Location Based Garbage Management System with IOT for Smart City

Project ID : 17-100

Project Proposal Report

Gangodawilage Shehan Brendon Dabarera	IT14005572
Ranaweera Kaluarachchige Ruvinda Ranaweera	IT14005718
Paththini Gamage Dinusha Madusanka Perera	IT14008306
Panangala archchige don vinu randula panangala	IT14006326

Supervised by: Ms.Shashika Lokuliyana

Bachelor of Science Special Honors Degree in Information Technology

(Specialized in Computer System & Networking Engineering)

Department of Information Systems

Sri Lanka Institute of Information Technology

Sri Lanka

March 2017

Location Based Garbage Management System with IOT for Smart City

Project ID : 17-100

Project Proposal Report

(Proposal documentation submitted in partial fulfilment of the requirement for the Degree of Bachelor of Science Special (honors) In Information Technology)

Bachelor of Science Special Honors Degree in Information Technology
(Specialized in Computer System & Networking Engineering)
Department of Information Systems

Sri Lanka Institute of Information Technology
Sri Lanka

March 2017

Declaration

We declare that this is our own work and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Name	Student ID	Signature
G.S.B. Dabarera	IT14005572	
R.K.R. Ranaweera	IT14005718	
P.G.D.M. Perera	IT14008306	
P.A.V.D.R. Panangala	IT14006326	

The supervisor/s should certify the proposal report with the following declaration.
The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

…..................................... ……..............................

Signature of the supervisor Date

Abstract

This proposal, is prepared for the citizens and the municipal council of a smart city which will describe how easily and efficiently garbage can be managed in a city with a location based garbage management system.

This project was inspired by our day to day life on the streets of Sri Lanka and our campus environment. There is also an island wide project (Mega Polis) currently underway which is the first step towards a smart city in Sri Lanka. Our project is intended to fulfill part of this project, namely garbage management and encouraging people to use garbage bins.

Our project aims to optimize waste collection through a system which actively monitors garbage bin levels and utilizes cleaning staff efficiently to clean them. Our app helps the users and the cleaners to locate bins according to their specific needs. We are hoping that by keeping garbage bins partially or fully empty always would encourage people to use the bins and not throw trash everywhere, which intern will keep the surrounding clean.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1.0 Introduction

1.1 Background

Proper waste management is a basic requirement in any kind of an environment. Usually cleaning in these environments are done in the morning and the afternoon. If you take an urban city like Colombo usually there are about 100,000 to 150,000 employees heading for their workstations every morning. For all those people there are just not enough garbage bins available. On streets of urban cities hundreds of people are passing the same location around one minute. Around 60% of people are carrying food covers, polythene bags and plastic bottles. If they put all use the garbage cans, they will be filled in several minutes. When they fill up people just litter their trash around the garbage bins because there is nowhere else to put them. The obvious solution to this is for the cleaning staff to stay near garbage bins everyday till they fill up to clean them. This is not a real solution. It takes way more cleaning staff and costs a lot of money. So it is impractical. The same scenario is happening in workstations. For instance a bank or a government office cafeteria usually has about 5 to 6 garbage bins to serve hundreds of employees. This is simply not enough.

When you consider the negative effects of the garbage bins always being full, there are some notable effects. One of the main effects is the surrounding area starting to smell and be very unpleasant. When the garbage bins are full people put their trash on sides of the garbage bins. When this is done for some time, first it starts to smell bad. So others who come later tend not to go close and throw their trash in the direction of the garbage bins. If there are any leftover food items, throwing it causes them to spill. This attracts animals like cats, dogs and flies. And these animals spill them even more. Another negative effect is the diseases that spread. It’s not just the garbage that spread them, but the animals also can be a source.

1.2 Literature Survey

1.2.1 Smart Bin Implementation for Smart Cities (2015) [7]

This research is focused on creating optimal changes in the conventional methodology of waste collection. This is done by creating a smart bin that will upload the fill levels via SMS. SMS received from the GSM modules of the dustbin is taken in the form of text files. The text file is connected to the excel sheets. The updated values of the dustbin level are taken to form a real time smart bin status. The excel application designed creates a real-time dashboard along with a time series graph which shows the current trend as well as the historical trend of waste level in that particular smart bin. The data collected is then analyzed to gain insights.

These are the hardware components they used.

Hardware Components and Specifications	
Components	Specifications
Microcontroller	PIC-16F73
Ultrasonic Sensor	HC-SR04
GSM Module	SIM-900A, IMEI-865904022247974
Motor	60 rpm DC Motor
LCD	16X2 (JHD162A)
Motor Driving IC	L293D
Voltage Regulator	7805
Resistor	10kohm
Capacitor	100uf,22pf
Oscillator	Crystal Oscillator

Table 1.2.1.1: Hardware Components for Smart bins

1.2.2 Cruisers: A Public Automotive Sensing Platform for Smart Cities (2016) [9]

In this research paper, they mainly focus on how collecting urban data in a city-wide scale plays a fundamental role in the research, development and implementation of smart cities. This research introduces Cruisers, an automotive sensing platform for smart cities, which is developed based on the following ideas.

a) Garbage collecting trucks are used as host automobiles to accommodate sensors

b) 3G cellular communication network is used to wirelessly deliver sensed data directly to servers

c) Proxy server(s) are adopted to convert the format of sensed data to required ones.

This platform has been deployed to 24 garbage collecting trucks at Fujisawa city, i.e., nearly 1/4 of the total number of such trucks in the city. An iOS application is also developed to demonstrate the sensing process and the covered area.

The technology they propose are Cruisers consists of a collection of sensor nodes installed into the same number of garbage collecting trucks, one proxy server and one data server. In their implementation, sensor nodes are deployed into 24 garbage collecting trucks at Fujisawa City. There are 100 such trucks in the city, so our implemented system is expected to cover about 1/4 area of the city with one acceleration sensor and four environmental sensors as well as a GPS receiver. They are developing a Java program to control the sensor node.

1.2.3 IoT Based Solid Waste Management System A conceptual approach with an architectural solution as a smart city application (2016) [10]

One of the major hurdles in most cities is its solid waste management. This paper aims at providing an IoT based architectural solution to tackle the problems faced by the present solid waste management system. By providing a complete IoT based system, the process of tracking, collecting, and managing the solid waste can be easily automated and monitored efficiently. By making use of sensors, we collect data from the garbage bins and send them to a gateway using LoRa technology. The data from various garbage bins are collected by the gateway and sent to the cloud over the Internet using the MQTT (Message Queue Telemetry Transport) protocol. The main advantage of the proposed system is the use of LoRa technology for data communication which enables long distance data transmission along with low power consumption as compared to Wi-Fi, Bluetooth or ZigBee.

To implementation they use 802.15.4 standard. This is a new technology and is best suited for the Internet of Things. The LoRa module uses low power for communication and has a transmission range of around 30 Km LOS. LoRa is a combination of RF and IP based communication. The sensor nodes send the sensor data over the ISM RF channel to the gateway device. Also, they use the gateway then combines the data and sends it to the cloud over IP using LAN, Wi-Fi or 3G/4G. They are using the LoRaWAN in the link layer and the MQTT protocol in the application layer. They are making use of a Linux based gateway device. Protocol which they use is MQTT. It is a light-weight messaging protocol. For the implementation four garbage bins were taken for implementing the prototype. Each bin was fitted with the sensors, microcontroller and the communication module. Atmel’s ATmega328p microcontroller was chosen for the prototype.

1.2.4 Using Genetic Algorithm for Advanced Municipal Waste Collection in Smart City (2016) [11]

In this research paper, they mainly focus on IoT vision that introduces promising and economical solutions for massive data collection and its analysis which can be applied in many domains and so make them operate more efficiently. To optimize the logistic procedure of waste collection, they use own genetic algorithm implementation. The presented solution provides calculation of more efficient garbage-truck routes. As an output, we provide a set of simulations focused on mentioned area. All their algorithms are implemented within the integrated simulation framework which is developed as an open source solution with respect to future modifications.

The algorithms they use are Floyd-Warshal, TSP formulation, crossover algorithm, Mutation algorithm and Dijkstr, TSP formulation Algorithm

Used Floyd-Warshall algorithm for graph processing

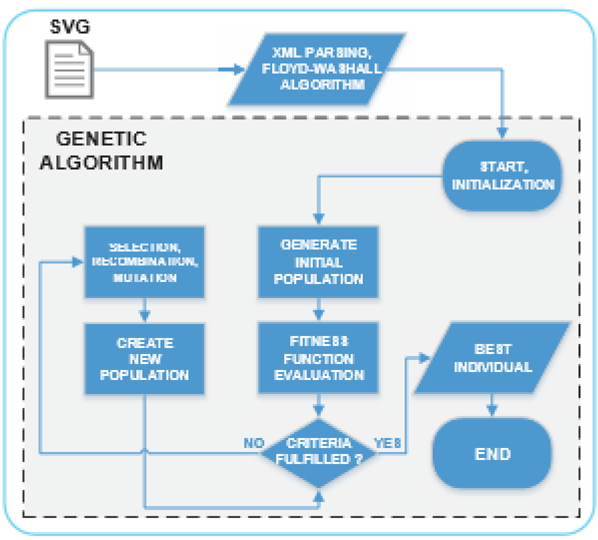

Figure 1.2.4.1: Genetic Algorithm

1.2.5 Top–k Query based Dynamic Scheduling for IoT-enabled Smart City Waste Collection (2015) [8]

This paper proposes a top-k query based dynamic scheduling model to address the challenges of near real-time scheduling driven by sensor data streams. An Android app along with a user-friendly GUI is developed and presented in order to prove feasibility and evaluate a waste collection scenario using experimental data.

In implementation regarding spatial information the Smart City is divided into multiple sectors which cover the entire city area. Each sector contains a number of multiple intermediate waste depots, which are temporary waste storage areas. The proposed system architecture incorporates a heterogeneous fleet of trucks for serving the waste collection infrastructure.

