SMART WASTE MONITORING

Report submitted for the partial fulfillment of the requirements for the degree of
Bachelor of Technology in
Information Technology

Submitted by:

BitanBhattacharya(IT/2014/078)

HariharanVenkitakrishnan(IT/2014/080)

Vinit Kr. Singh(IT/2014/086)

Under the Guidance of Mrs. Moumita Deb

RCC Institute of Information Technology
Canal South Road, Beliaghata, Kolkata – 700 015
[Affiliated to West Bengal University of Technology]

Acknowledgement

We would like to express our sincere gratitude to Mrs. MoumitaDeb(Assistant Professor) of the department of Information Technology, whose role as project guide was invaluable for the project. We are extremely thankful for the keen interest he / she took in advising us, for the books and reference materials provided for the moral support extended to us.

Last but not the least we convey our gratitude to all the teachers for providing us the technical skill that will always remain as our asset and to all non-teaching staff for the gracious hospitality they offered us.

Place: RCCIIT, Kolkata

Date:

………………………………

………………………………

………………………………

Department of Information Technology
RCCIIT, Beliaghata,
Kolkata – 700 015,
West Bengal, India

Approval

This is to certify that the project report entitled "Smart Waste Monitoring" prepared under my supervision by BitanBhattacharya(IT/2014/078), HariharanVenkitakrishnan(IT/2014/080) and Vinit Kr. Singh(IT/2014/086), be accepted in partial fulfillment for the degree of Bachelor of Technology in Information Technology.It is to be understood that by this approval, the undersigned does not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn thereof, but approves the report only for the purpose for which it has been submitted.

……………………………………….. … ………………………………………

Name & Designation of the HOD Name & Designation of Internal Guide

CERTIFICATE of ACCEPTANCE

The report of the Project titled “Smart Waste Monitoring ” submitted by
Bitan Bhattacharya(IT/2014/078),Hariharan Venkitakrishnan(IT/2014/080) and Vinit Kr. Singh(IT/2014/086) of (B.Tech. (IT) 8th Semester of 2018), is hereby recommended to be accepted for the partial fulfillment of the
requirements for B Tech (IT) degree in Maulana Abdul Kalam Azad University of Technology”.

Name of the Examiner **Signature with Date**

1. ……………………………………. …………………………….
2. …………………………………….. ……………………………..
3. ……………………………………… ………………………………
4. ………………………………………. ………………………………

Contents	***Page Numbers***
1. Introduction	***6***
2. Problem Definition	***7***
3. Literature Survey	***8***
4. SRS (Software Requirement Specification)	***9-10***
5. Planning	***11***
6. Design	***12-14***
7. Results and Discussion	***15-27***
8. Conclusion and Future Scope	***28***
9. References / Bibliography	***29***
10. List of Tables	***30***
11. List of Figures	***31***

Introduction

The ultimate need of the hour for a developing nation is the key for “Smart City”. The influential ecological factors that poses to be a threat to this may include: hazardous pollution and its subsequent effects on health of humanity, alarming global warming and depletion of ozone layer etc. Mostly Environmental pollution may be owing to the Municipal Solid Leftovers (MSL) [1]. A Proper maintenance becomes mandatory for an efficient and effective removal of the generated Municipal Solid Leftover [2]. It is perceived that often the waste space gets too much occupied due to irregular removal of garbage occupancy in the dustbin.. This exposition proposes an e-monitoring system that puts forth an e-monitoring system to eradicate or minimize the garbage disposal problem using IoT Technology. The Internet of Things (IoT) is a recent communication paradigm that envisions near future, in which the objects of everyday life will be equipped with microcontrollers, transceivers for digital communication, and suitable protocol stacks that will make them able to communicate with one another and with the users, becoming an integral part of the Internet. This design designates a technique in which the garbage level could be checked at regular intervals which would prevent the undesirable overflow of the bin. This system monitors the garbage bins and informs about the level of garbage collected in the garbage bins. For this, the system uses ultrasonic sensors placed over the bins to detect the garbage level and compare it with the garbage bins’ depth. The system makes use of Arduino family microcontroller and a Wi-Fi modem for sending data to webserver that monitors the garbage levels. Programming in the Arduino UNO is done in such a way that once a particular level of filling is sensed information message is sent requesting a clean-up.

Problem Definition

This project combats the problem of overflowing solid waste bins which pollute the surroundings. The level of garbage present in any bin is determined by the ultrasonic distance measuring sensor. When the garbage level in any garbage bin exceeds a pre-defined level, then the microcontroller send an alert message to the e-monitoring station, and, the workstation then assigns the nearest garbage collecting truck to collect the garbage from such bins, which have sent an alert message. It informs when the container is at full capacity and when it needs to be emptied, thus allowing the sanitation specialists to work more efficiently and cut unnecessary costs.

Literature Survey

In [3] an expensive version of the Arduino board (MKR1000) is used which would not be scalable. In [9], the authors have used Arduino UNO along with a wifi module, which would use much battery power and batteries will have to be replaced frequently. Also, theyhad used RFID to identify the location of the dustbins which will increase the expenses of this project unnecessarily. This could have been done by simply giving an Identification number to each microcontroller in the code. Similarly, in [1] they also used RFID for identification of bins which is an expensive addition to the project. Arduino increases power consumption and we need a separate Wi-Fi module for sending the collected data. In order to overcome this, this venture is using NodeMCU which consumes less power as compared to Arduino and it also has a built in Wi-Fi module which allows us to send data over wifi using the same component. Again in [1] GSM modules are used which unnecessarily use a chunk of battery power even when not in use ,so GSM has been done away with in this venture. NodeMCU is that works on 3.3V logic but Ultrasonic sensors require 4.5 - 5V,even though they can function at 3V sometimes, it makes the system unreliable, but we can overcome this problem using a Bi-directional [7] Logic Level Converter. In [4] too,the authors propose to use GSM module to send SMS to workstation when the bin is full. This not only makes the system in-efficient to analyse but also makes it power in-efficient. In [5] the authors propose to use INTEX IE305WC which is a 16MP camera to continuously take pictures of the garbage can, and also a weight sensor to check if the garbage can has crossed a predefined limit. These features complicate the analysis and is also very resource heavy. The system proposed in our design tackles these issues by using a simple microcontroller CP2102 which is combined with a wi-fi module ESP8266 , their power consumption can be reduced dramatically by using “deep sleep” [8] mode ,this will help us reduce not only cost but also reduce power consumption which is very important as far as this project is concerned because creating a smart waste management would be of no use if it is very expensive and requires human intervention very frequently.

SRS(Software Requirement Specification)

Functional Requirements:

- Taking sensor reading from the Sensor Circuit
- Pushing the data to a MySQL database.
- Retrieving information from database for Calculation of the nearest garbage bin which fulfils the condition for garbage collection, example : Collect garbage from bins whose level is over 50% of bin.
- A client side script to get Garbage collection truck's live Geolocation.

Non-functional Requirement:

- The project requires a user interface for monitoring and manually intervening (if required) in the efficient and timely collection of garbage from the selected Garbage bins.
- Another user interface where Garbage collection truck's driver(s) can see their next stop location.

Hardware Requirements:

- NodeMCU (ESP8266)
- 4* AA Battery
- Ultrasonic Sensors
- Logic Level Converter (Voltage Divider)

Minimum Computer Requirements:

- 512 MB RAM
- Core 2 Duo processor
- Windows XP SP3 or later version.

