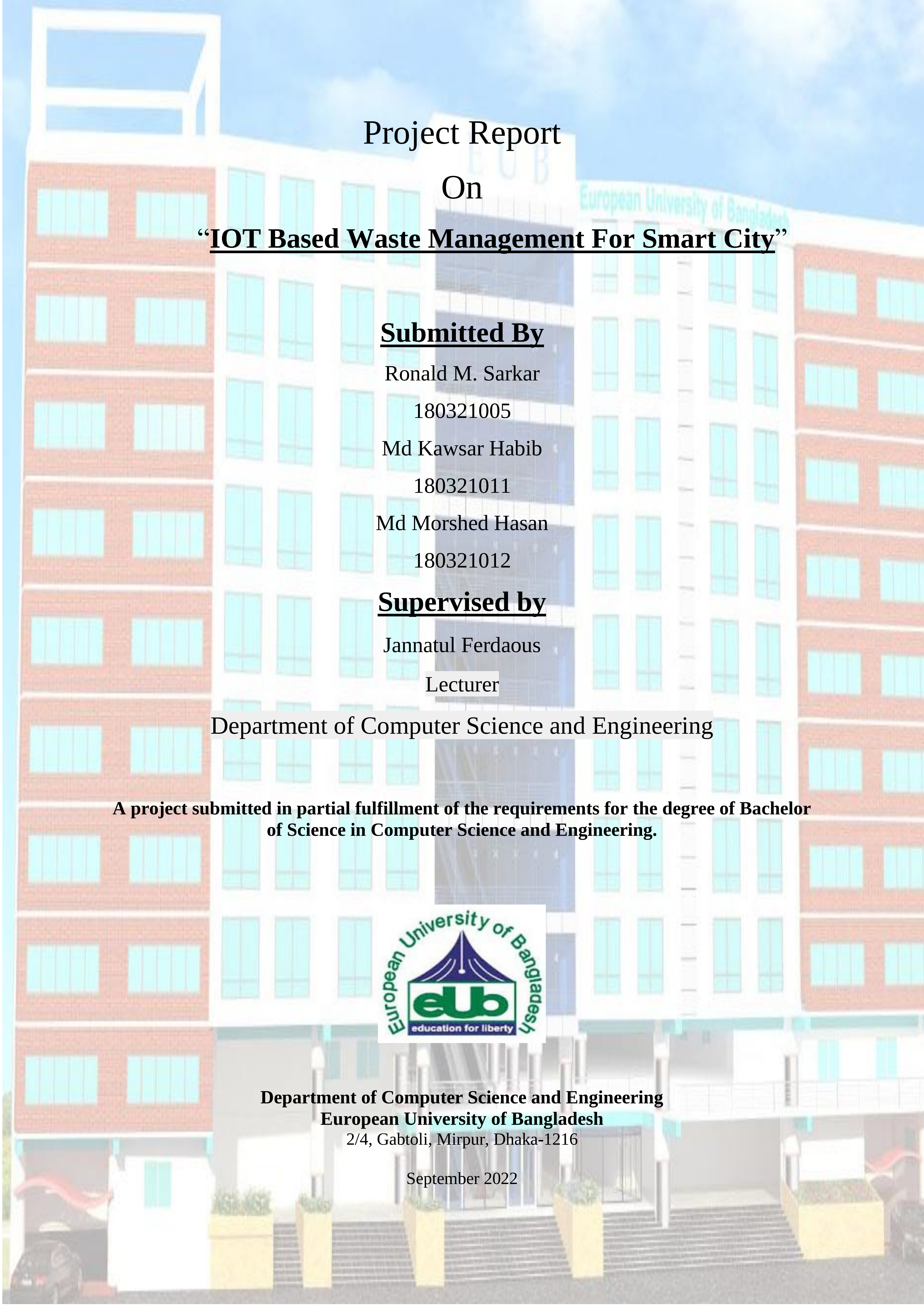


Project Report

On

"IOT Based Waste Management For Smart City"

Submitted By

Ronald M. Sarkar

180321005

Md Kawsar Habib

180321011

Md Morshed Hasan

180321012

Supervised by

Jannatul Ferdaous

Lecturer

Department of Computer Science and Engineering

A project submitted in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering.

Department of Computer Science and Engineering
European University of Bangladesh
2/4, Gabtoli, Mirpur, Dhaka-1216

September 2022

CANDIDATES' DECLARATION

This is to certify that the work presented in this project, titled, "IOT Based Waste Management For Smart City", is the outcome of the investigation and work carried out by us under the supervision of Jannatul Ferdaous.

It is also declared that this project nor any part thereof has been submitted anywhere else for the award of any degree, diploma or other qualifications.

Signature

Ronald M. Sarkar
180321005

Signature

Md Kawsar Habib
180321011

Signature

Md Morshed Hasan
180321012

CERTIFICATE OF APPROVAL

This thesis/project titled, “IOT Based Waste Management For Smart City”, submitted by the group as mentioned in the candidates’ declaration page has been accepted as satisfactory in partial fulfillment of the requirements for the degree B.Sc. in Computer Science and Engineering in September 2022.

Signature of Supervisor:

Jannatul Ferdaous

Lecturer

Department of Computer Science and Engineering
European University of Bangladesh, Dhaka, Bangladesh.

Signature of Chairperson:

Md. Obaidur Rahman
Associate Professor and Chairperson
Department of Computer Science and Engineering
European University of Bangladesh, Dhaka, Bangladesh

ACKNOWLEDGEMENT

First of all, we would like to show our gratitude to Almighty Allah for His kind blessings that allowed us to move forward towards the final project of our Undergrad Degree so successfully.

We will whole heartedly mention our parents for their constant love and support and providing us with such an opportunity.

We would like to express our outmost gratefulness to our **Supervisor, Jannatul Ferdaous**, Lecturer, Department of CSE, European University of Bangladesh, Dhaka. Our supervisor's in-depth expertise and keen interest in the field of ‘Internet Of Things’ has helped us carry out every task efficiently. This thesis could not have been finished without her never-ending patience, academic direction, constant encouragement, energetic supervision, constructive criticism, helpful suggestions, reviewing numerous subpar versions and correcting them at all levels.

Finally to acknowledge our entire department of CSE, all the faculty members including our honorable Chairman, **Md. Obaidur Rahman**, for the support and paitence towards us even in very tough and trying times, it has been a privilege.

Table of Contents

List of Figures

ABSTRACT

IOT, "internet of things" is a very popular term nowadays. There are many IoT-based studies, however studies based on IoT-based garbage management systems are not very common. An inadequate and ineffective system of rubbish management severely harms the ecosystem. Additionally, it makes the air poisonous. This issue has spread throughout the world, but is particularly prevalent in Bangladesh. The capital of Bangladesh, Dhaka, does not have an effective or well-organized waste management system. Most of the city's highways in Dhaka are lined with trash. People's mental health is impacted by the disturbing rubbish stench, and breathing polluted air causes various diseases. The lack of trash cans, strewn rubbish, and overflowing of trash cans endanger the environment and make city living exceedingly unhealthy. With this project, we provided an easy solution using an Internet of Things-based waste management system. The goal of this research project is to use IOT to offer a straightforward solution to this issue. In our project, a sensor-based smart waste system is proposed. It consists of ultrasonic sensors, a Wi-Fi module, a GPS system and an IOT platform that is a very functioning and fast mobile application. A new era can be ushered in by creating a clean nation and pollution-free environment with an effective and well-organized waste management system. It is hoped that the suggested IOT based waste management system will be successful in lowering the country's pollution level and keeping the environment clean and speed up recycling.

Chapter 1

Introduction

The amount of waste produced daily by businesses and households is rising at an alarming rate, and the main cause of this is the rising consumption of packaged goods, textiles, paper, food, plastics, metals, and glass, among other materials. As a result, managing this waste has become an important aspect of daily life.

The repercussions of trash on our health and the environment are terrible, and there are many of them. Trash serves as a breeding habitat for bacteria, insects, and flies—the same flies that fly around edibles and lay their eggs. Thus, they raise the risk of food poisoning, typhoid, gastroenteritis, salmonella, and other illnesses caused by insects, such as malaria and dengue fever.

The basic idea of this project is to design a smart waste detection system which would automatically notify the officials about the current status of various garbage bins in the city, would have real-time monitoring capabilities. Embedded devices that are connected to Internet and these devices are capable of reading the garbage levels of the bins in real time using ultrasonic sensors and simple microcontrollers. Not only will it notify the officials but is also capable of showing the location it is in using the help of a GPS module.

1.1 Motivation

Lack of waste management or its absence results in serious environmental problems, health issues, and economic losses. There have been numerous studies conducted on IoT-based trash management, but neither the quantity nor the quality of those studies is enough. In most of the developed countries there are many efficient techniques which are used for the proper management of this waste, but in some countries especially the developing ones like Bangladesh the careless attitude of the organization towards maintaining clean surroundings is a huge problem.

It comes down to the overflowing of dustbins. Due to an increase in garbage, neighborhood trash cans are overflowing and the area is littered, which not only makes the streets smell bad but also has a severe influence on the environment and public health. No proper system for the garbage collectors to knowing when to clean up and where to go first.

1.2 Objectives

The Implementation of IOT. The functions of IOT are connecting things, sensing, processing, gathering data, storing. These days IOT applications are all around us. IOT has a wide range of uses, including smart cities, smart homes, environmental monitoring, healthcare, and item tracking. Despite the fact that there are numerous applications related to making life easier based on IoT there isn’t much for waste management.

That’s where we come in. The main goal of this project is to create a smart waste detection system that can monitor waste in real-time and alert authorities to any changes in the location or the level of garbage of the city's different trash cans. With the use of ultrasonic sensors and microcontrollers, embedded devices, with Internet access will read the garbage levels of the bins in real time. Along with alerting the authorities, it also has a GPS module that can be used to display its location.