Cloud middleware is responsible to collect data from sensors, aggregate and clean them in order to provide them to the in engine which is implemented in OpenIoT. Data are stored in a spatial database in which mobile top-k queries specify the number of the full bins in order to initiate dynamic scheduling.

TABLE I. DYNAMIC SCHEDULING ALGORITHM

Input: *Bins*
Output: *k*
While (*true*) **Do**
    k = top−*k* query from relation *Bins*
    /*Select *k* *red* bins to load*/
End While

Figure 1.2.5.1: Dynamic Scheduling Algorithm

They use the dynamic scheduling algorithm which locates the first available truck which can load waste from the filled bins. Then it is performed a top-K query which exploits real time data from the relation. Also data are stored in a spatial database in which mobile top -k queries specify the number of the full bins in order to initiate dynamic scheduling. An Android app is implemented in order the drivers to have a user-friendly GUI interface with the IoT system.

1.2.6 Cloud Computing Based Smart Garbage Monitoring System (2016) [13]

In this paper, they present a solution about the Smart Bin is a network of dustbins which integrates the idea of IoT with Wireless Sensor Networks. They also put forward the concept of a network of smart garbage bins based on the Stack Based Front End approach of integrating Wireless Sensor Network with the Cloud computing and discuss how Machine Learning techniques like Decision Forest Regression can be applied to the sensor data leveraged by the system to gain useful insights to improve the efficiency of the garbage monitoring.

Hardware used:

- WSN Motes
- IRIS
- Ultrasonic sensors

Software used:

- TinyOs-2.1.2
- NesC
- Azure IOT Hub

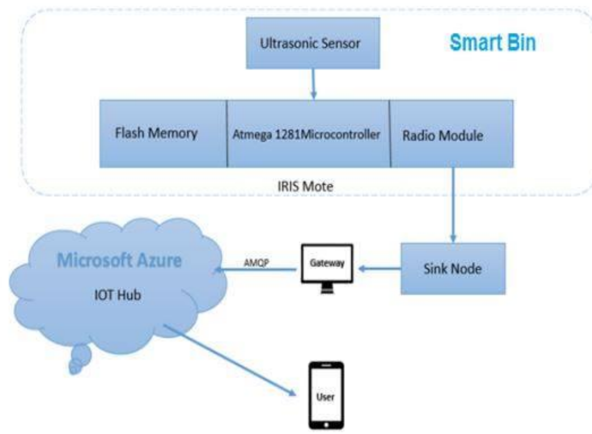

Figure 1.2.6.1: System Overview

Their windows app is coded by C#. It consists of two modes, User and Admin. The admin can add new devices, monitor fill levels and get reports about failures. The user can only view all the bins that has been deployed in the area by entering the area code that has been set up by the admin. They are using Bing Maps API to show the geographical location of the bin. The Admin can also get the shortest path comprising of all the filled bins and can redirect garbage vans.

1.2.7 IOT Based Smart Garbage alert system using Arduino UNO (2016) [14]

This paper proposes a smart alert system for garbage clearance by giving an alert signal to the municipal web server for instant cleaning of dustbin with proper verification based on level of garbage filling. This process is aided by the ultrasonic sensor which is interfaced with Arduino UNO to check the level of garbage filled in the dustbin and sends the alert to the municipal web server once if garbage is filled. After cleaning the dustbin, the driver confirms the task of emptying the garbage with the aid of RFID Tag. RFID is a computing technology that is used for verification process and in addition, it also enhances the smart garbage alert system by providing automatic identification of garbage filled in the dustbin and sends the status of clean-up to the server affirming that the work is done. The whole process is upheld by an embedded module integrated with RFID and IOT Facilitation.

An Android application is developed and linked to a web server to intimate the alerts from the microcontroller to the urban office and to perform the remote monitoring of the cleaning process, done by the workers, thereby reducing the manual process of monitoring and verification. The notifications are sent to the Android application using Wi-Fi module.

The proposed e-monitoring system is an embedded system that comprises of RFID technology interfaced with Arduino microcontroller and a web base which is completely computerized. By employing the proposed system, the municipal authority could monitor the waste collection status effectively. The e-monitoring system has two parts:

• Embedded system: It comprises of an RFID reader, a microcontroller, a Liquid Crystal Display (LCD) and a GPRS segment.

• Web based software system Interface: It comprises of a GPRS module, a Central Server, a Database Server and a Web server. The Figure reveals the block diagram of the web centered software system.

The overall flow diagram of the proposed model is shown in below Figure.

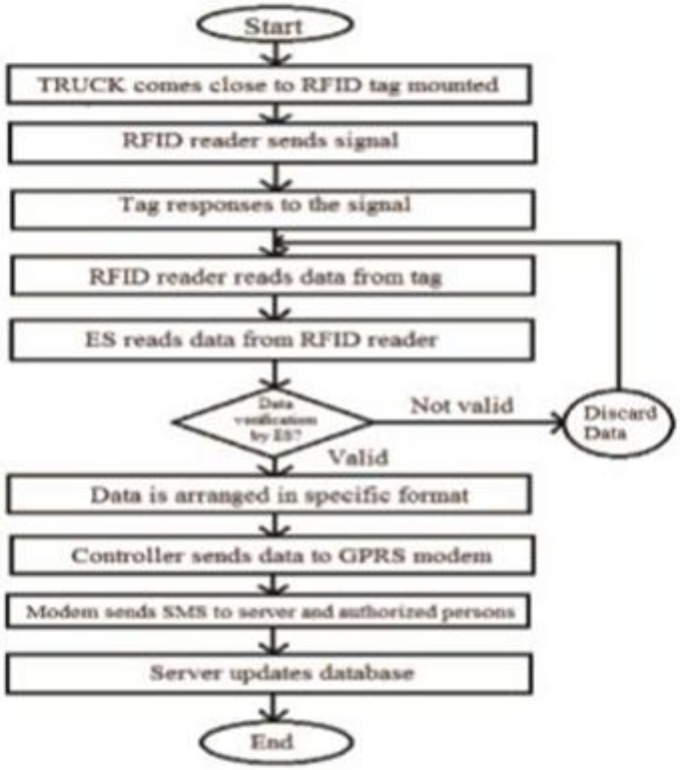

Figure 1.2.7.1: System Structure

1.2.8 Solid Waste Management Architecture using Wireless Sensor Network Technology (2012) [1]

This paper is focused on the on-site handling and storage processes and on the transfer process, with the main topic at developing a smart solid waste management system capable to ensure the public health with costs reduction and quality improvement. In order to enhance the efficiency of solid wastes on-site collection and transfer, an innovative solution for the monitoring and management system has been proposed. A Wireless Sensor Network (WSN) has been developed to improve the garbage bins monitoring process.

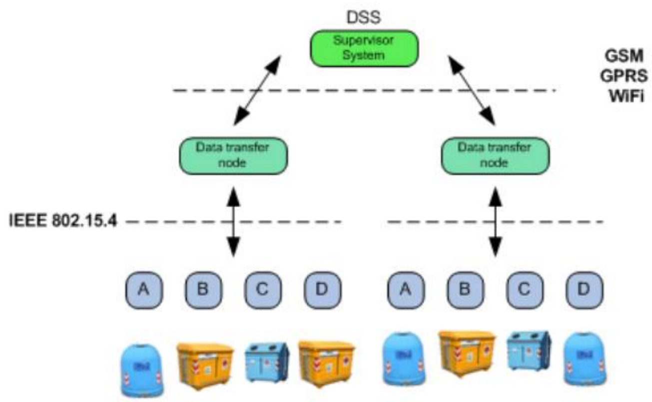

Figure 1.2.8.1: SEA project system architecture

The main components of the developed architecture on the SEA project are decomposed into three layers. Starting from the bottom, the garbage bins reside at the first one, the DTNs at the second one and the supervisor system at the third one.

- The first layer is composed by sensor nodes, which need for the filling monitoring and provide short-range transmissions through RF technology
- The second layer is composed by communication modules, which provide long-range transmission through GSM/GPRS;
- The third layer is composed by servers, which provide data storage and supervisor modules.

1.3 Research Gap

Most research papers have implemented garbage bins, but they are just a bin.

IOT Based Smart Garbage alert system using Arduino UNO has used an arduino board, this poses problems because arduino does not support multi-threaded programming [14]. To avoid this we decided to implement the Raspberry pie zero development board, which is about the same price as the Arduino Uno and supports multithreaded programming.

Most of the papers have used only one level measuring sensor in their bins. This poses a problem because garbage levels can be at different levels at different areas of the bin. To avoid this, we are implementing four sensors which will be much more accurate at reading the actual level of the garbage level.

Different research papers have used different methods to transfer data from the bin.
Smart Bin Implementation for Smart Cities has used GSM technology [7].
Cruisers: A Public Automotive Sensing Platform for Smart Cities has used GSM technology [9].
IoT Based Solid Waste Management System A conceptual approach with an architectural solution as a smart city application has used LORA technology [10].
IOT Based Smart Garbage alert system using Arduino UNO has used wifi
Solid Waste Management Architecture using Wireless Sensor Network technology has used RF and GSM technology
Most of these papers has used GSM technology. When using GSM you should go through the hassle of registering to a subscriber and has pay for the services. We have chosen wifi because in a smart city everything would be connected by wifi. So we just have to connect to the existing communication system without paying.
These are the technologies that were used in the papers that we mentioned.

So in addition to the technologies we have used, we have implemented:

- A security system to sound an alarm
- A hand gesture system to open and close the bin door
- An auto lock feature to lock the bin at 100% capacity

After the sensor data has been transferred different papers have used different methods to manipulate data,

Smart Bin Implementation for Smart Cities [7] has used an Excel application, which has some weaknesses such as

- Vulnerable to fraud
- Susceptible to trivial human errors
- Difficult to troubleshoot or test
- Obstructive to regulatory compliance
- Not designed for collaborative work

Most of the other papers has used cloud based systems to analyze and monitor data. We find no error in doing so as this is the best method to do this currently. So we have also taken the cloud based approach.