Software Requirements:

- Text Editor
- XAMPP local server
- Arduino IDE
- MySQLi Database

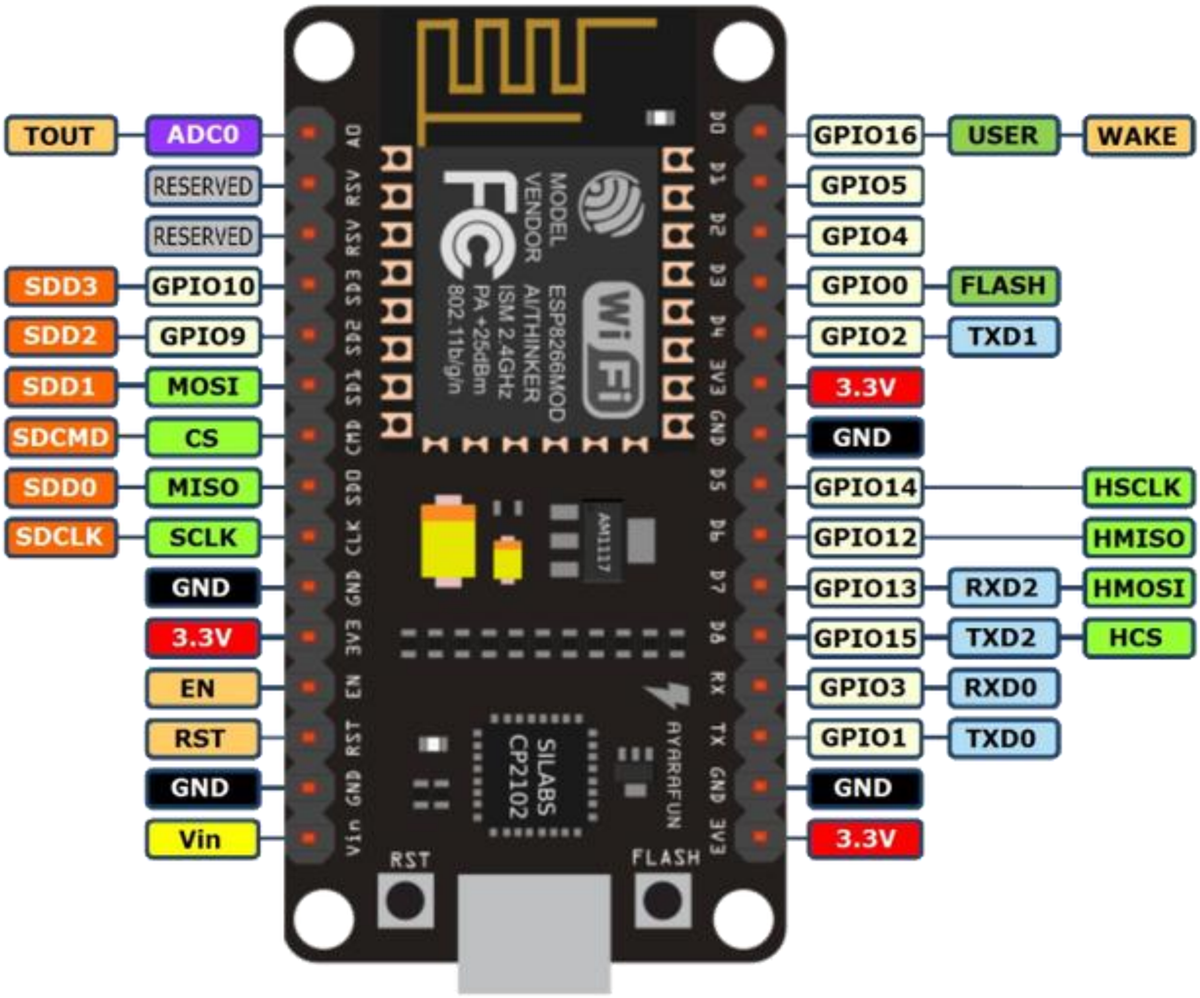
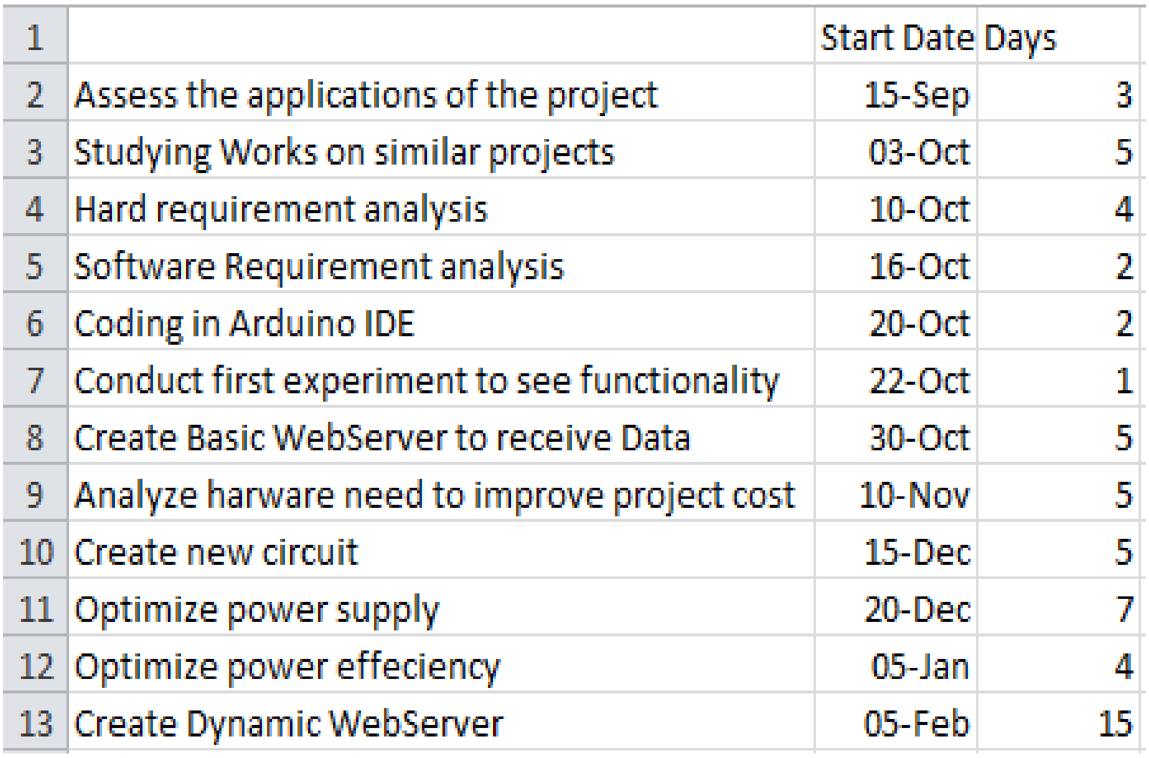


Fig1. Node MCU Pin Configuration

Planning



1		Start Date	Days
2	Assess the applications of the project	15-Sep	3
3	Studying Works on similar projects	03-Oct	5
4	Hard requirement analysis	10-Oct	4
5	Software Requirement analysis	16-Oct	2
6	Coding in Arduino IDE	20-Oct	2
7	Conduct first experiment to see functionality	22-Oct	1
8	Create Basic WebServer to receive Data	30-Oct	5
9	Analyze harware need to improve project cost	10-Nov	5
10	Create new circuit	15-Dec	5
11	Optimize power supply	20-Dec	7
12	Optimize power effeciency	05-Jan	4
13	Create Dynamic WebServer	05-Feb	15