1.3 Scope of Work

As we have seen number of times the dustbins are getting overflown and concern person don’t get the information within a time and due to which unsanitary condition formed in the surroundings, at the same time bad smell spread out due to waste, bad look of the city which paves the way for air pollution and to some harmful diseases around the locality which is easily spreadable.

1.3.1 Flaws of Existing Systems

The following are the primary issues with the current waste collecting procedures and management methods would be

- Not knowing when the garbage is filling up.
- No efficient network between the collectors and the officials.
- Not knowing where to go first and no easy route.
- Trash collectors needs to check every bin whether its full or not.
- Makes collecting garbage very time consuming.
- Leading to improperly work done and abandoning some spots.
- Eventually polluting the environment.

1.3.2 Our Proposed System

A Smart bin but modified using a microcontroller with built-in WIFI module named ESP-32, similar to Arduino, interfaced with an Ultrasonic sonar sensor modeled HC-SR04 and a GPS module, U-blox Neo-6M. The electric components' job is simple. It detects the level of trash in real time and sends it to the server of an IOT platform.

When the user dumps the trash into the dustbin, the Ultrasonic sonar sensor that is attached inside the bin will measure the level of the trash using simple arithmetic formula. The limit of the level of garbage is preset. Once the limit is hit, a notification will immediately be sent to an IOT platform (Blynk) with its own servers.

Considering there multiple such bins in an area, the GPS module will also show the respective location on the map of Blynk. The location will allow for the garbage collectors to know where to go first and collect trash on time and more efficiently.

Advantages of proposed system over the existing:

- Low implementation cost
- Simple module
- Easy functionality
- GPS tracking
- Real time information on the fill level of the dustbin.
- Deployment of dustbin based on the actual needs.
- Cost Reduction and resource optimization.
- Improves Environment quality (Fewer stench & cleaner cities).
- Intelligent management of the services in the city.
- Effective usage of dustbins.

Chapter 2

Literature Review

The concept of smart bins is not new, it has been in talks for much of the time. In fact, it is already being implemented is some parts of the world. Various authors have published their papers regarding smart bins.

In [1], the ZigBee, GSM (Global System for Mobile Communication) and ARM7 is used to form the Integrated system to monitor the waste bins remotely. The sensors are placed in the common garbage bins placed at the public places. When the garbage reaches the level of the sensor, then that indication will be given to ARM 7 Controller. The controller will give indication to the driver of garbage collection truck as to which garbage bin is completely filled and needs urgent attention. ARM 7 will give indication by sending SMS using GSM technology.

In [2], they came to a point It is important to understand the societal concerns over the increased rate of resource consumption and waste production and therefore the policy makers have encouraged recycling and reuse strategies to reduce the demand for raw materials and to decrease the quantity of waste going to landfill.

In [3], it is being proposed in this paper that introduction of an integrated system combined with an integrated system of Radio Frequency Identification, Global Position System, General Packet Radio Service, Geographic Information System and web camera will solve the problem of solid waste They also analyzed the actual performance of the system.

In [4], this paper objective of the study was to determine the characterization of the waste and the current system of management activities. The paper highlights an overview of the current municipal solid waste management (MSWM) system of Thoubal Municipality and it concludes with a few suggestions, which may be beneficial to the authorities to work towards further improvement of the current management systems.

In [5], the proposed system describes that the level of garbage in the dustbins is detected with the help of Sensor systems, and communicated to the authorized control room through GSM system. Microcontroller is used to interface the sensor system with GSM system. A GUI is also developed to monitor the desired information related to the garbage for different selected locations. This will help to manage the garbage collection efficiently.

In [6], it describes the application of our model of “Smart Bin” in managing the waste collection system of an entire city. The network of sensors enabled smart bins connected through the cellular network generates a large amount of data, which is further analyzed and visualized at real time to gain insights about the status of waste around the city. This paper also aims at encouraging further research in the topic of waste management.

Chapter 3

Implementations

3.1 System Requirements

Electronics:

- HC-SR04; Ultrasonic Sonar Sensor.
- ESP32; Microcontroller with integrated WIFI.
- U-blox Neo-6M; GPS Module.

Software:

- Arduino IDE; To compile and upload the sketch into the microcontroller
- Blynk; A common user friendly IOT platform for mobile devices.

Code:

- C++

3.2 Details of the electronics

3.2.1 ESP WROOM 32

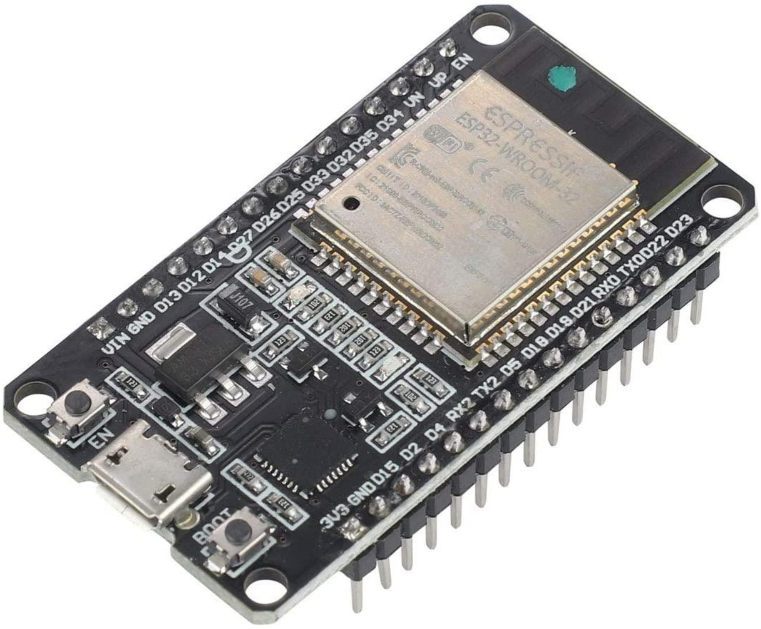

ESPRESSIF
ESP32-WROOM-32
VIN GND D13 D12 D14 D27 D26 D25 D33 D32 D35 D34 VN VP EN
3V3 GND D15 D2 D4 RX2 TX2 D5 D18 D19 D21 RX0 TX0 D22 D23
EN
BOOT

Figure 1. ESP WROOM 32 Microcontroller.

<u>Specification:</u>

- Processors:
  - CPU: Xtensa dual-core (or single-core) 32-bit LX6 microprocessor, operating at 160 or 240 MHz and performing at up to 600 DMIPS
  - Ultra-low power (ULP) co-processor
- Memory: 320 KiB RAM, 448 KiB ROM
- Wireless connectivity:
  - Wi-Fi: 802.11 b/g/n
  - Bluetooth: v4.2 BR/EDR and BLE (shares the radio with Wi-Fi)

Some features of the ESP32:

Robust design; ESP32 is capable of functioning reliably in industrial environments, with an operating temperature ranging from –40°C to +125°C. Powered by advanced calibration circuitries, ESP32 can dynamically remove external circuit imperfections and adapt to changes in external conditions.

Ultra-Low Power Consumption**;** Engineered for mobile devices, wearable electronics and IoT applications, ESP32 achieves ultra-low power consumption with a combination of several types of proprietary software. ESP32 also includes state-of-the-art features, such as fine-grained clock gating, various power modes and dynamic power scaling.

High Level of Integration; ESP32 is highly-integrated with in-built antenna switches, RF balun, power amplifier, low-noise receive amplifier, filters, and power management modules. ESP32 adds priceless functionality and versatility to your applications with minimal Printed Circuit Board (PCB) requirements.

Hybrid Wi-Fi & Bluetooth Chip**;** ESP32 can perform as a complete standalone system or as a slave device to a host MCU, reducing communication stack overhead on the main application processor. ESP32 can interface with other systems to provide Wi-Fi and Bluetooth functionality through its SPI / SDIO or I2C / UART interfaces.

Power Supply:

The ESP32 Board operates between 2.2V to 3.6V. But we supply 5V from Micro-USB port. For 3.3V there is already an LDO voltage regulator to keep the voltage steady at 3.3V. ESP32 can be powered using Micro USB port and VIN pin (External Supply Pin).

3.2.2 HC-SR04 Ultrasonic Sonar Sensor

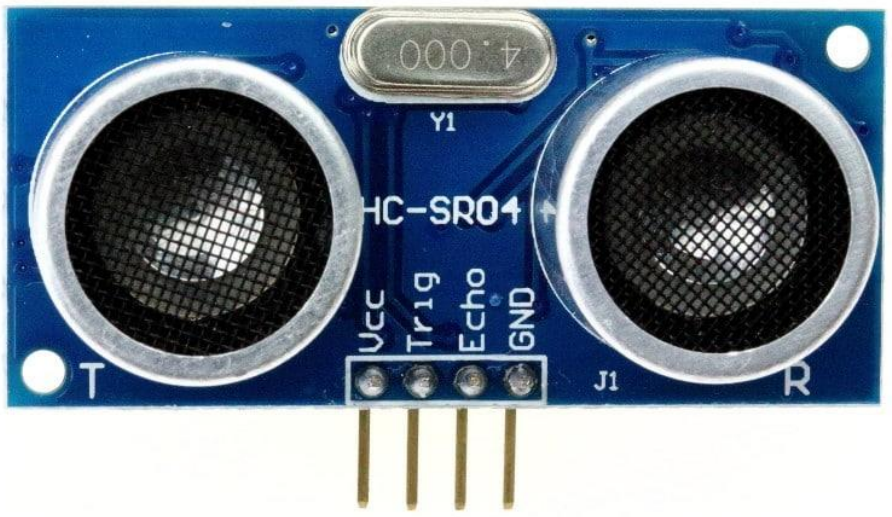


Figure 2. HC-SR04 Ultrasonic Sonar Sensor.