Smart Bin Implementation for Smart Cities, Cloud Computing Based Smart Garbage Monitoring System [7].

Smart Bin Implementation for Smart Cities [7] has used an android application to notify cleaners, a Cloud Computing Based Smart Garbage Monitoring System [13] is use windows application and some uses only SMS notifications. We have improved upon this to give notifications through our android application and SMS notification for better efficiency.

Though most papers have focused on optimizing garbage collection, based on what we read we the method we propose is better than most of the algorithms these papers provide.

Cloud Computing Based Smart Garbage Monitoring System proposes a route calculation between only two static points [13].

Cruisers: A Public Automotive Sensing Platform for Smart Cities has used a method that divides the city into sectors, this method can waste fuel and time [9].
Our method proposes a predictive system that predicts and clean bins that reaches certain levels even before they reach it.

1.4 Research Problem

Usually garbage bin cleaning in an urban city is done once a day, when it gets more further away the bins are not cleaned at least weekly. Usually an urban city has around 10000 citizens, but there are simply not enough garbage bins to serve all these people. This encourages more and more people to litter on the sides of roads, rivers, etc. These problems lead to many other problems such as :

* Diseases such as Dengue fever, malaria, etc. spread

* Attraction of wild animals

* Buildup of unpleasant odors

* Buildup of harmful toxins

The above-mentioned effects are some specific effects that can be seen. But the ultimate problem all this leads to is environmental pollution.

What we see is that the municipal council lack the tools to take any actions regarding these problems. They lack a proper garbage management system and lacks any sort of way to encourage people to use garbage bins, because they do not clean the bins in any periodic way. No matter what they do to encourage people, without a way to give people a convenient way to dispose of the garbage they collect every day, their efforts are hopeless.

2.0 Objectives

2.1 Main Objective

- Build a cost effective garbage management system for the municipal council that will help them keep the city a cleaner place.
- Build a system (Garbage bin/ android application/ website) that will encourage people to use garbage bins instead of littering.

2.2 Specific Objective

With the increase in population, the scenario of cleanliness with respect to garbage management is degrading tremendously. The overflow of garbage in public areas creates an unhygienic condition in the nearby surrounding. It may provoke several serious diseases amongst the nearby people. It also degrades the valuation of the area. To avoid this and to enhance the cleaning, 'Location Based Garbage Management System with IOT for Smart City' is proposed in this project. Our main objective is to make a better place for humans to get there day to day work done.

- User Satisfaction:

By adding several features we are try to motivate people to use garbage bins instead of littering garbage here and there.

- Environmentally Effective:

Garbage management system must be able to protect and reduce the environment from improper disposal of waste that can cause hazard (emissions to land, air and water, such as CO_2, CH_4, CO and heavy metals).

- Economically Affordable:

The co-operative management should be able to afford the cost placed on garbage management operations by waste management system. Means that the cost of effective waste management system should consider the living standard of people in such community.

- Socially Acceptable:

For garbage management system to operate effectively, public cooperation is important. Moreover, they should always try to provide vital information, educate, develop trust and gain support from the community. Provision of bins or containers for collection and sorting of waste is another means by which recycling can be effective in the communities.

3.0 Methodology

The proposed location smart trash can system can be broken into three main functions which have sub categories.

- Planning: In here we are going to discuss how we manage time and how we going to accomplish our project by reducing risks.
- Development: In here we are going to describe how we are going to develop our structure of our project.

3.1 Planning

We were inspired to do this project when we saw most of the populated cities in our county and how much of money government spend for garbage management but still the garbage bins were overflowing and had attracted wild animals (etc: An elephant which had been regularly eating garbage at Manampitiya). The whole area was very unpleasant. This gave us the idea to build an efficient management system for garbage collection. So we visited urban areas to do some research on this topic and found the same result. These were the results we found

Figure 3.1.1: Colombo City

Figure 3.1.2: Cafeteria

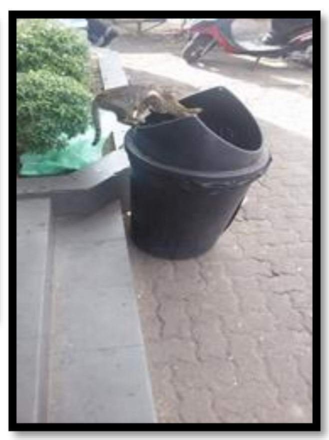

Figure 3.1.3: Kiribathgoda City

3.1.1 System Overview

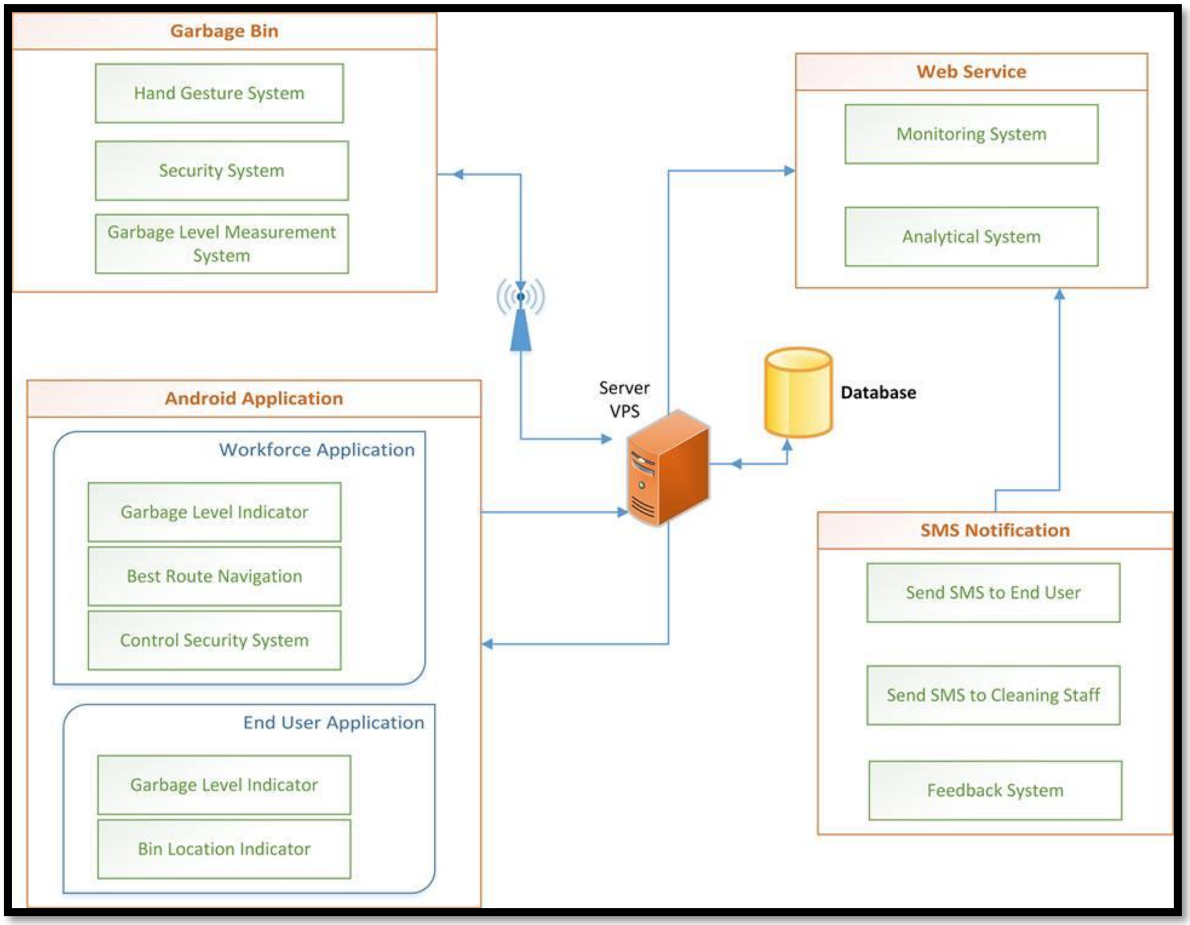

Figure 3.1.1.1: High Level System Architecture

3.2 Development

3.2.1 Setting up the hardware

3.2.1.1 Selecting a Development Board

For selecting hardware first of all we had choose a development board for the basic setup. There are many development boards available in the market, but we selected the most suitable board for our research requirements.

Our basic requirement are:

- Microchip type and the size
- Memory size (RAM)
- Number of I/O pins
- Parallel process availability
- Inbuilt features (Wifi, Bluetooth)
- Ethernet Protocol
- Price of the board

By going through the above requirements and which boards are available in the market we have selected 3 board types for further comparison.

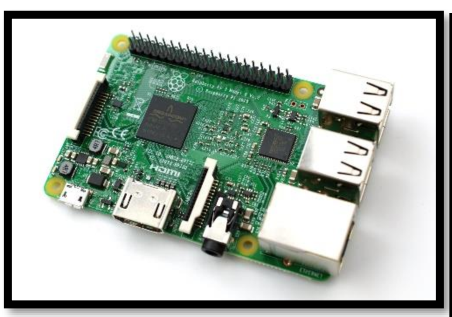

Figure 3.2.1.1.1: Raspberry pi 3

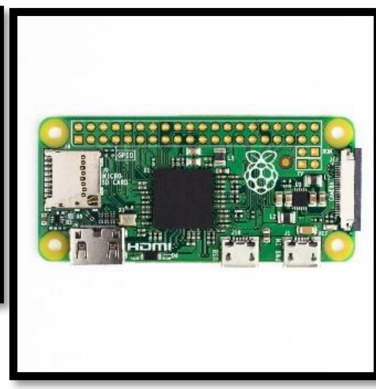

Figure 3.2.1.1.2: Raspberry pi zero [20]

Figure 3.2.1.1.3: Arduino

	Development Board		
Requirement	**Arduino UNO**	**Raspberry pi 3**	**Raspberry pi zero**
Microchip type and the size		A 1.2GHz 64-bit quad-core ARMv8 CPU	1GHz ARM11 core
Memory size (RAM)	32 KB (ATmega328P)	1GB RAM	512MB of LPDDR2 SDRAM
Number of I/O pins	26(analog and digital)	40 GPIO pins	40 GPIO pins
Parallel process availability	no	yes	yes
Inbuilt features (Wifi, Bluetooth)	no	WIFI and Bluetooth both	Bluetooth
Ethernet Protocol		802.11n Wireless LAN	If there is wifi inbuilt it will support 802.11n Wireless LAN
Price of the board	$50	$80	$5

Table 3.1.1.1: Development Board Comparison [20] [21] [22]

With the unavailability of the parallel processing in Arduino comparison, we will turn over to Raspberry pi 3 and zero.