Table 1 :- Different Stages of Work

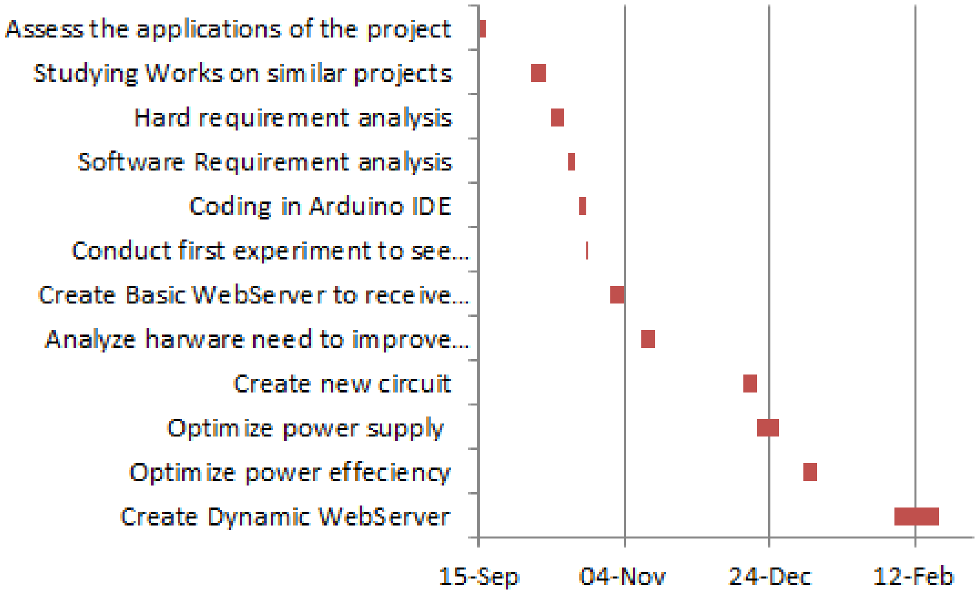


Fig2. Gnatt Chart

Design

Sensing: Ultrasonic sensors provide information on an absolute position of an object. This helps in early detection of events and enables to take the necessary steps.

Monitoring: Along with continuous monitoring of the sensors, the information should be reported in real time to the appropriate central server.

Control: The monitored data is analyzed in real time and the optimum control information is determined and transmitted over networks.

Storage & Backup: Rapid, flexible and accurate responses is desired since the analysis and control information is done on real time.

In accordance with the requirements, the intelligence of our system is provided by the IoT devices. Ultrasonic sensors are used in smart bins. The IoT devices send these data over the wireless networks to the processing server,and this data is used to generate optimum schedules forwaste collection.

Smart bins*:* The waste bins are attached with sensors that will sense the filling level of the bin. These bins are attached with sensors which make them 'smart'. Ultrasonic sensors act like a radar system and track the amount of garbage in the bin. These sensors send information and alert the processing system when a threshold limit is reached, which enables the processing system to generate an optimized schedule according to which the smart trucks will collect waste from these bins.

Collection trucks*:* Trucks are used for collection of waste from the smart bins and transporting them to the local collection points for further processing of the waste. The driver of the truck is provided with a smart device which will provide the driver with real time routes and optimized schedules according to which the waste collection can be done. The processing system provides the schedules based on the data generated by the smart bins.

The data generated by the IoT is sent from the device to the central processing system via a high speed wireless transmission medium. We are using Wi-Fi in this case.