<u>Technical Specification:</u>

Operating Voltage	DC 5V
Operating Current	15mA
Operating Frequency	40KHz
Max Range	4m
Min Range	2cm
Ranging Accuracy	3mm
Measuring Angle	15 degrees
Trigger Input Signal	10µS TTL pulse
Dimension	45 x 20 x 15mm

How the HC-SR04 works:

An HC-SR04 ultrasonic distance sensor actually consists of two ultrasonic transducers. One acts as a transmitter that converts the electrical signal into 40 KHz ultrasonic sound pulses. The other acts as a receiver and listens for the transmitted pulses.

When the receiver receives these pulses, it produces an output pulse whose width is proportional to the distance of the object in front.

This sensor provides excellent non-contact range detection between 2 cm to 400 cm (~13 feet) with an accuracy of 3 mm.

A brief explanation of Ultrasonic Sound:

Ultrasound is a high-pitched sound wave whose frequency exceeds the audible range of human hearing.

Humans can hear sound waves that vibrate in the range of about 20 times a second (a deep rumbling noise) to 20,000 times a second (a high-pitched whistle). However, ultrasound has a frequency of more than 20,000 Hz and is therefore inaudible to humans.

How Does HC-SR04 Ultrasonic Distance Sensor Work?

It all starts when the trigger pin is set HIGH for 10µs. In response, the sensor transmits an ultrasonic burst of eight pulses at 40 kHz. This 8-pulse pattern is specially designed so that the receiver can distinguish the transmitted pulses from ambient ultrasonic noise.

These eight ultrasonic pulses travel through the air away from the transmitter. Meanwhile the echo pin goes HIGH to initiate the echo-back signal.

If those pulses are not reflected back, the echo signal times out and goes low after 38ms (38 milliseconds). Thus, a pulse of 38ms indicates no obstruction within the range of the sensor.

Calculating the Distance:

The width of the received pulse is used to calculate the distance from the reflected object. This can be worked out using the simple distance-speed-time equation we learned in high school.

Let us take an example to make it clearer. Suppose we have an object in front of the sensor at an unknown distance and we receive a pulse of 500µs width on the echo pin. Now let's calculate how far the object is from the sensor. For this we will use the below equation.

$$\text{Distance} = \text{Speed x Time.}$$

Here we have the value of time i.e., 500 µs and we know the speed. Of course, it's the speed of sound! It is 340 m/s. To calculate the distance, we need to convert the speed of sound into cm/µs. It is 0.034 cm/µs. With that information we can now calculate the distance!

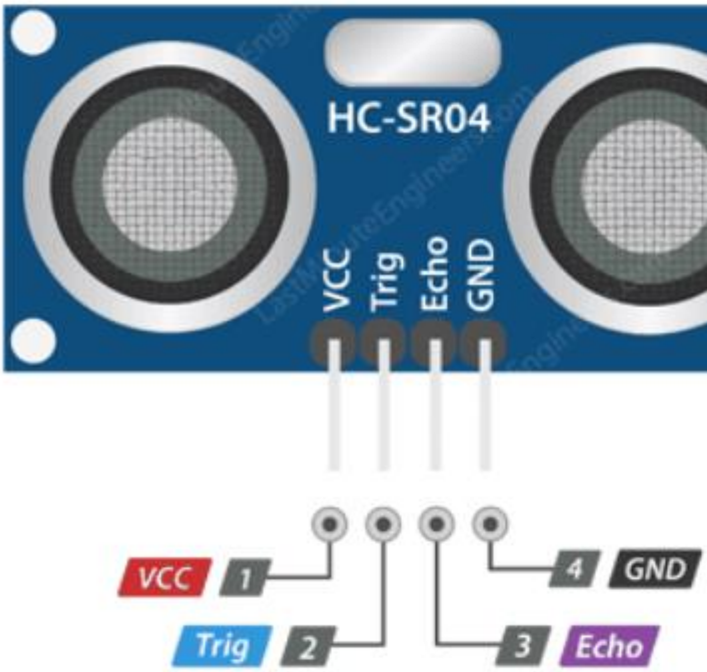


Figure 3. HC-SR04 Pinout

VCC supplies power to the HC-SR04 ultrasonic sensor. You can connect it to the 5V output from your Arduino.

Trig (Trigger) pin is used to trigger ultrasonic sound pulses. By setting this pin too HIGH for 10µs, the sensor initiates an ultrasonic burst.

Echo pin goes high when the ultrasonic burst is transmitted and remains high until the sensor receives an echo, after which it goes low. By measuring the time, the Echo pin stays high, the distance can be calculated.

GND is the ground pin. Connect it to the ground of the Arduino.

3.2.3 U-Blox Neo-6M GPS Module

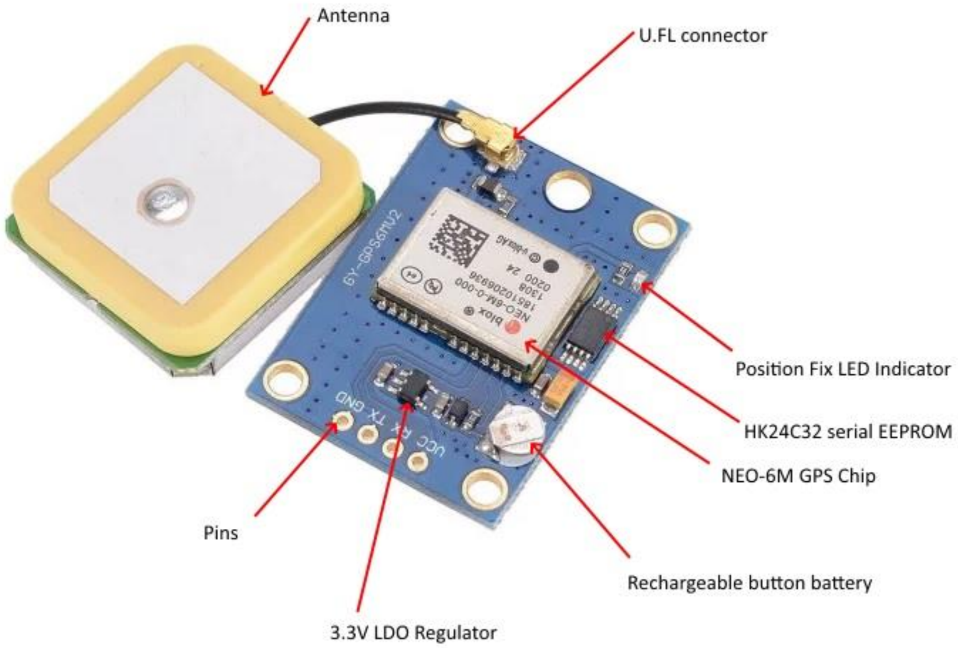


Figure 4. U-Blox Neo-6M GPS Module.

Technical Specification:

Receiver Type	50 channels, GPS L1(1575.42Mhz)
Horizontal Position Accuracy	2.5m
Navigation Update Rate	1HZ (5Hz maximum)
Capture Time	Cool start: 27sHot start: 1s
Navigation Sensitivity	-161dBm
Communication Protocol	NMEA, UBX Binary, RTCM
Serial Baud Rate	4800-230400 (default 9600)
Operating Temperature	-40°C ~ 85°C
Operating Voltage	2.7V ~ 3.6V
Operating Current	45mA
TXD/RXD Impedance	510Ω

About the Neo-6M GPS MODULE

It consists of two parts:

- The Neo-6m Chip
- The Antenna

Neo-6m Chip

At the heart of the module is a GPS chip from U-blox – NEO-6M. It can track up to 22 satellites over 50 channels and achieve the industry’s highest level of tracking sensitivity i.e., -161 dB, while consuming only 45 mA current.

Unlike other GPS modules, it can perform 5 location updates in a second with 2.5m horizontal position accuracy. The U-blox 6 positioning engine also has a Time-To-First-Fix (TTFF) of less than 1 second.

One of the best features offered by the chip is Power Save Mode (PSM). This allows a reduction in system power consumption by selectively switching certain parts of the receiver on and off. This dramatically reduces the power consumption of the module to just 11mA making it suitable for power sensitive applications such as GPS wristwatches.

The required data pins of the NEO-6M GPS chip are broken out to 0.1″ pitch headers. It contains the pins needed for communication with the microcontroller over the UART. The module supports baud rates from 4800bps to 230400bps with a default baud of 9600.

Position Fix LED Indicator

There is an LED on the NEO-6M GPS module that indicates the status of the ‘Position Fix’. It will blink at different rates depending on which state it is in:

- No blinking – it is searching for satellites.
- Blink every 1s – Position Fix is found (the module can see enough satellites).

3.3V LDO Regulator

The operating voltage of the NEO-6M chip ranges from 2.7 to 3.6V. But the good news is, this module comes with MICREL’s MIC5205 Ultra-Low Dropout 3V3 regulator. The logic pins are also 5-volt tolerant, so we can easily connect it to Arduino or any 5V logic microcontroller without using a logic level converter.

Antenna

The module comes with -161 dBm sensitivity patch antenna for receiving radio signals from GPS satellites. This antenna can be snap-fitted into the small U.FL connector located on the module.