Raspberry pi 3 has more memory and inbuilt features than the raspberry pi zero. But comparison with the price of the both boards there is 1:16 price ratio between pi zero and 3. There are pros and cons in both boards according to the comparison. For our product pi zero will be the most suitable development board to select.

3.2.1.2 Measure the garbage level

Selecting sensors was the next step. To measure the trash can level initially we selected two sensors.

- Ultrasonic - Supply voltage:5V, Global Current Consumption:15 mA, Ultrasonic Frequency:40k Hz, Maximal Range:400 cm, Minimal Range:3 cm,Resolution:1 cm, Trigger Pulse Width :10 µs, Outline Dimension: 43x20x15 mm. ultrasonic sensor is non-contact distance measurement module.
- Infrared - Working voltage: DC 3V-12V;Static power consumption: <0.1mA; Delay time: 2 seconds; The blocking time: 2 seconds; Trigger: Can be repeated; Working temperature: -20 - + 60 degree. This module detects the distance 2 ~ 30cm, detection angle 35 °, the distance can detect potential is adjusted clockwise adjustment potentiometer to detect the distance increases; counterclockwise to adjust the potentiometer to reduce the detection distance

Ultrasonic sensors use high-frequency sound pulse to identify the objects by the reflected sound pulse. If the reflection media was rough or damaged, distance calculation of the sensor will be wrong. But using Infrared sensors those error readings will be minimum.

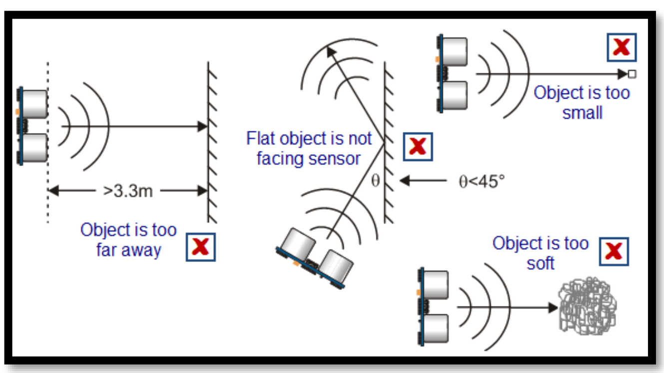

Figure 3.2.1.2.1: Ultrasonic Sensor Diagram

With the above key points adding to the account we've been convinced using infrared will be the best option.

3.2.1.3 Uploading data to Server

For sensor data transferring we've selected wifi as a transmission media. Raspberry pi 3 has an onboard wifi chipset but we selected pi zero as a development board which doesn't have a wifi sensor. XBEE and ESP8266 was our final sensors to compare.

Range	**Power Consumption**	**Frequency**	**Protocol**	**Price**
300 Ft	50mA @ 3.3v	2.4GHz	XBee® 802.15.4	$24
250 Ft	< 1.0mW @ 3.3v	2.4GHz	TCP/IP protocol stack	$$2.46

Table 3.2.1.3.1: WiFi Module Comparison [16]

With the comparison mention above ESP8266 will be the suitable module to select. Because it is the cheapest and the wider range wifi module available in the market.

3.2.1.4 Auto open and close trash bin lid with locking Mechanism

For the motion deection of the hand, according to the motion to open and close the lid of trash can, two ultrasonic sensors and a servo motor will be used.

- The left sensor configured to open the lid once a hand is detected.
- The right sensor configured to close the lid once a hand is detected.

The closing of the lid is done by the servo motor.

- When the left sensor detects, the servo motor rotates 90º (opening the lid). When the right sensor detects, servo motor rotates back the 90º to 0º (closing the lid).

Figure 3.2.1.4.1: Servo Motor

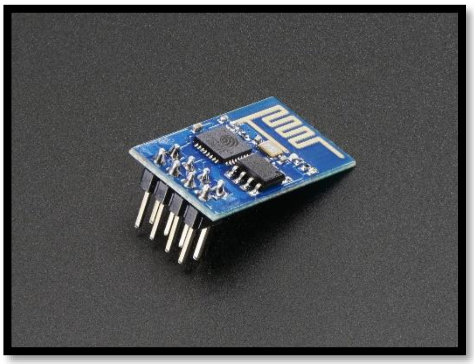

Figure 3.2.1.4.2: ESP8266 Wifi Module

Figure 3.2.1.4.3: Ultrasonic Sensor

We designed a suitable trash can which had the capability of housing the features that we were about to implement. Proposed design of the trash can will be as follow.

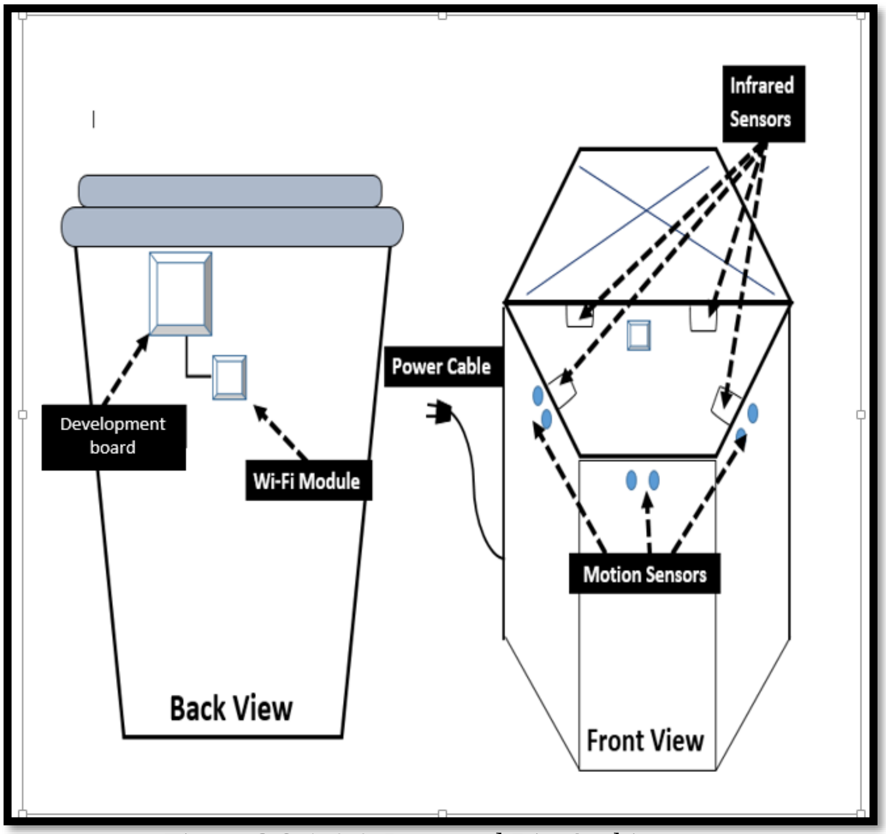

Figure 3.2.1.4.4: Proposed Bin Architecture

Planning to set the sensors in the right positions. That is:

- Placing the two ultrasonic sensors on the outside of the garbage can so that it can detect the hand gesture to open the lid of the trash can.
- Placing the garbage level sensing infrared sensor at an optimum height to detect the level accurately.
- Placing the microcontroller on the back of the bin so as to not get accidentally damaged.
- Placing the Wi-Fi module on a position that can have maximum signal strength from the access point

The following flow chart will show us how the sensors corporate with each other .