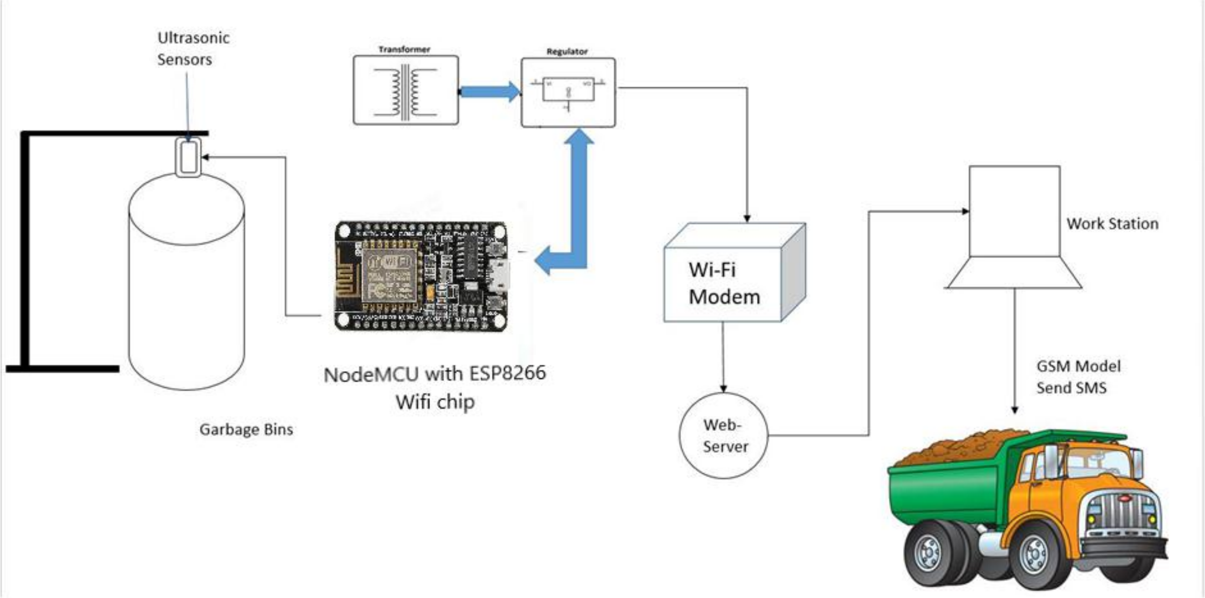


Fig3. Block Diagram

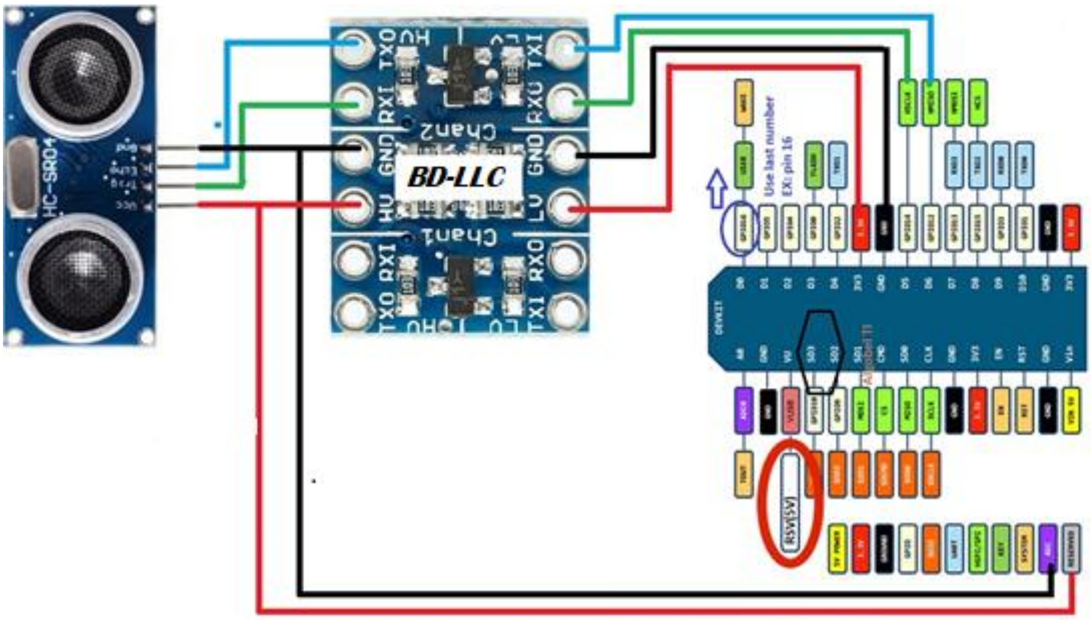


Fig4. Circuit Diagram

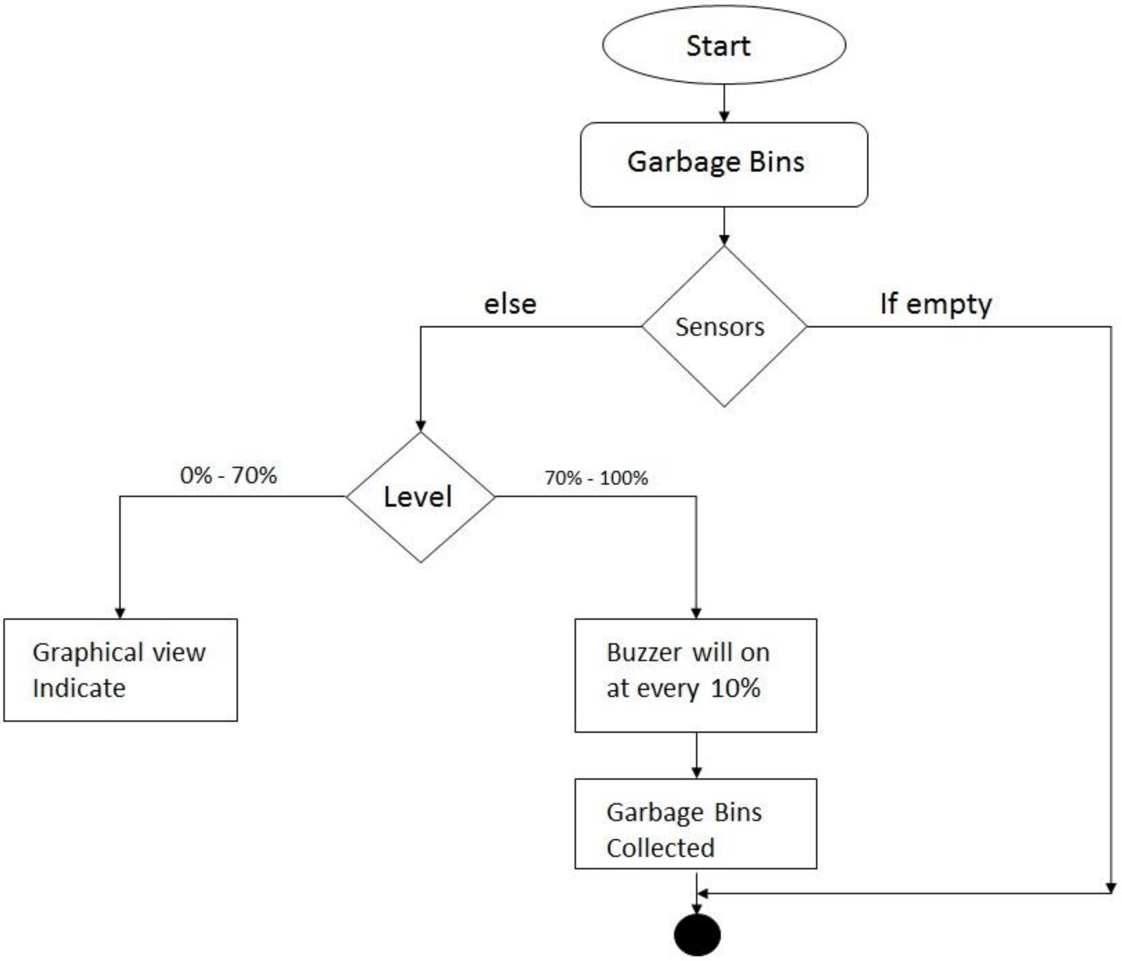
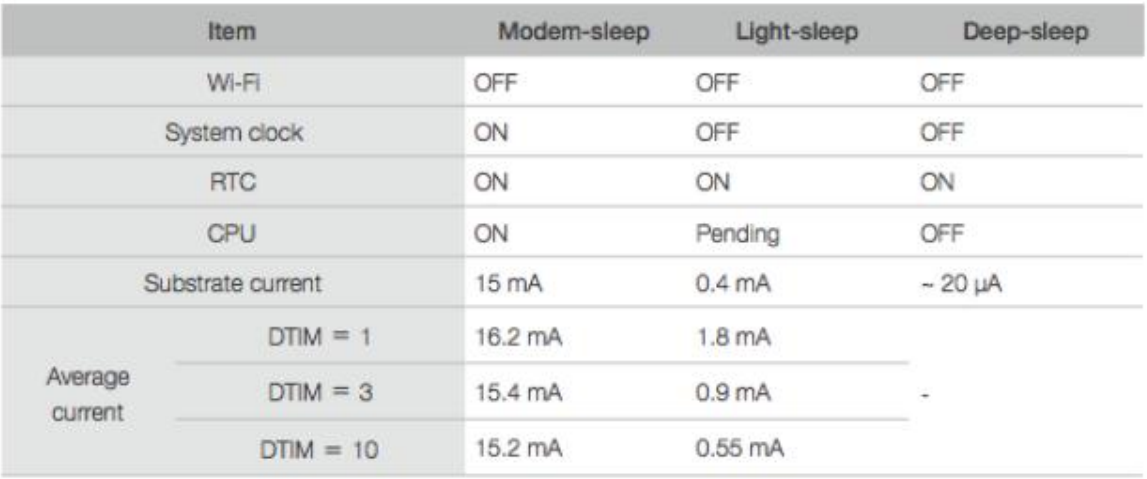


Fig5. Flow Chart

Table 2 – Deep Sleep Power Saving



Item		Modem-sleep	Light-sleep	Deep-sleep
Wi-Fi		OFF	OFF	OFF
System clock		ON	OFF	OFF
RTC		ON	ON	ON
CPU		ON	Pending	OFF
Substrate current		15 mA	0.4 mA	~ 20 µA
Average current	DTIM = 1	16.2 mA	1.8 mA	-
	DTIM = 3	15.4 mA	0.9 mA	
	DTIM = 10	15.2 mA	0.55 mA	