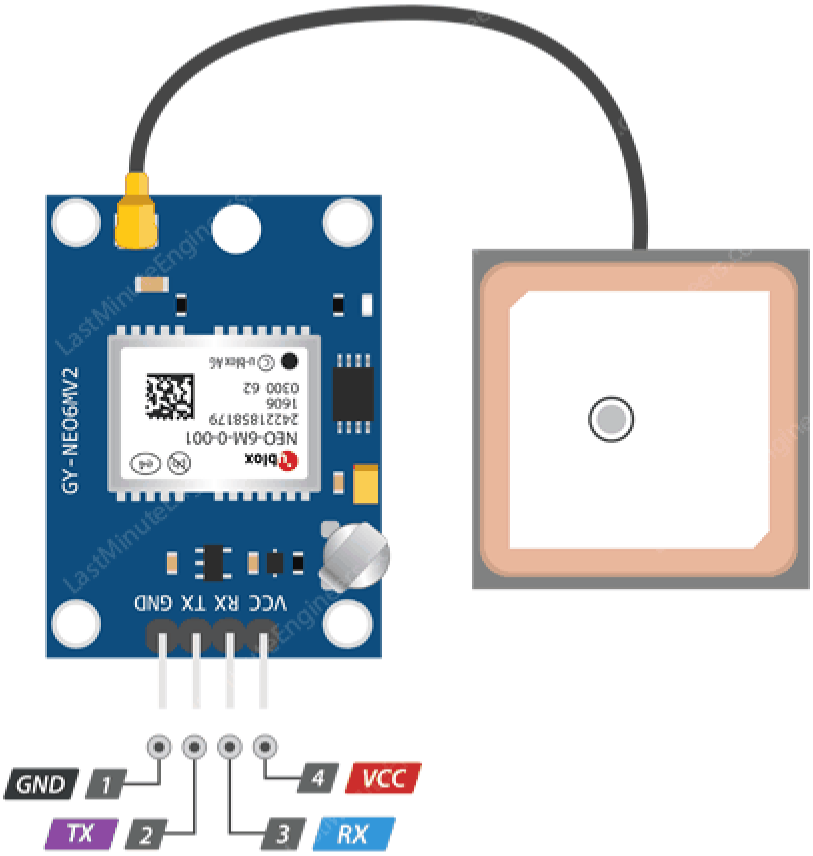


Figure 5. NEO-6M GPS Module Pinout

The NEO-6M GPS module has a total of 4 pins that connect it to the outside world. The connections are as follows:

GND is the ground pin and needs to be connected to the GND pin on the Arduino.

TxD (Transmitter) pin is used for serial communication.

RxD (Receiver) pin is used for serial communication.

VCC supplies power to the module. You can connect it directly to the 5V pin on the Arduino.

3.3 Algorithm

1. Start.
2. Initiate instructions to track location.
3. Initiate instructions for the Ultrasonic sensor.
4. Send and receive sonar waves from the Ultrasonic sensor.
5. If, calculated value>Threshold Value, send email and notify user
    a. Else, continue.
6. If location of bin is changed, update coordinates.
    a. Else, keep previous.
7. Send real time coordinates and garbage level.

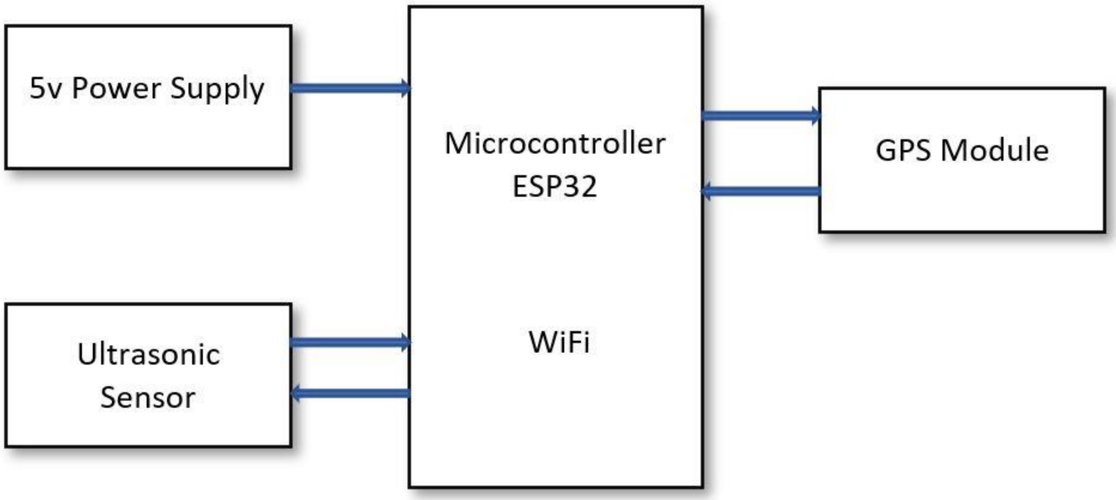


Figure 6. Smart Bin Interior

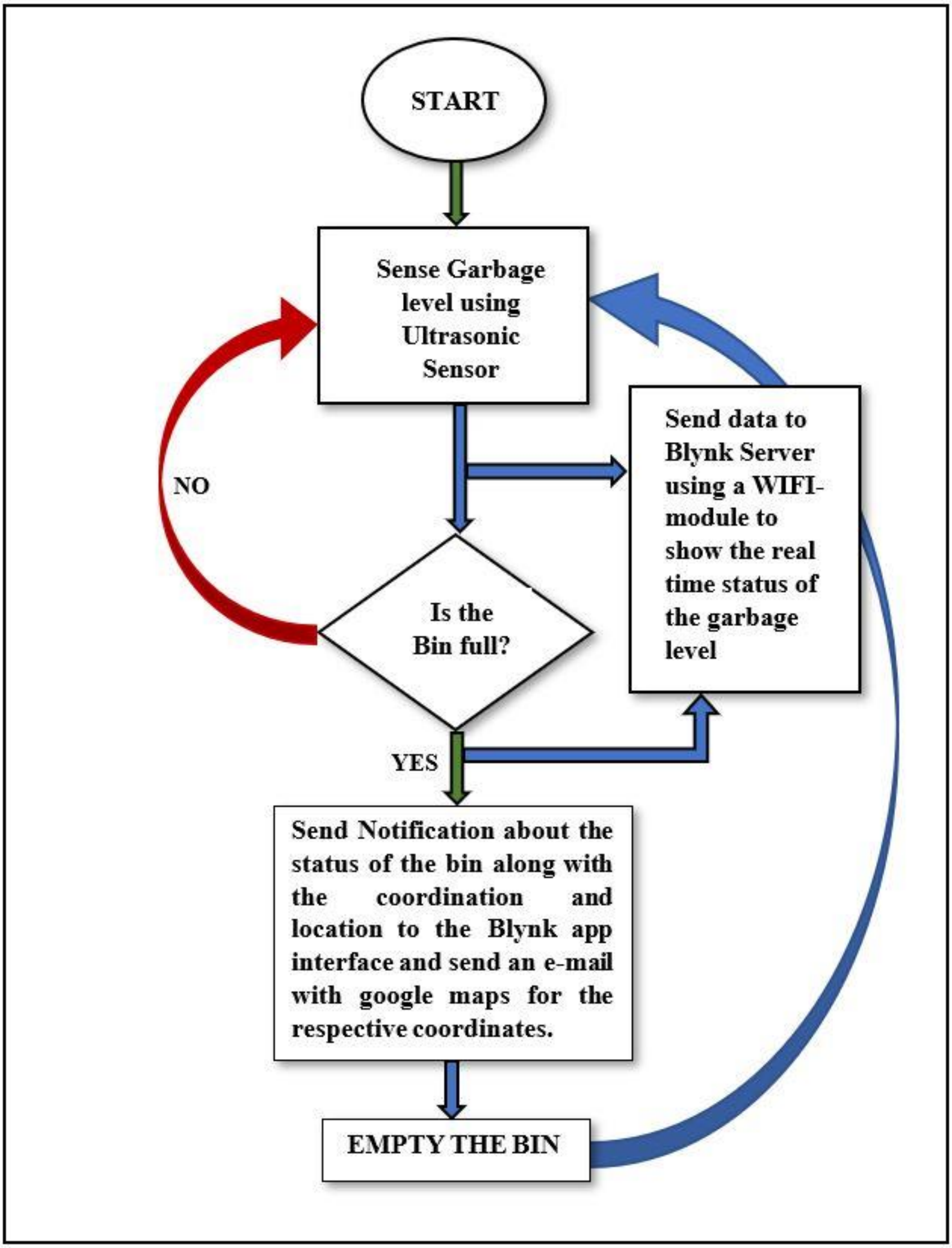

START
Sense Garbage level using Ultrasonic Sensor
NO
Send data to Blynk Server using a WIFI-module to show the real time status of the garbage level
Is the Bin full?
YES
Send Notification about the status of the bin along with the coordination and location to the Blynk app interface and send an e-mail with google maps for the respective coordinates.
EMPTY THE BIN