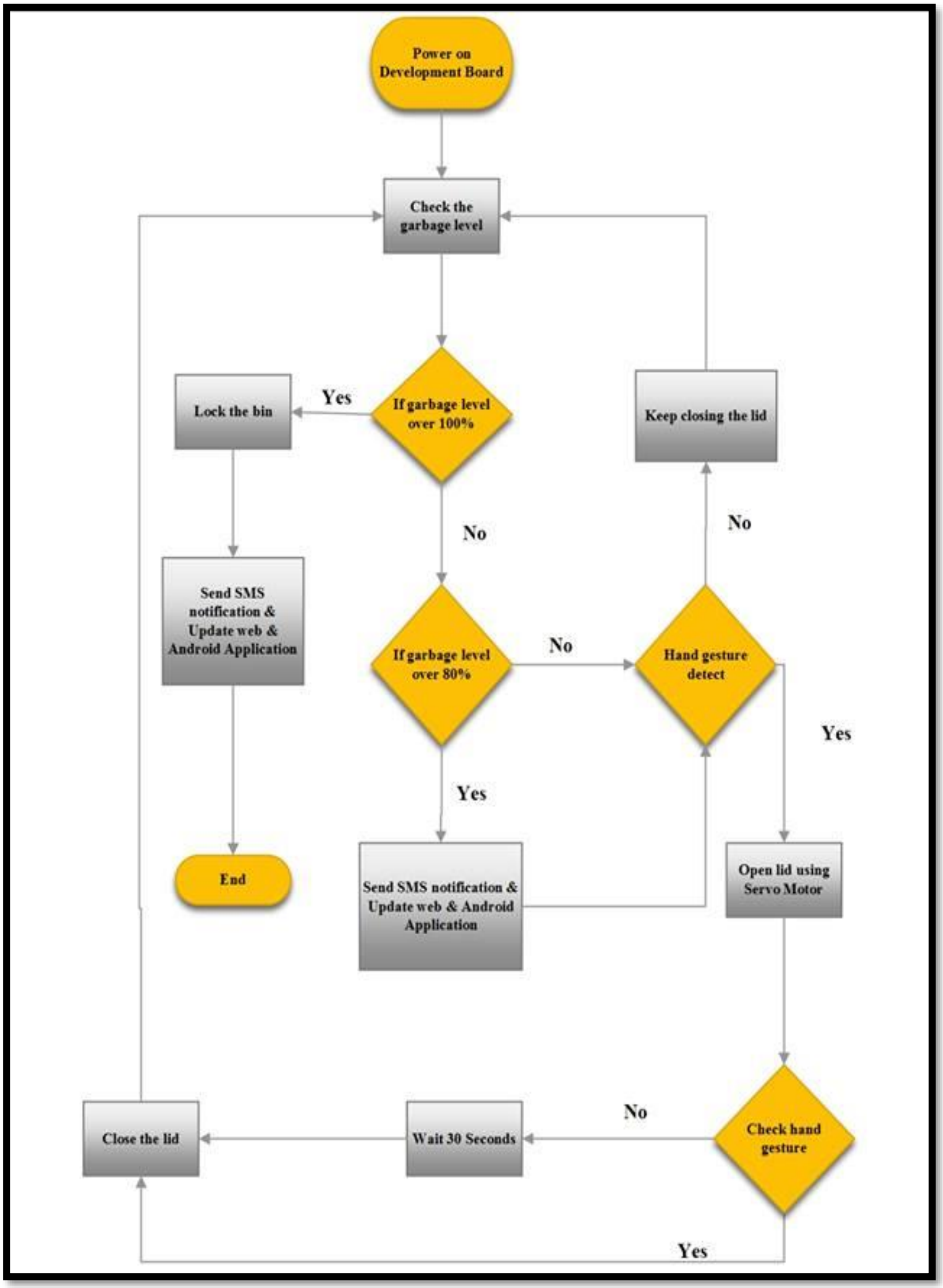

Figure 3.2.1.4.5: Bin Flowchart

3.2.2 Setting up the Server

First a VPS has to be bought. We are thinking of purchasing one from AmazonAWS, since they seem to have VPS that fit our specific needs. These are the different servers available

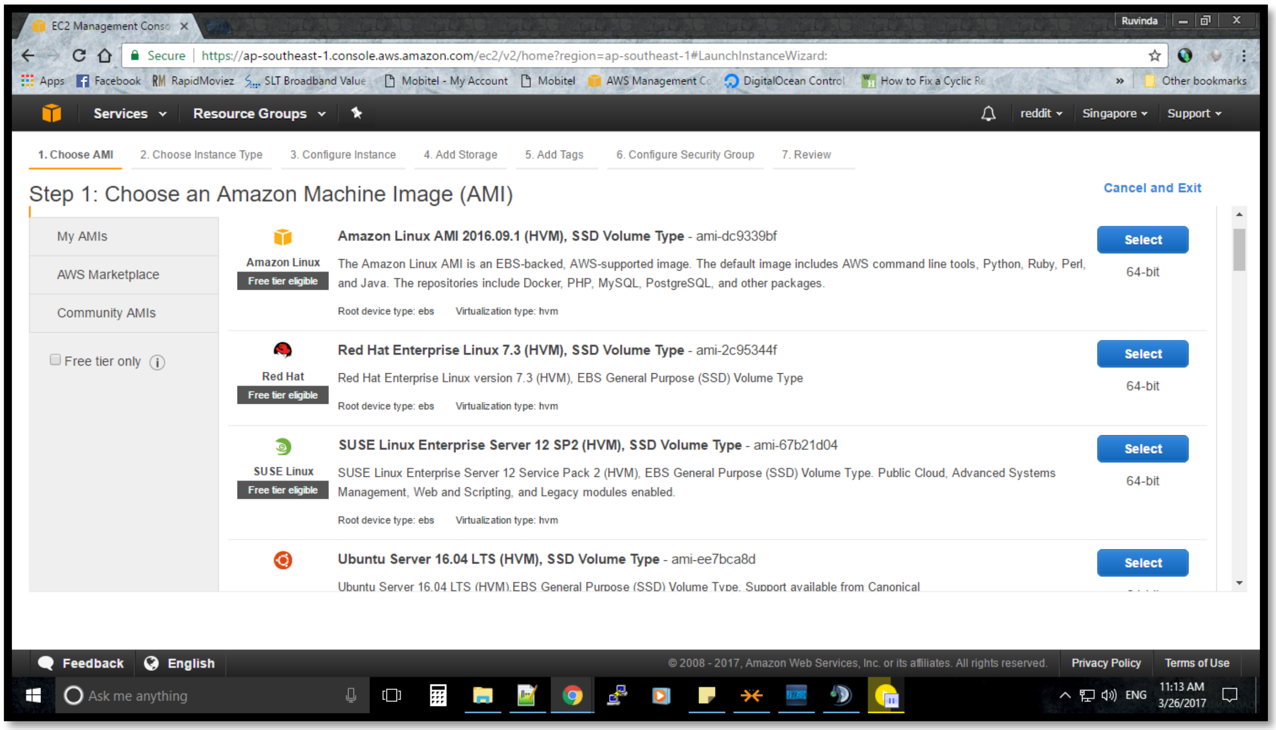

Figure 3.2.2.1: AmazonAWS

Once the VPS is bought these are the software that will be used to setup the server and website

Software	Task
Puttygen	To create private key
Putty	To connect to VPS
WinSCP	FTP client to transfer files
LAMP server	Package includes : Apache web server : MySQL database : PHP language

Table 3.2.2.1: Using Software

3.2.2.1 Website Setup

The purpose of the website is for monitoring and controlling to system remotely. It provides more reliable interface for users to monitor the data regularly. This web interfaces provide information about each and every Trash Can in the system. Also it provides the level of available space or the level of used space in real time. The website provides notifications when the trash bin is filled up to 80% or 100%. To monitor data in this web interface, the data is presented as tables, pie charts or bar charts. To develop this web system we would use HTML, CSS and PHP as a programming language and use MYSQL as the database.

There is two type of users that can be login to this website. There are normal users and administrators or managers. The normal user can only view information, give feedback, request bins on specific locations, etc. Other than that they can edit their profiles. Administrators and managers can view, insert, delete and update information, get annual or monthly reports, etc. The website contains the main monitoring system of the whole project.

3.2.2.2 SMS notification and feedback system

In this system we are building two separate account groups for the workforce staff and end users.

For the working staff we will send a sms in following manner.

When garbage bin fill level is 80% , it will send a sms to relevant the cleaning member with the garbage bin location and garbage bin id by telling this garbage bin is 80% filled.

When the garbage bin is 100% filled, it will send a sms to the relevant cleaning member by telling that the bin has to be cleaned immediately.

For the end user we are building simple web interface. Which can send sms to citizen to inform about our news etc. If there is a breakdown going on and our service is not available for certain days or hours, we can inform citizen for their convenience.

Also by accessing the web interface users can give feedback for our system. Etc: if in a certain area garbage is not being manged properly or a certain garbage bin is not working properly, users can inform us via our web interface. These systems will improve the communication between users and our system to provide a better service.

On the server side updates are taken from all the bins and messages are sent out to cleaning staff accordingly. A virtual map is created to find which bin is placed on which zone of each floor. Every bin is given a unique id, so that messages can be individually identified according to each bin.

When an update is received, cleaning staff is notified using two methods. Via SMS or by the monitor placed in the staff room. We are developing a monitoring application that can collect the information sent from the sensors and analyze them. Cleaning staff is divided into small teams and the server manages and sends notifications to teams in the most efficient way.

3.2.3 Android Application Development

For development purposes we have decided to build two separate applications for the workforce and the user. After the two applications have been developed we hope to merge them into one final application.

