, so no such cost.

Data Center - Volume and RAM for processing.
The estimated cost is around $200 per square foot to **build a data center**, according to Forrester.

8. SUMMARY

In the end, the IoT technology stack for smart city applications today is relatively easy and cheap for many use cases such as smart waste management or smart parking. Urban environments typically have good wireless coverage for cases with moving parts, you have the cloud, there are several point solutions and products which are designed for smart city projects and in several cities across the globe there is low-power wide-area network connectivity available (LPWAN) which suffices for many applications. In fact, when drafting its list of IoT applications, IoT Analytics found that 59 percent (!) of all LPWAN projects today are part of smart city initiatives while LPWAN in general is only present in 10 percent of the IoT projects the company identified.

Choices regarding connectivity, data exchange, IoT platforms and so forth will evolve as we start moving to real smart cities. It's best if these future choices are already being taken into account for current smart city deployments.

9. REFERENCES

[1] *Dhaval Patel |Aditya Kulkarni | Hrushikesh Udar | Sachin Sharma "Smart Dustbins for Smart Cities"MET's Institute of Engineering, 2019*

[2] *"Smart City Waste Management System for Swacha Bharat Under Digital India"Prof. Sarang Patil, Avinash Autade, Rutuja Patil, Vishal Waman*

[3]*Smart Streetlight Using IR Sensors; Sindhu.A.M, Jerin George, Sumit Roy, Chandra J - IOSR Journal of Mobile Computing & Application (IOSR-JMCA)(2016)*

[4]*SMART STREET LIGHTING Mrs.M.PriyaDharsini, M.Keerthana, R.Keerthana, V.Malarvizhi, G.Meena (2016) - (IRJET)*

[5] *"IOT Based Smart Door System", K. Sri Viraja1, K. Bharath Kumar, C. Keerthi, G. Sandeep, April 2018*

[6] *"RFID, Password and OTP based Door Lock System using 8051 Microcontroller", A. Hemalatha, G. Gandhimathi*

[7]*“Smart Car Parking System”, Jayakshei Dadaji Bachhav1, Prof. Mechkul M. A.2 - (IRJET) [2017]*

[8]*IOT based Smart Parking Management System J. Cynthia, C. Bharathi Priya, P. A. Gopinath - International Journal of Recent Technology and Engineering (IJRTE) 2018*

[9]*Air Quality Monitoring System for City Pradeep D. Landge, R.R.Harne (IRJET) | Jan-2018*

[10]*Polluino: An Efficient Cloud-based Management of IoT Devices for Air Quality Monitoring Giovanni B. Fioccola,Raffaele Sommese, Imma Tufano, Roberto Canonico and Giorgio Ventre (IJIR) | 2017*

[11] *Smart Garden Monitoring System Using IOT Authors : T.Thamaraimanalan, S.P.Vivekk, G.Satheeshkumar, P.Saravanan*

[12] *“Cloud based IoT for Smart Garden Watering System using Arduino Uno”,* Authors : *Alauddin Al-Omary*, *Haider M. Alsabbagh*, *Hussain Al-Rizzo*

[13] *Smart Traffic Light Scheduling in Smart City Using Image and Video Processing MeisamRazvai, Mehdi Hamidkhani, Rasool Sadeghi | 2019*

[14]*IoT based smart traffic signal monitoring system using vehicles counts Senthil Kumar Janahan,Veeramanickam Murugappan, Arun Sahayashas, Kumar Narayanan | 2018*

***