Results and Discussions

Code to take sensor reading from hardware: (NodeMcuUltrasonic.ino)

```
#include <ESP8266WiFi.h>

#include <ESP8266HTTPClient.h>


const char* ssid = "ShazAM";

// the name of the wifi


const char* password = "dhanyawad";

// the password of the wifi


#define TRIGGER_PIN  14 // D5

#define ECHO_PIN     12 //D6


long randNumber;

int nodeID = 1;

float depth = 100;

float filled;

void setup()

{

  Serial.begin (9600);

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED)

  {

    delay(1000);
  }
  pinMode(TRIGGER_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
  randomSeed(analogRead(0));
}

void loop()
{
  if (WiFi.status() == WL_CONNECTED)
  {
    double duration;
    float distance;
    HTTPClient http;    //Declare an object of class HTTPClient

    digitalWrite(TRIGGER_PIN, LOW);  // Get Start
    delayMicroseconds(2); // stable the line
    digitalWrite(TRIGGER_PIN, HIGH); // sending 10 us pulse
    delayMicroseconds(10); // delay

    digitalWrite(TRIGGER_PIN, LOW); // after sending pulse wating to receive signals

    duration = pulseIn(ECHO_PIN, HIGH); // calculating time

    distance = (duration/2) / 29.1; // single path
    filled = ((depth - distance)/depth ) * 100;
    String data;
    data = "dist=" + String(distance,DEC) +"&" + "id=" + String(nodeID) + "&" + String(filled);
```

```
   String link ="http://192.168.31.115/IOT/write_DB.php?"+data;
   //Serial.println(link);     //Serial.print(distance);  //Serial.println();  //Serial.println(" cm");
   http.begin(link); //Specify request destination
   int httpCode = http.GET();    //Send the request
   http.end(); //Close connection
   randNumber = random(240,360);
   ESP.deepSleep(1e6 * randNumber); // sleep for random number of seconds between 240 and 360
  }
  else
  {
   delay(5000);
  }
}
```

Code to take input from NodeMCU Ultrasonic and pushing the data into the database: (write_DB.php)

```
<?php

   // Capture Data

   $dist = $_GET['dist'];

   $ID = $_GET['id'];

   $percent = $_GET['filled'];

   // Prepare variables for database connection

   $dbusername = "garbnode";  // enter database username,

   $dbpassword = "bitu";  // enter database password,

   $server = "localhost"; // IMPORTANT: if you are using XAMPP enter "localhost", but if
you have an online website enter its address

   // Connect to your database

   $dbconnect = mysqli_connect($server, $dbusername, $dbpassword,"garbdb"); //garbdb is
the database

   $dbselect = mysqli_select_db($dbconnect,"garbdb");

   $sql =  "UPDATE garbdb.sensor SET nodeValue='$dist' , Percent_Filled = '$percent'
WHERE nodeID= '$ID'" ;

   mysqli_query($dbconnect,$sql); // Execute SQL statement

?>
```

Code for routing algorithm:
(Calculator.php)

```
<?php
  include("connect.php");
  $dbconnect_calc = Connection();
  if (empty($_GET)) {
   $latd = "22.6293778"; //Using POST we can make this dynamic by receiving driver location from geo_pos.html
  $longd = "88.4225929";
  }
  else{
   $latd = $_GET['latd'];
   $longd = $_GET['longd'];    }


  $lat2; //To do-> Change code to Get this via SQL
  $long2; //This too


function GetDrivingDistance($lat1, $lat2, $long1, $long2)
{
  $url =
"https://maps.googleapis.com/maps/api/distancematrix/json?origins=".$lat1.",".$long1."&destinations=".$lat2.",".$long2."&mode=driving&language=en-US";
  $ch = curl_init();
  curl_setopt($ch, CURLOPT_URL, $url);
  curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
  curl_setopt($ch, CURLOPT_PROXYPORT, 3128);
  curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0);
  curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0);
  $response = curl_exec($ch);
  curl_close($ch);
  $response_a = json_decode($response, true);
  $dist = $response_a['rows'][0]['elements'][0]['distance']['text'];
  $time = $response_a['rows'][0]['elements'][0]['duration']['text'];

  return array('distance' => $dist, 'time' => $time);
}

      $calc_query = " SELECT * FROM `sensor` \n"."WHERE `sensor`.`Percent_Filled` > 20";
      $calc_result = mysqli_query($dbconnect_calc,$calc_query);
      $drivedist = array();
      $Loc = array();
      $i=0;
     if($calc_result!==FALSE){
      while($row = mysqli_fetch_array($calc_result)) {
           $dist = GetDrivingDistance($latd, $row["Latitude"], $longd, $row["Longitude"]);
           // echo 'Distance to  '.$row["Location"].'  <b> '.$dist['distance'].'</b><br>Travel time duration: <b>'.$dist['time'].'</b>'."\n";
           $drivedist[$i] = $dist['distance'];
           $DD = $drivedist[$i];
           $ID = $row["nodeID"];
           $ETA = $dist['time'];
           $sql = "UPDATE garbdb.sensor SET `sensor`.`Distance` = '$DD' , `sensor`.`ETA` = '$ETA' WHERE `sensor`.`nodeID` = '$ID' ";
           mysqli_query($dbconnect_calc,$sql);
           $Loc[$i] = $row["Location"];
           $i = $i+1;
         }
         mysqli_free_result($calc_result);

       }
       $arrlength = count($drivedist);
       $min = $drivedist[0];
       $count=0;
       for($x = 0; $x < $arrlength; $x++)
       {
        if( $drivedist[$x] < $min)
        {
           $min = $drivedist[$x];
           $count = $x;
        }
       }

       $next_location = $Loc[$count];

       $lat_query= "SELECT Latitude FROM `sensor` WHERE Location = '$next_location'";
       $lONG_query= "SELECT Longitude FROM `sensor` WHERE Location = '$next_location'";
       $lat_result = mysqli_query($dbconnect_calc,$lat_query);
       $long_result= mysqli_query($dbconnect_calc,$lONG_query);
       $dest_lat = mysqli_fetch_array($lat_result);
       $dest_long = mysqli_fetch_array($long_result);
       $destination_lat = $dest_lat['Latitude'];
       $destination_long = $dest_long['Longitude'];
       echo "<br>\n Your Next STOP is at:-->  ".$next_location." at ".$destination_lat." N ".$destination_long." E";
       mysqli_close($dbconnect_calc);
        $url_r = "https://maps.google.com/maps?saddr=".$latd.",".$longd."&daddr=".$destination_lat.",".$destination_long;
        header("Location:".$url_r);
```

```
?>
```

Code for taking driver location and finding nodes nearest to its location: (Geo_pos.php)

```
<!DOCTYPE html>
<html>
<body>

<p>Click the button to get your coordinates.</p>

<button onclick="getLocation();">Get Location</button>

<p id="demo"></p>

<script>
var x = document.getElementById("demo");

function getLocation() {
   if (navigator.geolocation) {
       navigator.geolocation.getCurrentPosition(redirectToPosition);
   } else {
       x.innerHTML = "Geolocation is not supported by this browser.";
   }
}

function redirectToPosition(position) {

window.location='calculator.php?latd='+position.coords.latitude+'&longd='+position.coords.lon
gitude;
   x.innerHTML="Latitude: " + position.coords.latitude +
   "<br>Longitude: " + position.coords.longitude;
}
</script>

</body>
</html>
```

Code for the home page:
(index.php)