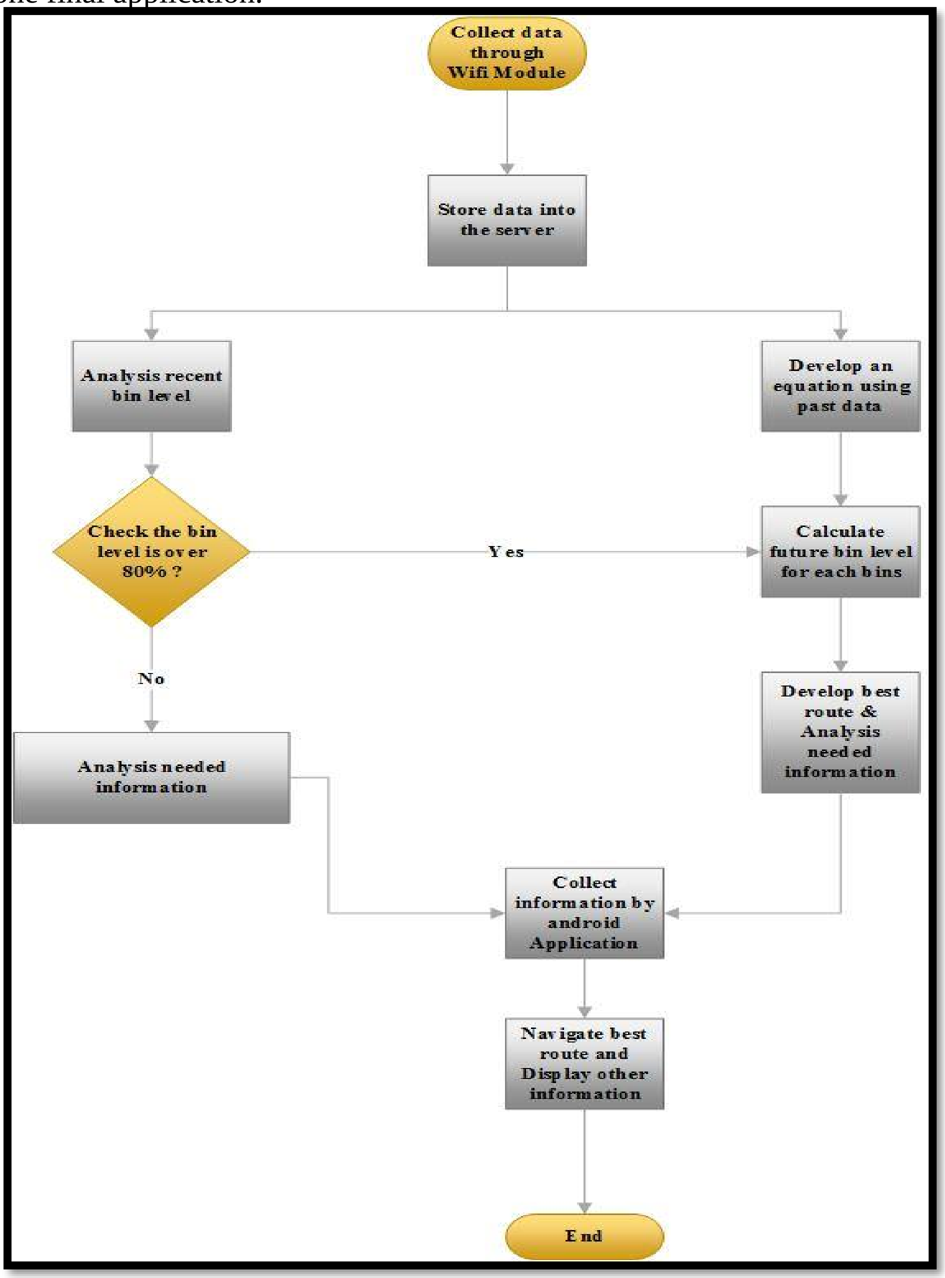

Figure 3.2.3.1: Android Application Flowchart

3.2.3.1 Workforce Application

There are three functions on the workforce application. Showing a map of all the bins of the city with the real-time levels, showing the best route calculated by the server which the cleaner has to take and disabling/re-enabling the security system of a bin.

3.2.3.2 Best route

When a certain bin reaches 80% fill level the cleaner receives the best route calculated to the 80% filled bin from the base station. This route is calculated taking into consideration some other bins that will be filled in a certain period of time into the future. These extra bins will be added as waypoints into the route. This route calculation mechanism is further explained later.

3.2.3.3 Real time levels

The map shows all the bins that are placed throughout the city. The cleaner can access each bin to get all the specific information about the bin like real time level, history of fill levels, etc.

3.2.3.4 Disabling/re-enabling the security system

As mentioned earlier the bin has a security system that is enabled when its placed in the city. When a cleaner has to clean a bin, the security system has to be disabled in order to do so. This function enables the cleaner to disable the security system at the start and re-enable it when the cleaning is finished.

3.2.3.5 User Application

This application also has the functionality to view all the bin locations and their fill levels through the given map. But the main function is to provide a route to the nearest available bin. This route is calculated by considering the GPS location of the user. The best route from the current location to the nearest available bin is provided in the map provided in the application.

3.2.4 Route Calculation

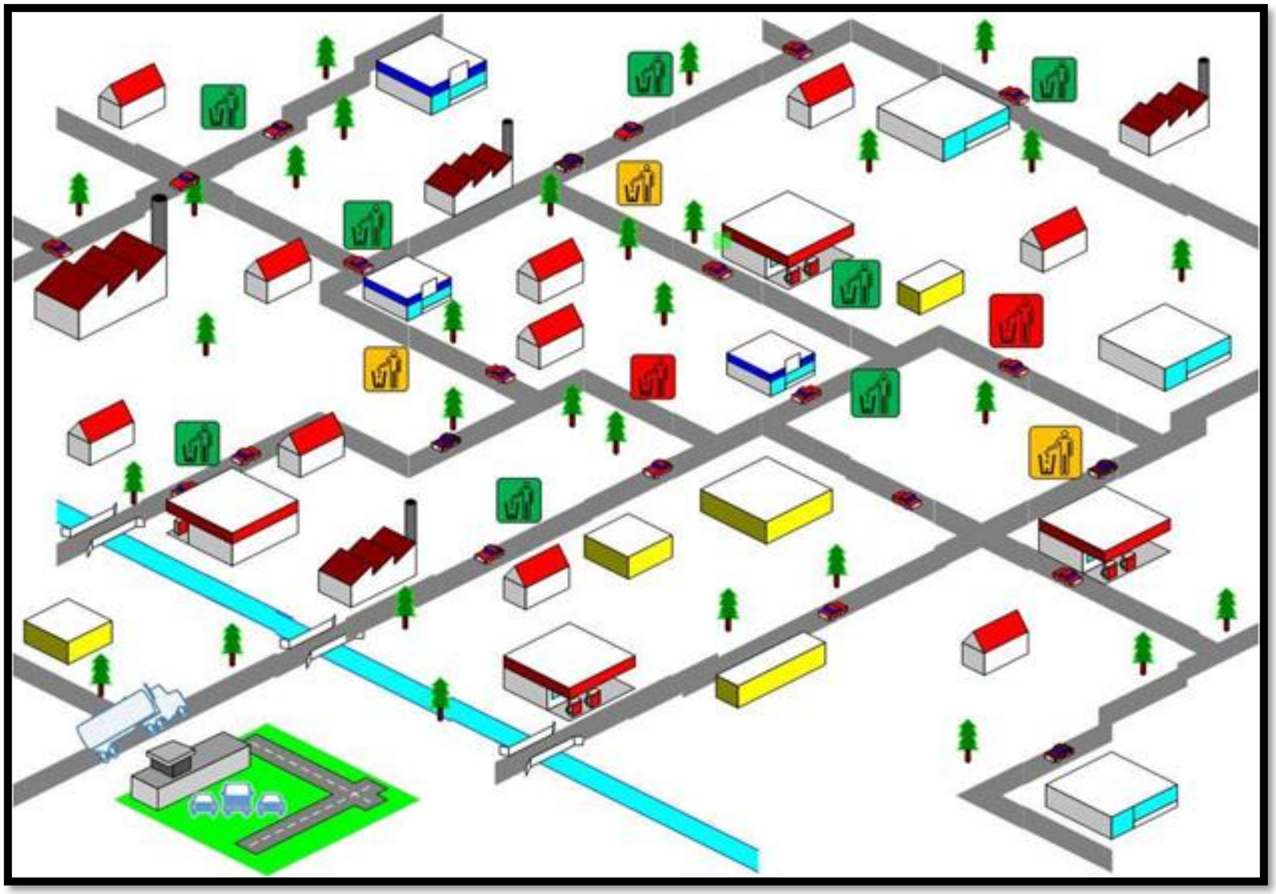

Figure 3.2.4.1: Sample Distribution of Bins

The above figure indicates a normal map of a city. Bins have been placed on usual locations. The available bins are colored green, bins that are going to be over a certain percentage in yellow, bins that are over 80% capacity in red.

The priority for calculating the route is given to the red bins. The yellow bins are added as waypoints. When a bin reaches 80% capacity the system checks which of the nearest bins are going to be filled in the future on a given period of time. These bins are marked in yellow based on our algorithm.

The algorithm :Basically the algorithm predicts which of the bins that are nearest to the 80% capacity bin is going to be filled in the next (x) hours. This is done by using the newton forward interpolation formula.

Interpolation Newton's Forward Difference Formula [17] [18].

Let $y = f(x)$ be a function which takes values $f(x_0), f(x_0 + h), f(x_0+2h)$, …, corresponding to various equi-spaced values of x with spacing h, say x_0, $x_0 + h$, $x_0 + 2h$, …. Suppose, we wish to evaluate the function $f(x)$ for a value $x_0 + ph$, where p is any real number, then for any real number p, we have the operator E such that $E^p f(x) = f(x + ph)$.

- $f(x_0 + ph) = E^p f(x_0)$
  $=(1+\Delta)^p f(x_0)$

  $$= \left[1 + p\Delta + \frac{p(p-1)}{2!}\Delta^2 + \frac{p(p-1)(p-2)}{3!}\Delta^3 + \ldots + \frac{p(p-1)(p-2)\ldots(p-n+1)}{n!}\Delta^n \right] f(x_0)$$

- $$f(x_0 + ph) = f(x_0) + p\Delta f(x_0) + \frac{p(p-1)}{2!}\Delta^2 f(x_0) + \frac{p(p-1)(p-2)}{3!}\Delta^3 f(x_0) + \ldots\ldots + \frac{p(p-1)(p-2)\ldots(p-n+1)}{n!}\Delta^n f(x_0) + \text{Error}$$

This is known as Newton's forward difference formula for interpolation, which gives the value of $f(x_0 + ph)$ in terms of $f(x_0)$ and its leading differences.

This formula is also known as Newton-Gregory forward difference interpolation formula. Here $p = (x-x_0)/h$. An alternate expression is

$$y_x = y_0 + p\Delta y_0 + \frac{p(p-1)}{2!}\Delta^2 y_0 + \frac{p(p-1)(p-2)}{3!}\Delta^3 y_0 + \ldots\ldots + \frac{p(p-1)(p-2)\ldots(p-n+1)}{n!}\Delta^n y_0 + \text{Error}$$

If we retain $(r + 1)$ terms, we obtain a polynomial of degree r agreeing with y_x at $x_0, x_1, \ldots, x_r$.

This formula is mainly used for interpolating the values of y near the beginning of a set of tabular values and for extrapolating values of y, a short distance backward from y_0 .

By using this algorithm, we are developing our own algorithm for each bin which describes how the bins are going to be filled. Our database is constantly updated by new capacity values. This data is taken as the past data to predict future fill levels.