Figure 7. Flow Chart of The Project

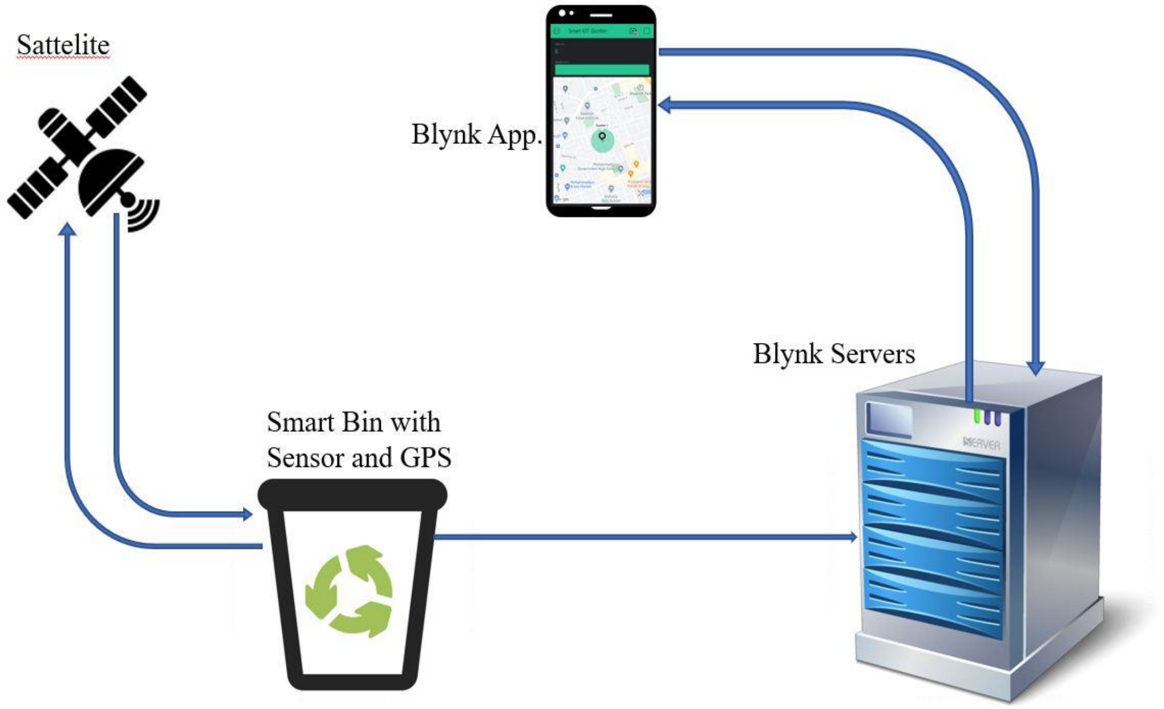


Figure 8. Diagram of the System operation.

3.5 System Operation:

The HC-SR04 Ultrasonic Sensor from the bin constantly tracks the level of the garbage disposed in it. It instantly updates the status to the Blynk Servers for the Mobile app to be viewed by the users. Furthermore, it also shows the location of the bin which is done by the Neo6M GPS module that updates its location connecting itself to several satellites.

If the location of the smart bin is changed for whatever reason, it is immediately updated by taking the latitudes and longitudes for the coordinates and sending the data to the servers and then to the Mobile app. For a means of energy saving, the location is only updated if the bin is somehow moved for some reason. Finally, ESP32, the microcontroller that handles everything is also integrated with WIFI for ease of use.

3.6 Setting Up the Arduino IDE

About Arduino IDE and why

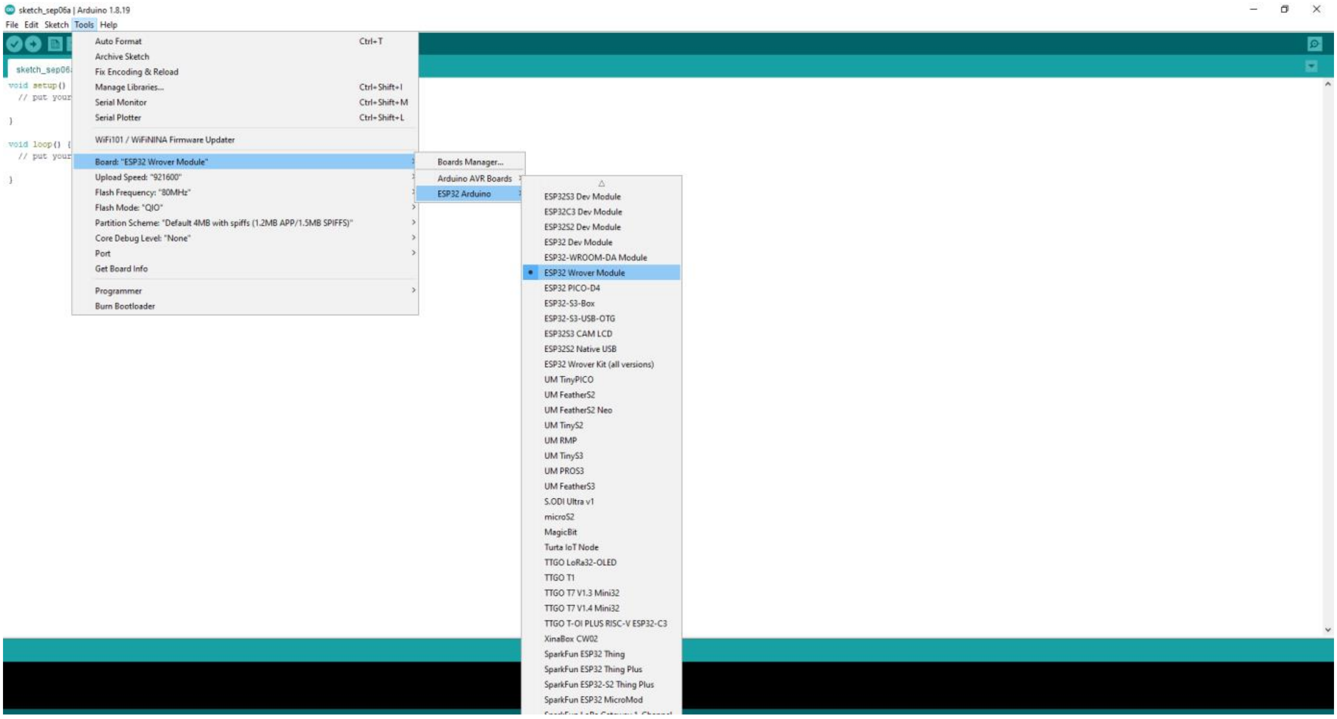
One will require the proper software and programming to make any microcontroller come to life and allow it to interact with various sensors and shields. It is the open-source Arduino Software that makes it easy to write code and upload it to the board.

Arduino IDE makes it easy for you to write code and upload it on your Arduino board. This program is cross-platform which means it is able to run on Windows, Mac OS X, and Linux compared to other microcontroller systems which can only run Windows.

This software can be used with any Arduino-like boards like the very one our project is going to use. The environment is written in Java and based on processing and other open-source software.

This program uses a simplified version of C++ with syntax highlighting and other features which makes it easier to learn to program which is perfect for beginners to learn programming and coding!

It is preferred to always use the latest version available from the official website.
During the time of our work, we took advantage of the IDE version 1.8.19.



```
void setup() {
  // put your

}

void loop() {
  // put your

}
```

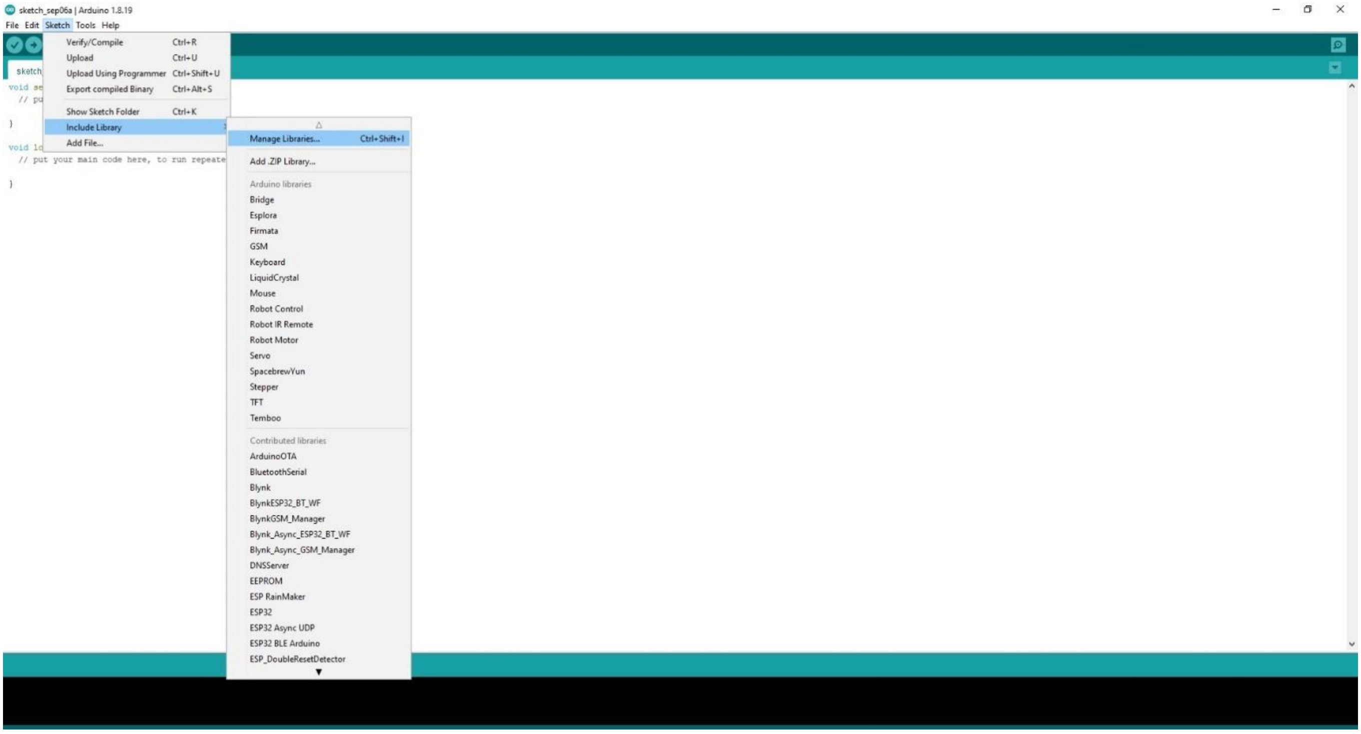
Board: "ESP32 Wrover Module"
Upload Speed: "921600"
Flash Frequency: "80MHz"
Flash Mode: "QIO"
Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)"
Core Debug Level: "None"

Figure 9. Installing Board Manager

Assuming That the IDE is already installed. Then, in order for it to recognize the microcontroller, The Board Manager for that particular device needs to be set up. It is fairly very simple. There are two ways.