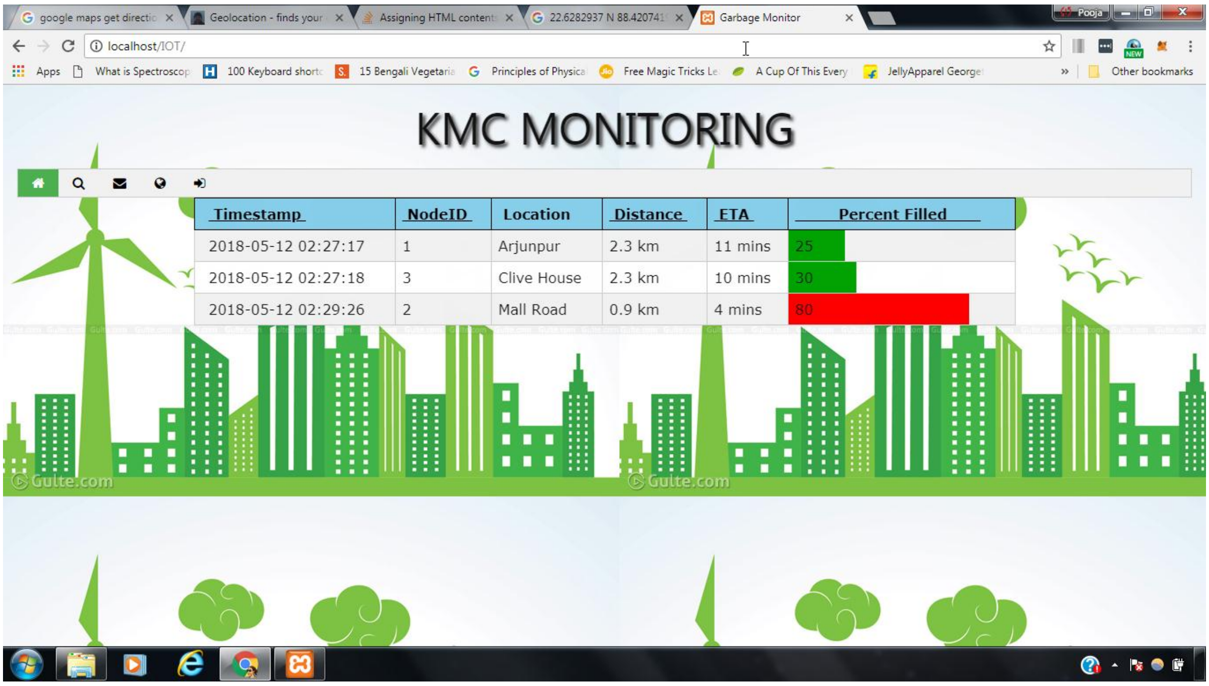
```
<?php
  include("connect.php");
  $dbconnect = Connection();
  $query=";
  $result=";
  if (empty($_GET))
  {
        $query = "SELECT * FROM `sensor`  \n". "ORDER BY `sensor`.`Distance`  DESC
";
   //use ->DESC LIMIT 0,2 if you want to display only 2 most recent result
         }
        elseif(!empty($_GET)) {
        $query = "SELECT * FROM `sensor` \n";
        if ($_GET['sort'] == 'nodeID')
        {
        $query .= " ORDER BY nodeID";
        }
        elseif ($_GET['sort'] == 'Percent_Filled')
        {
        $query .= "ORDER BY Percent_Filled";
        }
 elseif ($_GET['sort'] == 'Time')
 {
   $query .= "ORDER BY Time";
 }
 elseif($_GET['sort'] == 'ETA')
 {
  $query .= "ORDER BY ETA";
 }
 elseif($_GET['sort'] == 'Distance')
 {
  $query .= "ORDER BY Distance";
 }
        }

 $result = mysqli_query($dbconnect,$query); // Execute SQL statement
 $result_c = $result;
 ?>
<html>
  <head>
    <title>Garbage Monitor</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
```

```
      <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
      <style>
table, th {
    border: 1px solid black;
    border-collapse: collapse;
    text-align: center;
}

body  {
    background-image: url("clean.jpg");
    background-size: auto;
    /*background: #2c2b2b;*/
    color: #131313;
}
tr{
            background-color: #f8f8f8;
    opacity: .999;
}
td {
    border: 1px solid #ccc;
    text-align: center;
}
td {
    position: relative;
    text-align: center;
}
.bg {
    position: absolute;
    left: 0;
    top: 0;
    bottom: 0;
    background-color: #00a302;
    z-index: -1;
}
.bg1 {
    position: absolute;
    left: 0;
    top: 0;
    bottom: 0;
    background-color: #dc9a02 ;
    z-index: -1;
}
.bg2 {
    position: absolute;
    left: 0;
    top: 0;
    bottom: 0;
    background-color: red;
    z-index: -1;
}
#h1container h1{
  text-align: center;
  font-size: 50px;
 text-transform: uppercase;
 text-shadow: #363535 3px 3px 5px;
 font-variant: small-caps;
}
.navbar {
    overflow: hidden;
    background-color: #333;
    font-family: Arial, Helvetica, sans-serif;
}

.navbar a {
    float: left;
    font-size: 13px;
    color: white;
    text-align: center;
    padding: 12px 13px;
    text-decoration: none;
}
.dropdown {
    float: left;
    overflow: hidden;
}

.dropdown .dropbtn {
    font-size: 13px;
    border: none;
    outline: none;
    color: white;
    padding: 12px 13px;
    background-color: inherit;
    font-family: inherit;
    margin: 0;
}

.navbar a:hover, .dropdown:hover .dropbtn {
    background-color: red;
}

.dropdown-content {
    display: none;
    position: absolute;
    background-color: #f9f9f9;
    min-width: 140px;
    box-shadow: 0px 8px 13px 0px rgba(0,0,0,0.2);
    z-index: 1;
}
```

```
.dropdown-content a {
   float: none;
   color: black;
   padding: 12px 13px;
   text-decoration: none;
   display: block;
   text-align: left;
}

.dropdown-content a:hover {
   background-color: #ddd;
}

.dropdown:hover .dropdown-content {
   display: block;
}
}

</style>
  </head>
 <body>
   <div id="h1container">
      <h1 align = "center">KMC Monitoring <h1>
     </div>
     <div class="navbar">
      <div class="dropdown">
     <button class="dropbtn">GO-TO <i class="fa fa-caret-down"></i></button>
      <div class="dropdown-content">
        <a href="geo_pos.php">Driver's Page</a>
        <a href="/phpmyadmin/sql.php?db=garbdb&table=sensor&pos=0">View/Change
Database</a>
        <a href="#">Contact US</a>
      </div>
     </div>
   </div>
  <table cellspacing="" style="width:70%" class="w3-table-all " align= "center">

                  <tr class=w3-grey>
                           <th><a
href="index.php?sort=Time">&nbsp;Timestamp&nbsp;</a></th>
                           <th><a
href="index.php?sort=nodeID">&nbsp;NodeID&nbsp;</a></th>
                           <th>&nbsp;Location&nbsp;</th>
     <th><a href="index.php?sort=Distance">&nbsp;Distance&nbsp;</a></th>
     <th ><a href="index.php?sort=ETA">&nbsp;ETA&nbsp;</a></th>
                           <th><a
href="index.php?sort=Percent_Filled">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&
nbsp;&nbsp;
     Percent Filled&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</a></th>
```