Take look at the below example.

X(Time – 24hr)	8	9	10	11	12
Y (level – cm)	10	30	55	70	85

Find the filed level of bin at 1.00 pm using below data?

Solution

X	y = f(x)	Δ y	Δ^2 y	Δ^3 y	Δ^4 y
8	10				
9	30	20			
10	55	25	5		
11	70	15	-10	-15	
12	85	15	0	10	25

We have Newton's forward difference interpolation formula

$$y_x = y_0 + p\,\Delta y_0 + \frac{p(p-1)}{2!}\Delta^2 y_0 + \frac{p(p-1)(p-2)}{3!}\Delta^3 y_0 + \ldots\ldots\ldots + \frac{p(p-1)(p-2).....(p-n+1)}{n!}\Delta^n y_0$$

Here we have

$X_0 = 8$

$Y_0 = 10$

$\Delta y_0 = 20$

$\Delta^2 y_0 = 5$

$\Delta^3 y_0 = -15$

$\Delta^4 y_0 = 25$

Let y_{13} be the value of y when x = 13, then

$$p = \frac{x-x0}{h} = \frac{13-8}{1} = 5$$

$$f(13) = y_{13} = 10 + (5)(20) + \frac{5(5-1)}{2!}(5) + \frac{5(5-1)(5-2)}{3!}(-15) + \frac{5(5-1)(5-2)(5-3)}{4!}(25)$$

$$f(13) = 10 + 100 + 50 - 150 + 125 = 135$$

Then, we can say filled level of bin is 135 cm at 1.00 pm.

After predicting the bins using the above algorithm, the identified bins are marked as waypoint on the route that is being calculated. Then use google maps to calculate the route using the base station, 80% filled bin and the waypoints. Google maps use Dijkstra's algorithm to calculate the shortest path [19].

The path calculated is demonstrated in the below image.

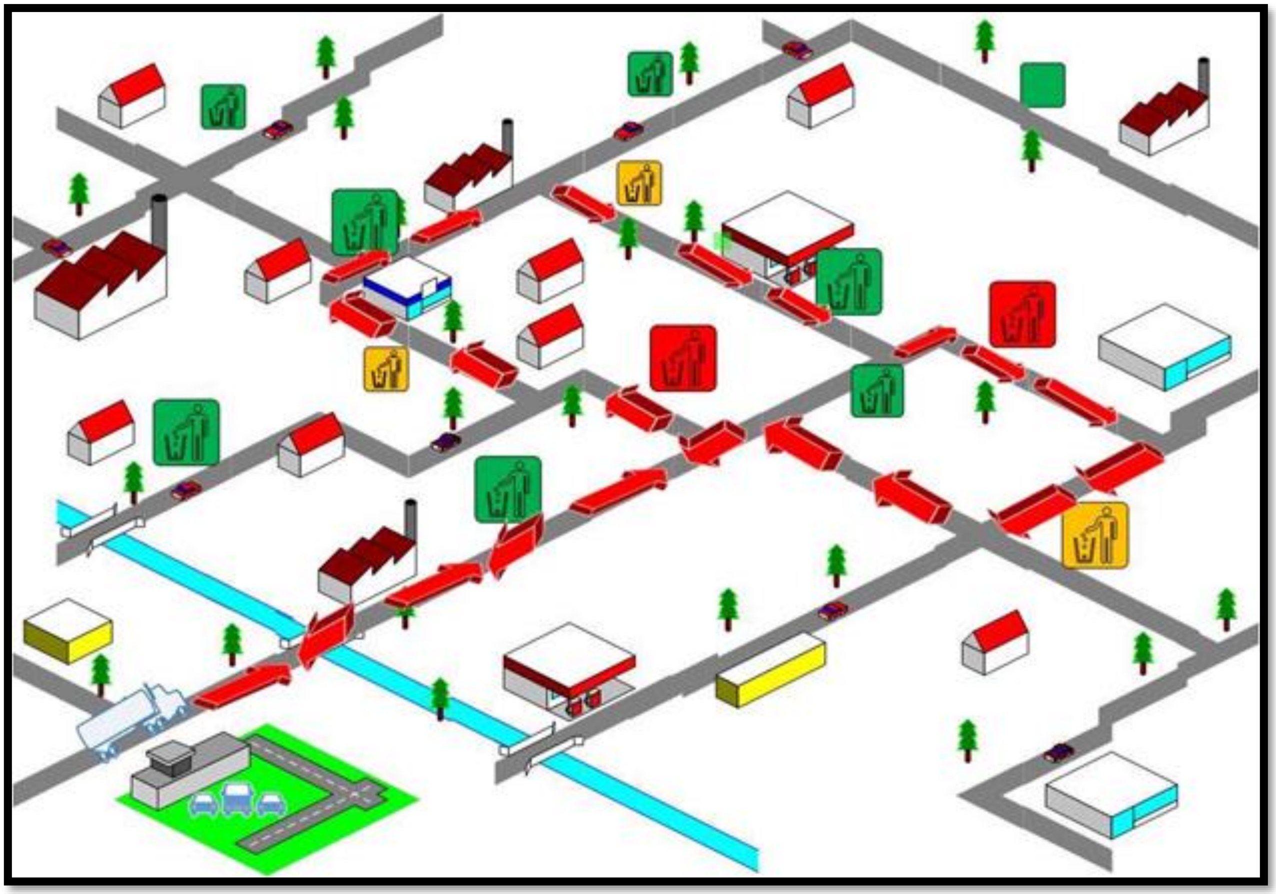

Figure 3.2.4.2: Calculated Shortest Path

3.3 Gantt chart

ID	Task Name	Start	Finish	Duration
1	Project Topic Selection	1/5/2017	2/3/2017	22d
2	Literature Reviews	1/20/2017	3/15/2017	39d
3	Project Pre-Assessment	2/13/2017	2/15/2017	3d
4	Project Charter Submission	3/3/2017	3/6/2017	2d
5	Project Proposal Report	3/10/2017	3/17/2017	6d
6	Project Proposal Presentation	3/20/2017	3/24/2017	5d
7	Design Basic Structure of Trash Can	3/24/2017	4/3/2017	7d
8	Gather information & Select Suitable Sensor	3/24/2017	4/6/2017	10d
9	Buy Required Material	4/4/2017	4/21/2017	14d
10	Develop Basic Structure of Trash Can	4/10/2017	4/25/2017	12d
11	Develop Workforce Android App	3/31/2017	9/14/2017	120d
12	Develop End User Android App	3/31/2017	9/14/2017	120d
13	Web Server Setup	4/10/2017	4/21/2017	10d
14	SRS Document	4/11/2017	4/25/2017	11d
15	Configure Development Board	4/19/2017	4/27/2017	7d
16	Individual Sensors & WiFi Module Configure	4/28/2017	5/26/2017	21d
17	Configure Local Database	5/10/2017	5/18/2017	7d
18	Upload Sensor's Data to Server	5/26/2017	6/5/2017	7d
19	Design & Develop Web Application	5/1/2017	9/15/2017	100d
20	Develop Algorithm For Route Calculation	6/20/2017	8/14/2017	40d
21	Calculate shortest route to available bins	6/9/2017	8/3/2017	40d
22	Progress presentation (50%)	7/28/2017	8/1/2017	3d
23	Configure Hand Gesture System	8/1/2017	8/9/2017	7d
24	Develop SMS Notification System	8/10/2017	8/23/2017	10d
25	Develop Security Management	8/15/2017	9/11/2017	20d
26	Progress Presentation (90%)	9/5/2017	9/13/2017	7d
27	Hosting Web Site on VPS	9/14/2017	9/22/2017	7d
28	Migrate Local Database to VPS	9/14/2017	9/22/2017	7d
29	Integrate all sub Systems	9/22/2017	10/31/2017	28d
30	Testing and Fixing bugs	3/31/2017	10/31/2017	153d
31	Final Report (draft)	10/4/2017	10/4/2017	1d
32	Final Report (draft) – Feedback Submission	10/20/2017	10/20/2017	1d
33	Final Presentation	11/20/2017	11/24/2017	5d
34	Viva	11/20/2017	11/24/2017	5d
35	Research Paper	8/18/2017	8/18/2017	1d