The first would be to copy an installation link from a any website that would provide it or from the official website and paste it on a section at the ‘preferences’ window that says ‘Additional Board Manager URLs’.

The second method is even simpler, just like shown if the image above (Figure 6). Under the tools tab there is an option for a sub tab that says Boards and then Board Manager. Inside the Board Manager window, the ESP32 can be searched in the search bar and installed.



```
void se
  // pu

}

void lo
  // put your main code here, to run repeate

}
```

Figure 10. Installing Necessary Libraries

All The necessary Libraries must be installed before the writing the sketch.

In order for the every electronic as well as the software to be embedded the libraries are installed. Just like the previous step (Installing Board Manager), Installing Libraries is also fairly simple.

Under Sketch tab>

Manage Library>

Then in the search bar just the name of the electronics needs to be typed and the respective libraries installed. Then Finally the Blynk libraries for all the functions to work properly.

3.7 Setting Up Blynk

About Blynk:

Blynk is an IoT platform for iOS or Android smartphones that is used to control Arduino, Raspberry Pi and many similar microcontrollers via the Internet. This application is used to create a graphical interface or human machine interface (HMI) by compiling and providing the appropriate address on the available widgets. Blynk was designed for the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, visualize it and do many more.

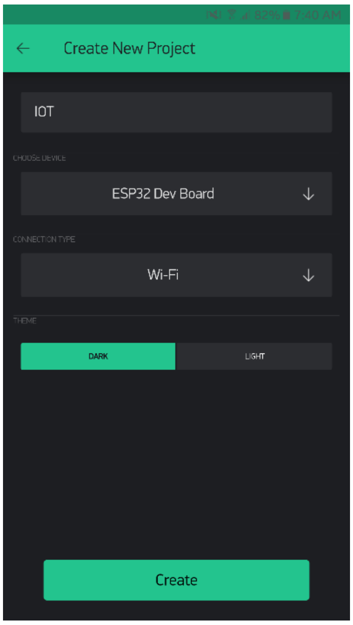
There are three major components in the platform:

Blynk App: It allows you to create amazing interfaces for your projects using various widgets which are provided.

Blynk Server: – It is responsible for all the communications between the smartphone and hardware. The Blynk Cloud can be used or run a private Blynk server locally. It's open-source, could easily handle thousands of devices.

Blynk Libraries: It enables communication, for all the popular hardware platforms, with the server and process all the incoming and outcoming commands.

The Process of work is when someone presses the ‘Button’ in the Blynk application the data will move to Blynk Cloud, where data finds its way to the hardware that has been installed. It works in the opposite direction and everything happens in a blink of an eye.



Create New Project

IOT

CHOOSE DEVICE

ESP32 Dev Board

CONNECTION TYPE

Wi-Fi

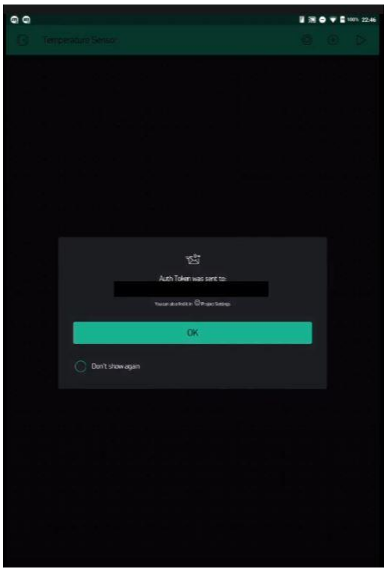
THEME

DARK | LIGHT

Create

Figure 11. Creating New project

After creating new project an authentication key is generated which is then automatically sent to the user via e-mail for the user to later use it in a code.



Auth Token was sent to:
[redacted]
You can also find it in Project Settings

OK

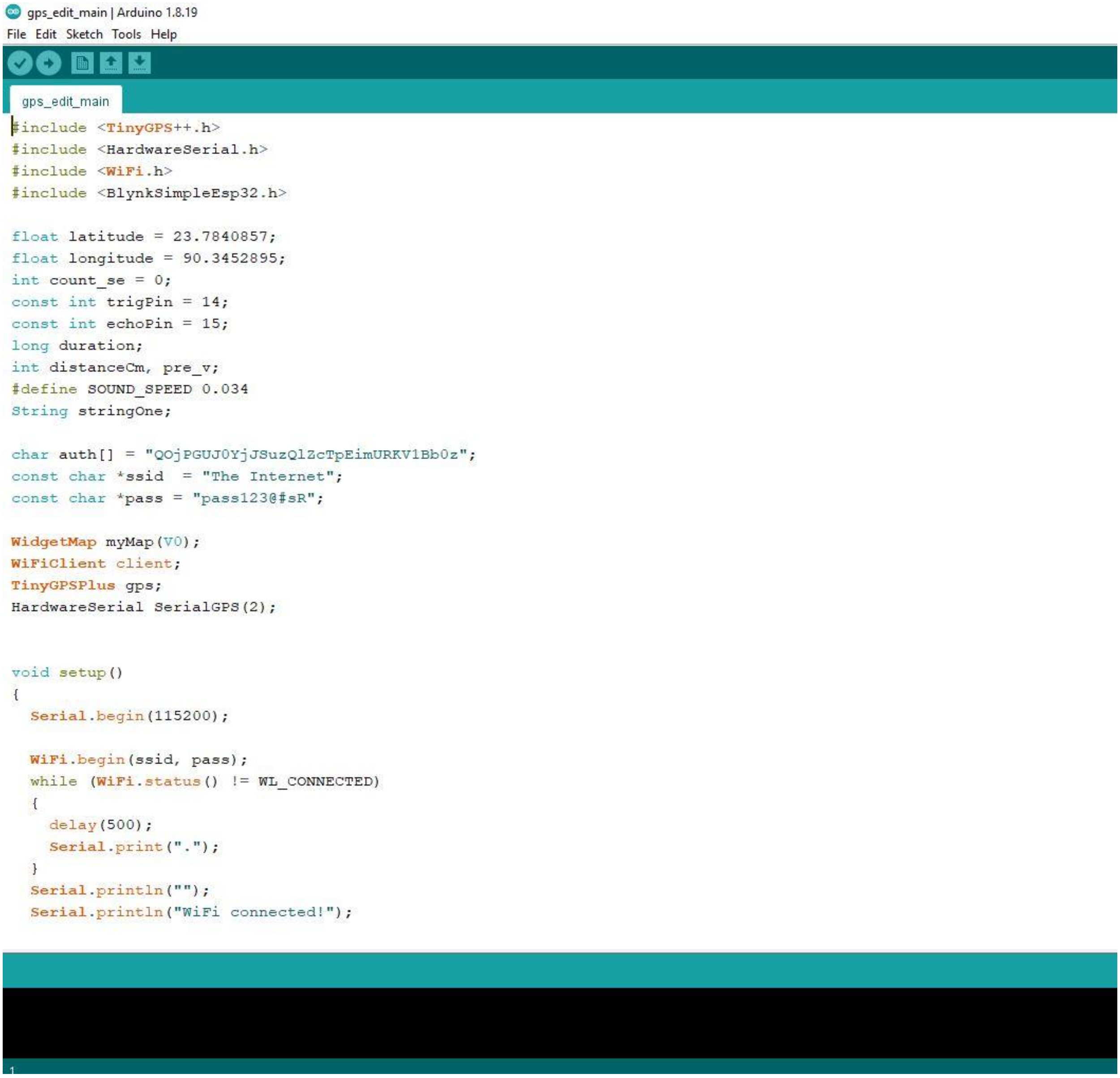
○ Don't show again

Figure 12. Authentication token Generation

Chapter 4

Analysis and Result

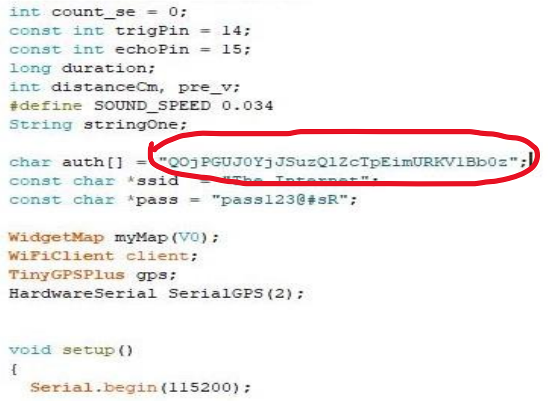
4.1 Writing the Sketch



```
#include <TinyGPS++.h>
#include <HardwareSerial.h>
#include <WiFi.h>
#include <BlynkSimpleEsp32.h>

float latitude = 23.7840857;
float longitude = 90.3452895;
int count_se = 0;
const int trigPin = 14;
const int echoPin = 15;
long duration;
int distanceCm, pre_v;
#define SOUND_SPEED 0.034
String stringOne;

char auth[] = "QOjPGUJ0YjJSuzQlZcTpEimURKV1Bb0z";
const char *ssid  = "The Internet";
const char *pass = "pass123@#sR";

WidgetMap myMap(V0);
WiFiClient client;
TinyGPSPlus gps;
HardwareSerial SerialGPS(2);


void setup()
{
  Serial.begin(115200);

  WiFi.begin(ssid, pass);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected!");
```