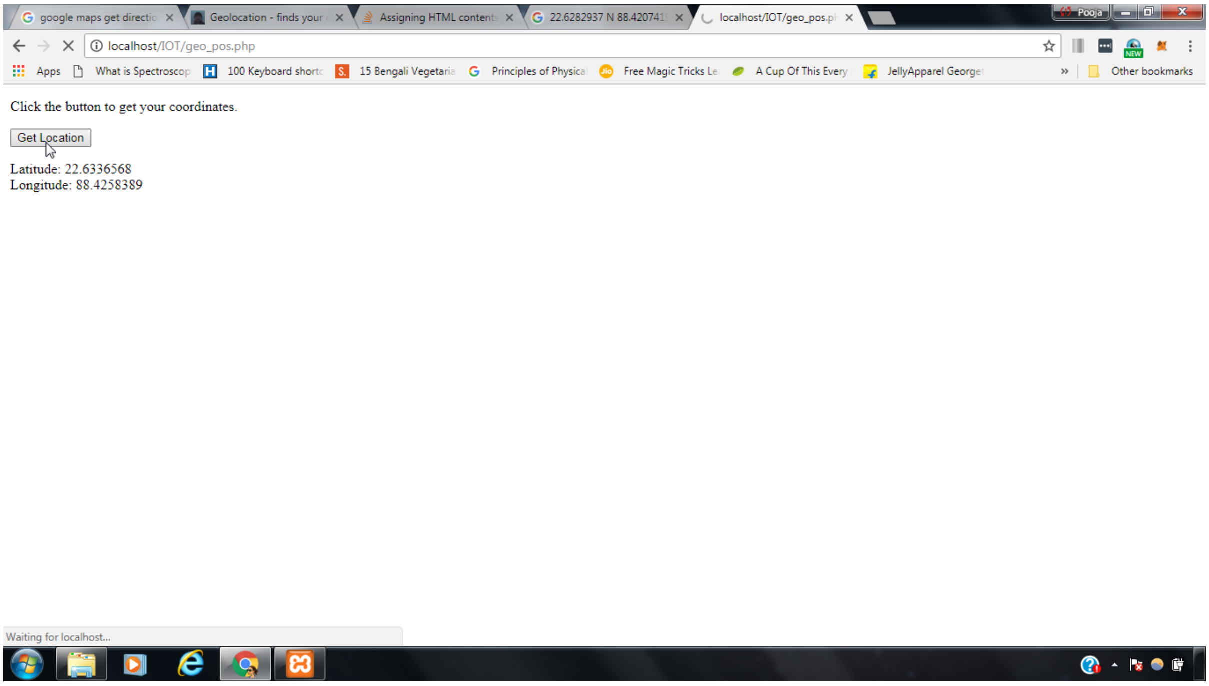
```
                    </tr>

<?php
  if($result!==FALSE)
  {
    while($row = mysqli_fetch_array($result))
    {
      if ($row['Percent_Filled']<=50)
      {
        echo
"<tr><td>".$row["Time"]."</td><td>".$row["nodeID"]."</td><td>".$row["Location"]."</td
><td>".$row["Distance"]."</td><td>".$row["ETA"]."</td><td><div class='bg'
style='width:".$row['Percent_Filled']."%'></div>".$row["Percent_Filled"]."</td></tr>";
            }
            elseif ($row['Percent_Filled']>50 && $row['Percent_Filled']<=75)
            {
              echo
"<tr><td>".$row["Time"]."</td><td>".$row["nodeID"]."</td><td>".$row["Location"]."</td
><td>".$row["Distance"]."</td><td>".$row["ETA"]."</td><td><div class='bg1'
style='width:".$row['Percent_Filled']."%'></div>".$row["Percent_Filled"]."</td></tr>";
            }
            elseif($row['Percent_Filled']>75)
            {
              echo
"<tr><td>".$row["Time"]."</td><td>".$row["nodeID"]."</td><td>".$row["Location"]."</td
><td>".$row["Distance"]."</td><td>".$row["ETA"]."</td><td><div class='bg2'
style='width:".$row['Percent_Filled']."%'></div>".$row["Percent_Filled"]."</td></tr>";
            }
                          }
                          mysqli_free_result($result);
                          mysqli_close($dbconnect);
                       }
      ?>

   </table>
 </div>
</body>
</html>
```

Sample Output:



Timestamp	NodeID	Location	Distance	ETA	Percent Filled
2018-05-12 02:27:17	1	Arjunpur	2.3 km	11 mins	25
2018-05-12 02:27:18	3	Clive House	2.3 km	10 mins	30
2018-05-12 02:29:26	2	Mall Road	0.9 km	4 mins	80

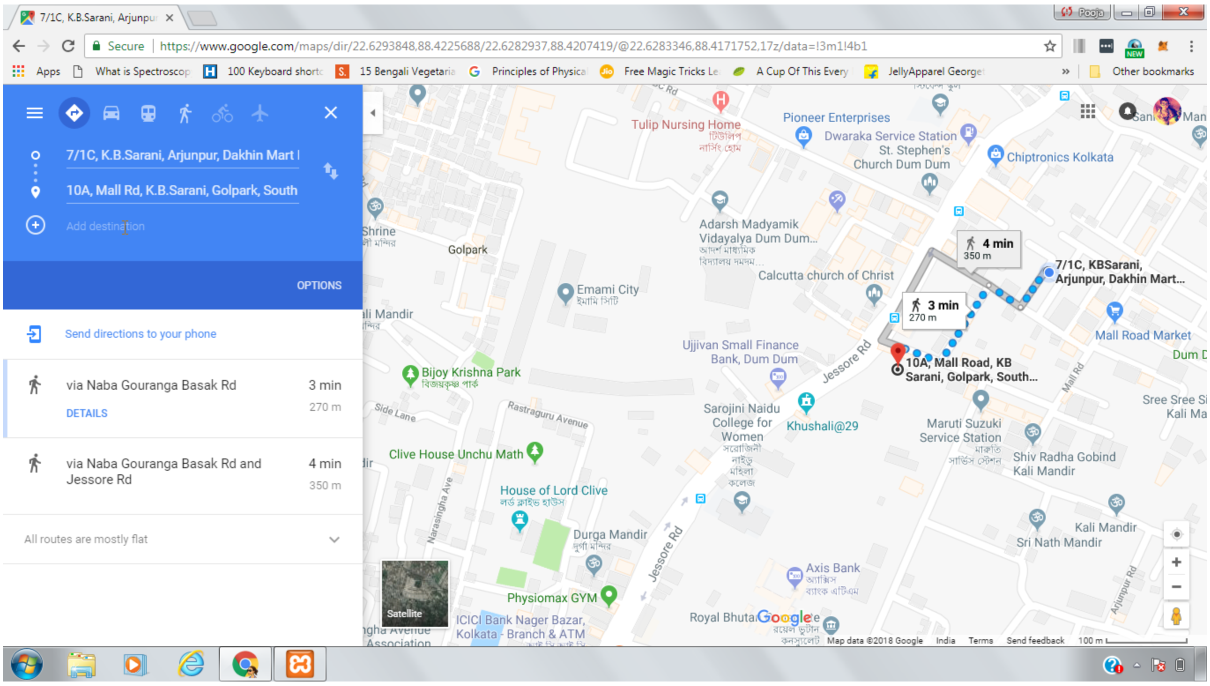
Fig6. The Home Page



Click the button to get your coordinates.