Figure 3.3.1: Gantt chart

3.4 Work Breakdown Structure

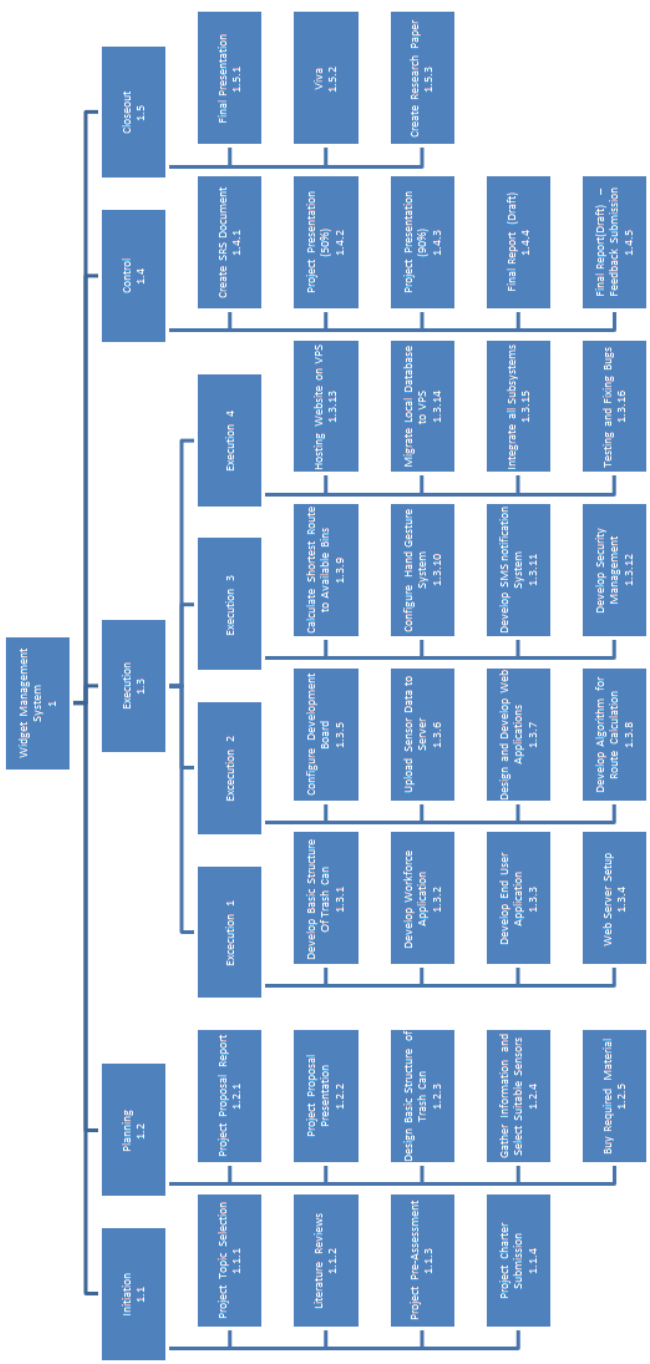

Figure 3.4.1: Work Breakdown Structure

4.0 Description of Personal and Facilities

Member	Task Area	Specific Task
G.S.B. Dabarera	Garbage Bin Setup and Management	• Use sensor data to calculate garbage level and upload data to server via Wi-Fi module. • If the garbage level is at 100%, the bin door gets locked. • Bin door open when hand in proximity and closes again when in proximity or 30s after opening
	Security management system	• Sound an alarm if the garbage bin is moved without disabling the alarm system manually through the cleaner application.
R.K.R. Ranaweera	Web Server Setup	• Setting up initial configurations on VPS to setup web server • Hosting website • Installing and setting up Databases
	Web Services	• Analytical system to analyze past, present fill levels, workforce information to generate reports. • Monitoring system using real time data to track bin levels and cleaner progress.
P.G.D.M. Perera	Workforce Application	• Show and navigate the most efficient cleaning route provided by the server. • Views fill levels of all bins. • Function to disable and enable security system on the bin for cleaning purposes.
	Route Calculation	• Get previous fill levels of bins and calculate which bin gets filled at times of the day.

		• Calculate route to the 80% full bin adding waypoints to the route from nearby bins that are 50% full. • Waypoint bins are selected considering the analysis of previous fill levels
P.A.V.D.R. Panangala	SMS Notification/Feedback S ystem	• Sending SMS to cleaners and or administrators when a bin gets 80% and 100% full. • Develop web interface to get and send feedbacks • Adding SMS notification system settings page to the website. • Feedback system for users to place their complaints. • A system for users to request bins on specific locations.
	End User Application	• Calculate and navigate the path to the nearest available bin using the location the client is at. • Show a map of all the bins and their fill levels

Table 4.0.1: Description of Personal and Facilities

5.0 Budget

Description	No. Of Items	Cost per Item	Total
Garbage Container	1	Rs.2000.00	Rs.2000.00
Raspberry pi zero	1	Rs.500.00	Rs.500.00
Motion Sensor	2	Rs.250.00	Rs.500.00
Infrared Sensor	4	Rs.180.00	Rs.720.00
Server(VPS)	1	Rs.600.00	Rs.600.00
Servo motor	1	Rs.350.00	Rs.350.00
Wi-Fi Module	1	Rs.270.00	Rs.270.00
Sub total			Rs.4940.00
Additional Expenses			Rs.2000.00
Grand Total			Rs.6940.00

Table 5.0.1: Budget

6.0 Reference list

[1] S. Longhi *et al*., "Solid Waste Management Architecture Using Wireless Sensor Network Technology," *2012 5th International Conference on New Technologies, Mobility and Security (NTMS)*, Istanbul, 2012, pp. 1-5.

[2] H. Chourabi, T. Nam, S. Walker and others,. "Understanding Smart Cities: An Integrative Framework," In Proc. of the *45th Hawaii International Conference on System Sciences* (2012), pp. 2289–2295.

[3] United States Environmental Protection Agency. Roadmap for Next Generation Monitoring. March 2013.

[4] N. Kumar, C. Swamy, and K. Nagadarshini, "Efficient Garbage Disposal Management in Metropolitan Cities Using VANETs", Journal of Clean Energy Technologies, Vol. 2, No. 3, July 2014, pp. 258-262.

[5] T. Anagnostopoulos, A. Zaslavsky, and A. Medvedev, "Robust Waste Collection exploiting Cost Efficiency of IoT potentiality in Smart Cities", *IEEE 1st International Conference on Recent Advances in Internet of Things (RIoT)*, [Accepted on February 4, 2015].

[6] W. Xie and Y. Chen, "A Cache Management Scheme for Hiding Garbage Collection Latency in Flash-Based Solid State Drives," 2015 IEEE International Conference on Cluster Computing, Chicago, IL, 2015, pp. 486-487.

[7] Narayan Sharma, Nirman Singha, Tanmoy Dutta, Smart Bin Implementation for Smart Cities, International Journal of Scientific & Engineering Research, Volume 6, Issue 9, September-2015, pp. 787- 791.

[8] T. Anagnostopoulos, A. Zaslavsky, A. Medvedev, S. Khoruzhnikov, "Top-k Query based Dynamic Scheduling for IoT-enabled Smart City Waste Collection," In Proc. of

the 16th IEEE International Conference on Mobile Data Management (MDM 2015), Pittsburgh, US.

[9] Y. Chen, J. Nakazawa, T. Yonezawa, T. Kawsaki and H. Tokuda, "Cruisers: A Public Automotive Sensing Platform for Smart Cities," *2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS)*, Nara, 2016, pp. 767-768.

[10] A. S. Bharadwaj, R. Rego and A. Chowdhury, "IoT based solid waste management system: A conceptual approach with an architectural solution as a smart city application," *2016 IEEE Annual India Conference (INDICON)*, Bangalore, 2016, pp. 1-6.

[11] R. Fujdiak, P. Masek, P. Mlynek, J. Misurec and E. Olshannikova, "Using genetic algorithm for advanced municipal waste collection in Smart City," *2016 10th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, Prague, 2016, pp. 1-6.

[12] A. Borozdukhin, O. Dolinina and V. Pechenkin, "Approach to the garbage collection in the “Smart Clean City” project," 2016 4th IEEE International Colloquium on Information Science and Technology (CiSt), Tangier, 2016, pp. 918-922.

[13] J. Joshi *et al*., "Cloud computing based smart garbage monitoring system," *2016 3rd International Conference on Electronic Design (ICED)*, Phuket, 2016, pp. 70-75.

[14] N. S. Kumar, B. Vuayalakshmi, R. J. Prarthana and A. Shankar, "IOT based smart garbage alert system using Arduino UNO," *2016 IEEE Region 10 Conference (TENCON)*, Singapore, 2016, pp. 1028-1034.

[15] Team, Denizon. "Top 10 Disadvantages of Spreadsheets." Denizon. Denizon, 11 June 2016. [Online]. Available: https://www.denizon.com/spreadsheets/top-10-disadvantages-of-spreadsheets/. [Accessed: 26 Mar. 2017]

[16] What’s the difference between RF options (wifi, xbee, NRF24L01) 2013. [Online]. Available: http://arduino.stackexchange.com/questions/3270/whats-the-difference-between-rf-options-wifi-xbee-nrf24l01/. [Accessed: 26-Mar-2017]

[17] Newton's Forward Difference Formula 2012. [Online]. Available: https://mat.iitm.ac.in/home/sryedida/public_html/caimna/interpolation/nfdf.html/. [Accessed: 22-Mar-2017]

[18] "Newton's Forward Difference Formula." Newton's Forward Difference Formula -- from Wolfram MathWorld. [Online]. Available: http://mathworld.wolfram.com/NewtonsForwardDifferenceFormula.html/.[Accessed: 26-Mar-2017]

[19] "Waypoints in directions | Google Maps JavaScript API | Google Developers." Google Developers. [Online]. Available: https://developers.google.com/maps/documentation/javascript/examples/directions-waypoints/. [Accessed: 15-Mar-2017]

[20] "Raspberry Pi Zero." Raspberry Pi. [Online]. Available: https://www.raspberrypi.org/products/pi-zero/. [Accessed: 25-Jan-2017]

[21] "Raspberry Pi 3." Raspberry Pi. [Online]. Available: https://www.raspberrypi.org/products/pi-3/. [Accessed: 25-Jan-2017]

[22] "Arduino - ArduinoBoardUno." Arduino - ArduinoBoardUno. [Online]. Available: https://www.arduino.cc/en/main/arduinoBoardUno/. [Accessed: 25-Jan-2017]

7.0 Appendices

Raspberry pi zero specification:

Type	Model A	Zero
Memory (SDRAM)	256 MB (shared with GPU)	512 MB (shared with GPU)
USB 2.0 ports	1 (direct from BCM2835 chip)	1 Micro-USB (direct from BCM2835 chip)
Video input	15-pin MIPI camera interface (CSI) connector, used with the Raspberry Pi camera or Raspberry Pi NoIR camera	None

Table 7.0.1: Raspberry pi zero specification

Python:

Raspberry Pi Python Code Library

- Adafruit_ADS1x15
- Adafruit_ADXL345
- Adafruit_BMP085
- Adafruit_CharLCD
- Adafruit_DHT_Driver

Server Specification:

Server OS	**Type**	**vCPUs**	**Memory (GiB)**	**Instance Storage (GB)**	**EBS-Optimized Available**	**IpV6 Support**	**SSD size**
Ubuntu Server 16.04 LTS (HVM)	General purpose	t2.micro Free tier eligible	1	1	EBS only	Yes	30GB

Table 7.0.2: Server Specification