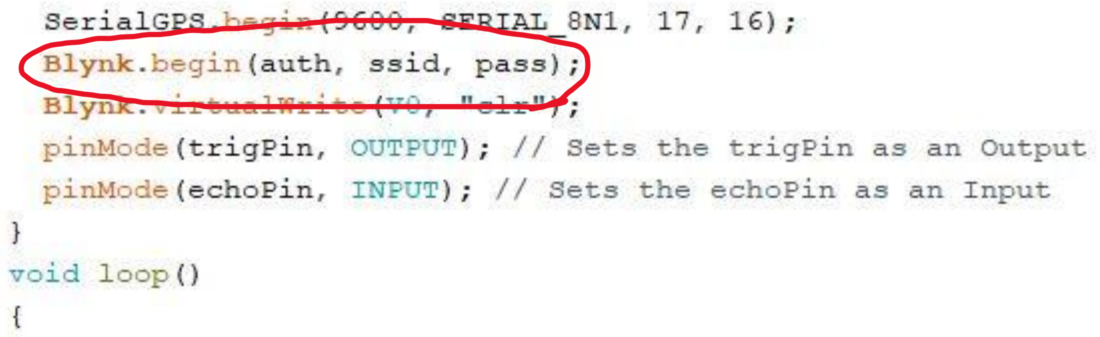
Figure 13. Code part 1

The Code is written using C++. After all the necessary libraries and variable declaration one very important step would be to acquire the Authentication Key that is generated from the Blynk, IOT platform, software to be used. The use of such a key is highlighted in figure 14.



```
int count_se = 0;
const int trigPin = 14;
const int echoPin = 15;
long duration;
int distanceCm, pre_v;
#define SOUND_SPEED 0.034
String stringOne;

char auth[] = "QOjPGUJ0YjJSuzQlZcTpEimURKV1Bb0z";
const char *ssid  = "The Internet";
const char *pass = "pass123@#sR";

WidgetMap myMap(V0);
WiFiClient client;
TinyGPSPlus gps;
HardwareSerial SerialGPS(2);


void setup()
{
  Serial.begin(115200);
```

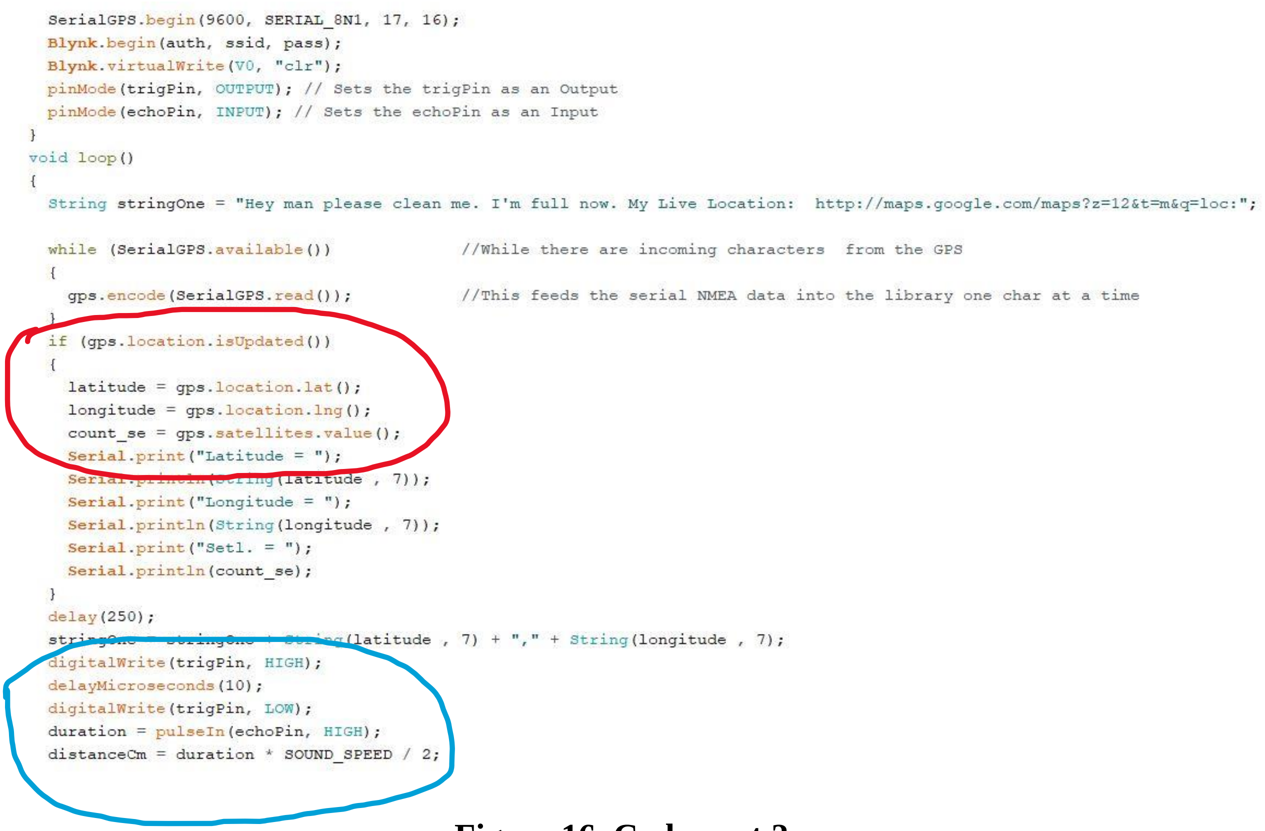
Figure 14. Blynk Authentication Key



```
  SerialGPS.begin(9600, SERIAL_8N1, 17, 16);
  Blynk.begin(auth, ssid, pass);
  Blynk.virtualWrite(V0, "clr");
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input
}
void loop()
{
```

Figure 15. Blynk Initialization

Blynk is initialized using the function shown in figure 15 with the authentication key shown in figure 14.



```
  SerialGPS.begin(9600, SERIAL_8N1, 17, 16);
  Blynk.begin(auth, ssid, pass);
  Blynk.virtualWrite(V0, "clr");
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input
}
void loop()
{
  String stringOne = "Hey man please clean me. I'm full now. My Live Location:  http://maps.google.com/maps?z=12&t=m&q=loc:";

  while (SerialGPS.available())             //While there are incoming characters  from the GPS
  {
    gps.encode(SerialGPS.read());           //This feeds the serial NMEA data into the library one char at a time
  }
  if (gps.location.isUpdated())
  {
    latitude = gps.location.lat();
    longitude = gps.location.lng();
    count_se = gps.satellites.value();
    Serial.print("Latitude = ");
    Serial.println(String(latitude , 7));
    Serial.print("Longitude = ");
    Serial.println(String(longitude , 7));
    Serial.print("Setl. = ");
    Serial.println(count_se);
  }
  delay(250);
  stringOne = stringOne + String(latitude , 7) + "," + String(longitude , 7);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distanceCm = duration * SOUND_SPEED / 2;
```

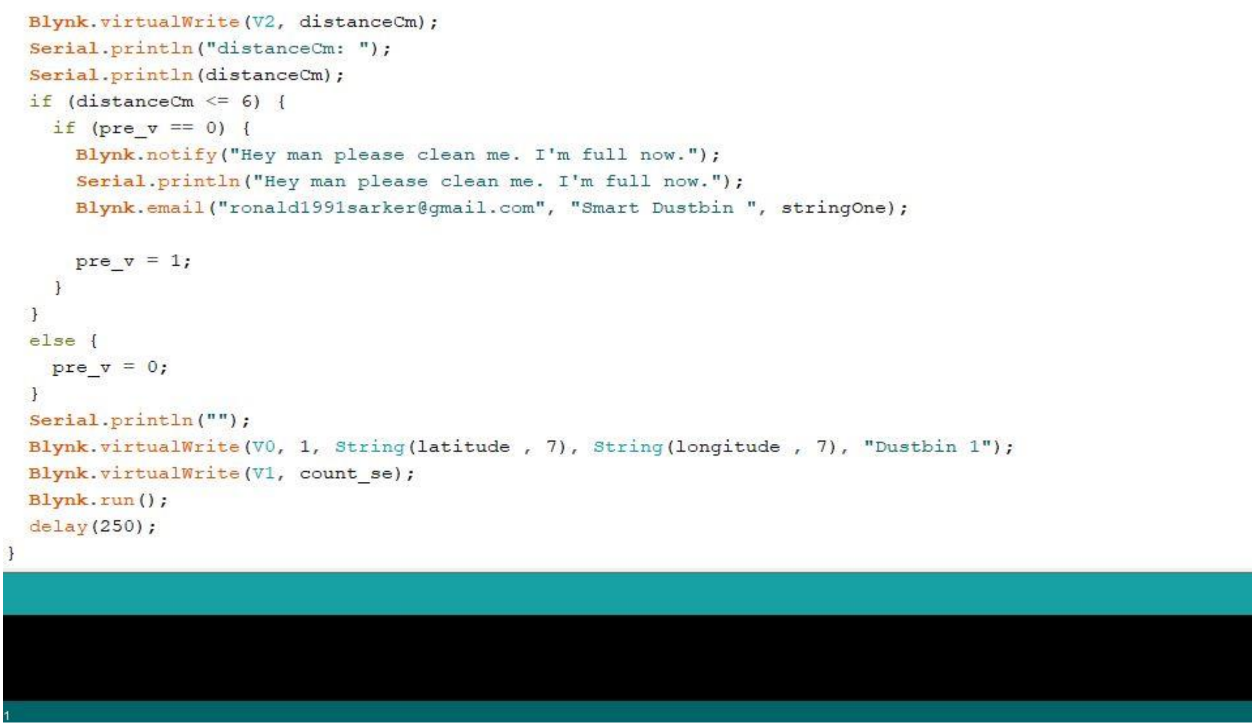
Figure 16. Code part 2

In Figure 13, it is clearly shown how all the basic and default elements are written to initialize all the electronics, sending and receiving the data with each other. Now figure 16 shows the loop which does the main operation, that is the calculation of how much of the bin is used up by using a very simple mathematical formula.