Latitude: 22.6336568
Longitude: 88.4258389

Fig7. Getting The Location



7/1C, K.B.Sarani, Arjunpur, Dakhin Mart I

10A, Mall Rd, K.B.Sarani, Golpark, South

Add destination

OPTIONS

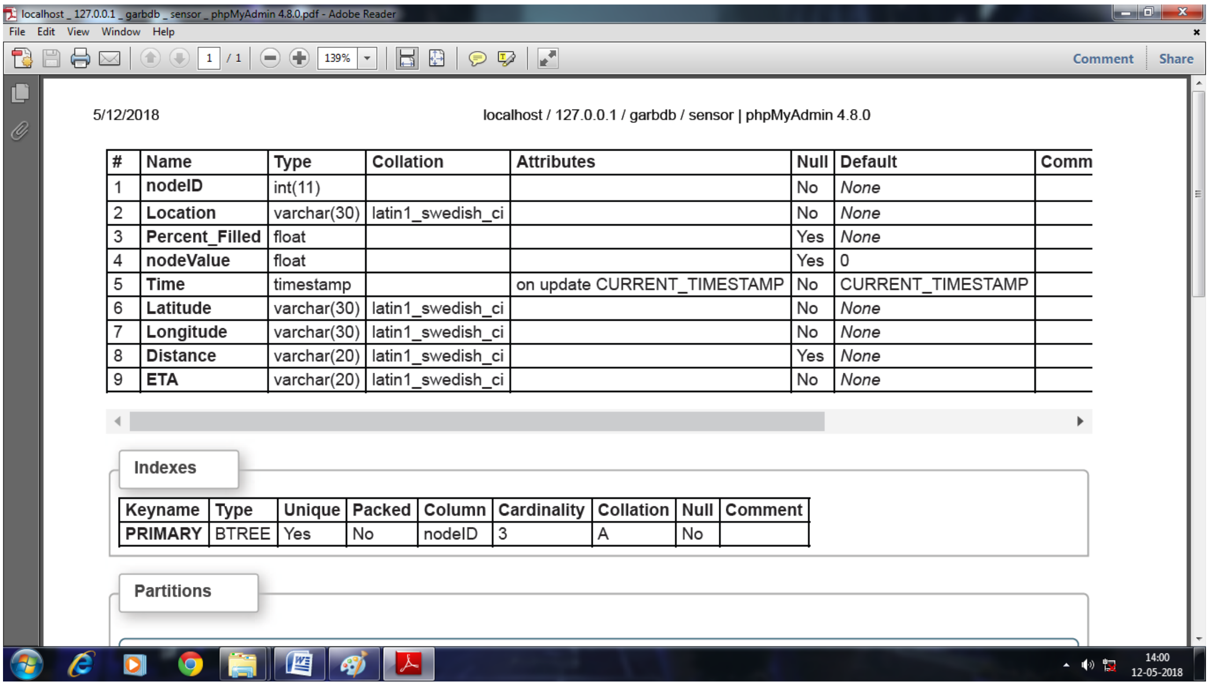
Send directions to your phone

via Naba Gouranga Basak Rd DETAILS	3 min	270 m
via Naba Gouranga Basak Rd and Jessore Rd	4 min	350 m

All routes are mostly flat

https://www.google.com/maps/dir/22.6293848,88.4225688/22.6282937,88.4207419/@22.6283346,88.4171752,17z/data=!3m1!4b1
4 min
350 m
3 min
270 m
7/1C, KBSarani, Arjunpur, Dakhin Mart...
10A, Mall Road, KB Sarani, Golpark, South...
Tulip Nursing Home
Pioneer Enterprises
Dwaraka Service Station
St. Stephen's Church Dum Dum
Chiptronics Kolkata
Adarsh Madyamik Vidyalaya Dum Dum...
Calcutta church of Christ
Golpark
Emami City
Ujjivan Small Finance Bank, Dum Dum
Mall Road Market
Bijoy Krishna Park
Sarojini Naidu College for Women
Khushali@29
Maruti Suzuki Service Station
Shiv Radha Gobind Kali Mandir
Clive House Unchu Math
House of Lord Clive
Durga Mandir
Kali Mandir
Sri Nath Mandir
Physiomax GYM
ICICI Bank Nagar Bazar, Kolkata - Branch & ATM
Axis Bank
Royal Bhuta
Jessore Rd
Rastraguru Avenue
Satellite
Map data ©2018 Google

Fig8.Next Stop with Directions



5/12/2018 localhost / 127.0.0.1 / garbdb / sensor | phpMyAdmin 4.8.0

#	Name	Type	Collation	Attributes	Null	Default	Comm
1	**nodeID**	int(11)			No	*None*	
2	**Location**	varchar(30)	latin1_swedish_ci		No	*None*	
3	**Percent_Filled**	float			Yes	*None*	
4	**nodeValue**	float			Yes	0	
5	**Time**	timestamp		on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP	
6	**Latitude**	varchar(30)	latin1_swedish_ci		No	*None*	
7	**Longitude**	varchar(30)	latin1_swedish_ci		No	*None*	
8	**Distance**	varchar(20)	latin1_swedish_ci		Yes	*None*	
9	**ETA**	varchar(20)	latin1_swedish_ci		No	*None*	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	nodeID	3	A	No	

Partitions

localhost _ 127.0.0.1 _ garbdb _ sensor _ phpMyAdmin 4.8.0.pdf - Adobe Reader
File Edit View Window Help
1 / 1
139%
Comment
Share
14:00
12-05-2018

Fig9. Database Structure

Conclusions and Future Scope

In this project, an integrated system of Wi-Fi modem, NodeMCU and Ultrasonic Sensor is introduced for efficient and economic garbage collection. The developed system provides improved database for garbage collection time and waste amount at each location. By implementing this project we will avoid over flowing of garbage from the container in residential area which is previously either loaded manually or with the help of loaders in traditional trucks. It can automatically monitor the garbage level & send the information to collection truck. The technologies which are used in the proposed system are good enough to ensure the practical and perfect for solid garbage collection process monitoring and management for green environment.

We have successfully completed the project “Smart Garbage Management”, but, there’s still room for change.

- For instance, we intend to expand our model and make it more dynamic.
- Our model works on the assumption that there’s only one collecting truck. This can be implemented on a larger scale, with more collector trucks, given the required funds and time.
- We also intend to make our routing algorithm better and more efficient.

References/Bibliography

[1] IOT based smart garbage alert system using Arduino UNO - N. Sathish Kumar ; B. Vuayalakshmi ; R. Jenifer Prarthana ; A. Shankar -22-25 Nov. 2016

[2] H. Rahman A. Al-Muyeed "Solid and Hazardous Waste Management" ITN-BUET Center for Water Supply and Waste Management 2010.

[3] http://www.instructables.com/id/Smart-Garbage-Monitoring-System-Using-Internet-of/

[4]Smart Garbage Monitoring System for Waste Management - MATEC Web of Conferences – 2017

[5] SMART GARBAGE COLLECTION SYSTEM IN RESIDENTIAL AREA –IJRET- GaikwadPrajakta1 ,Jadhav Kalyani2 , MachaleSnehal - 2012
[6]http://www.instructables.com/id/Powering-Arduino-with-a-Battery/

[7]http://www.14core.com/wiring-esp8266-with-bi-directional-logic-converter/

[8] https://www.losant.com/blog/making-the-esp8266-low-powered-with-deep-sleep

[9]www.setfirelabs.com(ultrasonic sensor interfacing with arduino and node MCU)

[10] IJIREEICE Volume 5, Issue 1, Smart Garbage Monitoring System using IoT - Prof. Dr.Sandeep M. Chaware1 , Shriram Dighe2 , Akshay Joshi3 , Namrata Bajare4 , Rohini Korke5 – 20 - January 2017

List of Tables

Table No.	Table Name
1.	Gnatt Chart
2	Deep Sleep Data

List of Figures

Figure No.	Figure Name
1.	Node MCU Pin Configuration
2.	Gnatt Chart
3.	Block Diagram
4.	Circuit Diagram
5.	Flowchart
6	The Home Page
7	Getting The Location
8	Next Stop
9	Database Structure