4.1.1 The main operation.

In the red circled zone, the instructions are simple, if the longitude and latitude of the remains constant The location is kept constant and the app interface shows the static location of the bin.

In the blue circle zone, the Ultrasonic Sensor is manipulated. The time taken for the wave to echo back is used for the simple formula (explained in page12) to calculate the distance.



```
  Blynk.virtualWrite(V2, distanceCm);
  Serial.println("distanceCm: ");
  Serial.println(distanceCm);
  if (distanceCm <= 6) {
    if (pre_v == 0) {
      Blynk.notify("Hey man please clean me. I'm full now.");
      Serial.println("Hey man please clean me. I'm full now.");
      Blynk.email("ronald1991sarker@gmail.com", "Smart Dustbin ", stringOne);

      pre_v = 1;
    }
  }
  else {
    pre_v = 0;
  }
  Serial.println("");
  Blynk.virtualWrite(V0, 1, String(latitude , 7), String(longitude , 7), "Dustbin 1");
  Blynk.virtualWrite(V1, count_se);
  Blynk.run();
  delay(250);
}
```

Figure 17. Code part 3

4.1.2 Notifying the user

Here the desired distance is set. As shown in the figure If the distance is less than 6cm, a notification is sent to the app and an email to the app user along the link to google maps with the coordinates of the bin that was last updated when the bin fills up.

4.2 Blynk Interface

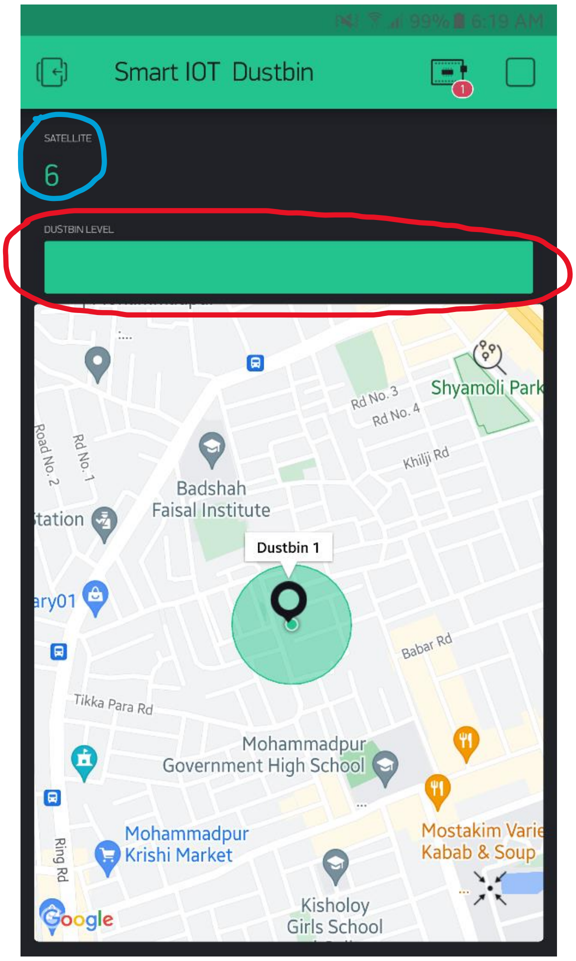


Figure 18. Blynk Custom Interface.

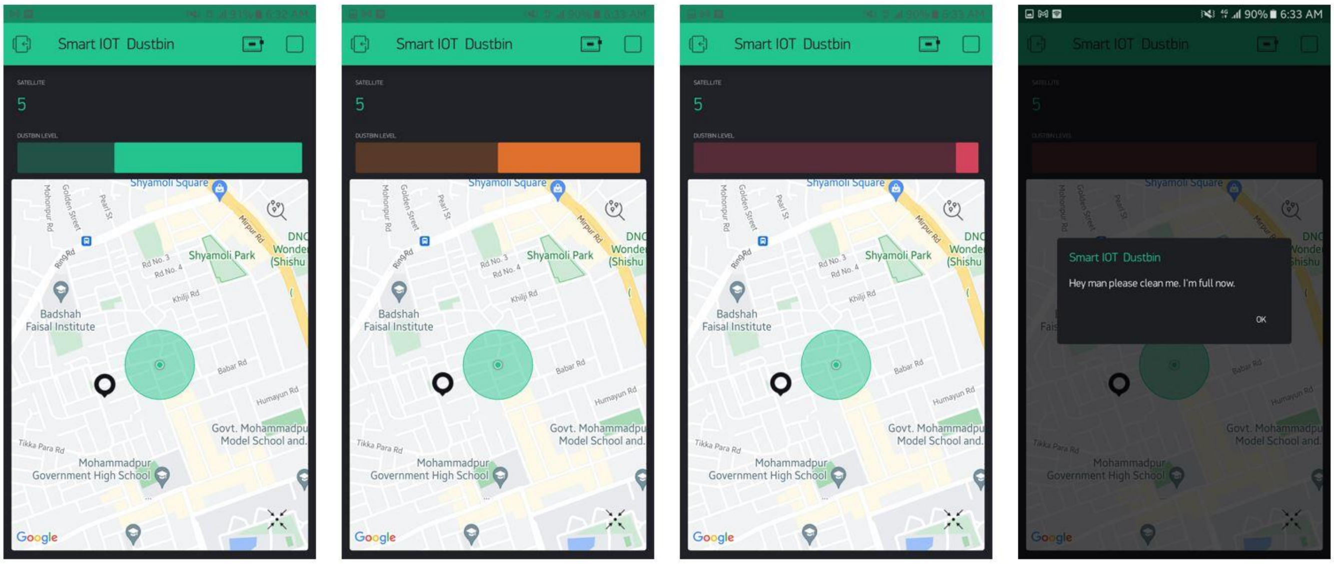
4.2.1 Custom Widgets

As Shown in the figure There are 3 widgets selected for our project.

The one circled in blue is the Satellite Counter, showing the number of satellites connected to the GPS module inside the bin. The virtual pin selected for this widget is v1.

The one circled in red is the progress bar showing the level of trash inside the bin pulling data with help of our Ultrasonic Sensor. The virtual pin selected for this widget is v2.

The third widget is the map itself showing the user’s location in a green icon and the pin point location of the bin using the coordination provided by the GPS module. The virtual pin selected for this widget is v0. In this case we only have one sample bin thus showing only one.



Smart IOT Dustbin

Hey man please clean me. I'm full now.

Figure 19. Trash Level and Notification

4.2.2 Trash Level and Notification

As shown in the figure 14, as the trash level gradually increases the Echoing distance decreases and so does the indicator on the progress bar.

Eventually when it fills up a notification is sent to the user along with the dustbin name, in this case it is named dustbin1.

Chapter 5

Conclusion

Garbage Collection on time is very much important towards having clean and smart society. Traditional Garbage Collection employing human is not very effective with no proper supervision. Research has been done in employing IoT based technology in monitoring the status of bin towards collecting the garbage once threshold reached. There were certain drawbacks in the existing IoT based system which led to the development of IoT based Smart Garbage Collection System employing ultrasonic sensors.

Waste management is indeed a crucial element for any city in order to practice good sustainability, enhancing urban mobility and at the same time maintaining natural resources. These goals should be targeted by any city that intended to become a smart city.

Therefore, an IoT-based 'Waste Management System for Smart City' is Useful in many ways such as monitoring garbage level in real-time, tracking the location of garbage bins, optimizing waste collection route using an algorithm or even as a communication medium between residents and local authorities for better waste disposal and management practice.

5.2 Future Work

This model is developed with the aim to keep environment clean and green. For future work purposes, a better solution of SSWMS can be proposed with enhanced features by taking into consideration the gap in the current solution.

- The scope for the future work is this system can be implemented with time stamp in which real-time clock shown to the concern person at what time dust bin is full and at what time the waste is collected from the smart dustbins.
- If this system is used to monitor dustbins in larger areas, Android app with dustbin locator can be developed so that person can track nearest bin and its status.
- A small grinder can be used along with a wet waste bin to make pieces of organic waste substances so that it will be decomposed rapidly.
- Employing camera sensor for image processing of the cleanliness of the roads and penalizing persons not throwing the garbage properly in the bin.

5.3 References:

[1] Kanchan Mahajan, “Waste Bin Monitoring System Using Integrated Technologies”, International Journal of Innovative Research in Science, Engineering and Technology, Issue 3, Issue 7, July 2014.

[2] M. Al-Meaded, N.K. Madi, Ramazan Kahraman, A. Hodzic, N.G. Ozerkan, An Overview of Solid Waste Management and Plastic Recycling in Qatar, Springer Journal of Polymers and the Environment, March 2012, Volume 20, Issue 1, pp 186-194.

[3] Islam, M.S. Arebey, M.; Hannan, M.A.; Basri, H, “Overview for solid waste bin monitoring and collection system” Innovation Management and Technology Research (ICIMTR), 2012 International Conference, Malacca, 258 – 262

[4] Raghumani Singh, C. Dey, M. Solid waste management of Thoubal Municipality, Manipur- a case study Green Technology and Environmental

Conservation (GTEC 2011), 2011 International Conference Chennai 21 – 24

[5] Vikrant Bhor, “Smart Garbage management System International Journal of Engineering Research & Technology (IJERT), Vol. 4 Issue 03, March-20152000.

[6] Narayan Sharma, “Smart Bin Implemented for Smart City”, International Journal of Scientific & Engineering Research, Volume 6, Issue 9, September-2015.