Efficient Garbage Management Using IoT

Ms.Lakshmi Priya S [1], Prasanna Sai Kanamaluru[2], Joshua Teja.S[3], Manigandan.S[4]
[1]Assitant Professor, Dept of C.S.E
[2, 3, 4]Dept of C.S.E
[1, 2, 3, 4] K.C.G College of Technology

Abstract- *The uncollected waste material when the waste bin is full is a common problem nowadays. Thus, an efficient waste management for the waste material is essential in ensuring a clean and green surrounding environment. This paper presents an Internet of Things (IoT) based Smart Waste Collection Monitoring and Alert System to monitor the waste material at the selected site of the garbage collection area. The system is implemented using an ultrasonic sensor which is connected to Arduino UNO to monitor waste bin garbage level. In this system, waste bin depth level will be sent via Arduino Ethernet Shield with an Internet connection to the Firebase IoT Cloud. The Firebase stores the collected waste bin level data into an IoT database and displays the waste bin depth level on an online dashboard for real-time visualization. The Firebase Event manager invokes a notification alert to garbage collector mobile phones via an SMS when the waste bin is nearly filled for immediate waste collection. Therefore, the waste collection became more effective and systematic.*

Keywords- waste management, IoT, smart bins, sensors, GPRS, Arduino UNO.

I. INTRODUCTION

The Introduction part explains the motivation behind the paper, the objective of the paper, the scope of impact and implementation, and an outline of the following sections with short explanations of each of them.

1.1 Motivation

Though there are various cleanliness campaigns, missions and policies such as the Swachh Bharat Mission, still there are many countries such as our India which is highly polluted by the disposable waste produced by its people. If properly observed we can find that the main problem lies in the perception of people and hence we came up with an idea to change this perception of people by making them take this issue of waste management seriously through money rewards.

1.2 Objective

The objective of this project is to evaluate, analyze and implement different technologies surrounding the Internet of Things and develop a smart waste management system suitable for smart cities.

II. LITERATURE SURVEY

A Smart Dustbin proposed by [1], based on IoT in which the smart bin was built on a platform which was based on an Arduino Uno board which was interfaced with a GSM modem and an ultrasonic sensor. The sensor was placed on the top of the bin. A threshold level was set as 10cm. As the garbage reaches the level of threshold, the sensor triggers the GSM modem which alerts the associated authority till the garbage in the bin is emptied. At the end a conclusion was made that various issues like affordability, maintenance and durability were addressed when these smart bins were designed. It also contributed towards a hygienic and clean environment in the process of building a smart city.

The researchers [2], suggest the method for garbage management which is as follows. The bin was interfaced with a system based on microcontroller which had IR wireless systems with a central system that showed the current status of the garbage in the bin. The status was seen on a mobile based web browser with an html page by using Wi-Fi. To reduce the cost, they only used weight based sensors and on the sender's side they only used a Wi-Fi module to send and receive the data. In the end the sensor could only detect the weight of waste present in the bin but not the level of waste.

The author proposed a method for organizing the collection of the garbage in the commercial and residential areas of the cities [3]. In this system, the level of garbage in the bin was detected by the ultrasonic sensor which will send the data to the control room using the GSM module. A GUI was also developed to check the information that was related to the garbage for different locations, GUI was based on MATLAB so it was different. Two units were present in the system; the slave unit was in the bin whereas the master unit was there in the control room. The sensor will check the level of garbage and send it to the slave unit which will further send the data to the master unit which at last will inform the authorities to clean the bin.

This paper proposes a Decision Support System which would be used for garbage collection in the cities [4]. This system handled the ineffective waste collection in the inaccessible areas of the city. The cameras were placed in those parts of the cities which were facing the most problems. The system worked in two parts, the first part was to find the companies that were involved in collecting the waste and owned trucks and who could also organize some drivers for collecting the garbage from various parts of the city in the truck and pass on the city dumps or the recycling organizations. The second part was to make a system which could handle all the communications of all the people involved and could also maintain the data which will be collected while working around in the city.

III. SYSTEM ANALYSIS

3.1 EXISTING SYSTEM:

In the existing system, typically the populace in the city locations is boosting day after day. These bring about the extra requirements of the resident in the metropolitan location. As the populace is enhancing the quantity of the waste generated is likewise high. The environments obtain filthy delicately. This causes extreme strike on the atmosphere if it is not recycled and also there by impacts a person's wellness.

3.2 PROPOSED SYSTEM:

The existing system has limitations as time consuming, trucks go and empty the containers, even they are empty. The cost is high with an unhygienic environment. Even the bad odour causes the unhealthy environment. So, the proposed model talks about how to make use of the recent advancements in technology to make our place clean and tidy. The implementation starts by setting ESP8266 by flashing the latest version of the firmware. To flash the latest firmware, download the ESP8266 flasher tool and the latest firmware from the internet which would be in the bin format and flash the ESP8266 with it. Once the ESP8266 flashing is done, other components can be added to the configuration.

3.3 REQUIREMENT SPECIFICATIONS

The requirements specification is a technical specification of requirements for the software products. It is the first step in the requirements analysis process; it lists the requirements of a particular software system including functional, performance and security requirements. The requirements also provide usage scenarios from a user, an operational and an administrative perspective. The purpose of software requirements specification is to provide a detailed overview of the software project, its parameters and goals. This describes the project target audience and its user interface, hardware and software requirements. It defines how the client, team and audience see the project and its functionality.

3.4 HARDWARE AND SOFTWARE SPECIFICATION

Software Requirement:

Language	:	c++
Compiler	:	Arduino IDE
OS	:	windows or LINUX

Hardware Requirement:

- Arduino uno
- ESP8266
- Ultrasonic sensor
- GPRS modem
- LCD 16x2
- Power supply

3.5 BLOCK DIAGRAM DESCRIPTION:

Above the block diagram contains an arduino uno controller, ultrasonic sensor, GPRS modem, and LCD and power supply unit. Ultrasonic sensor is connected to the GPIO pin of the Arduino uno. LCD also connected to the GPIO pin of the controller. Power supply unit is providing power to the controller. GPRS modem and ESP8266 are connected to the UART port of the controller.

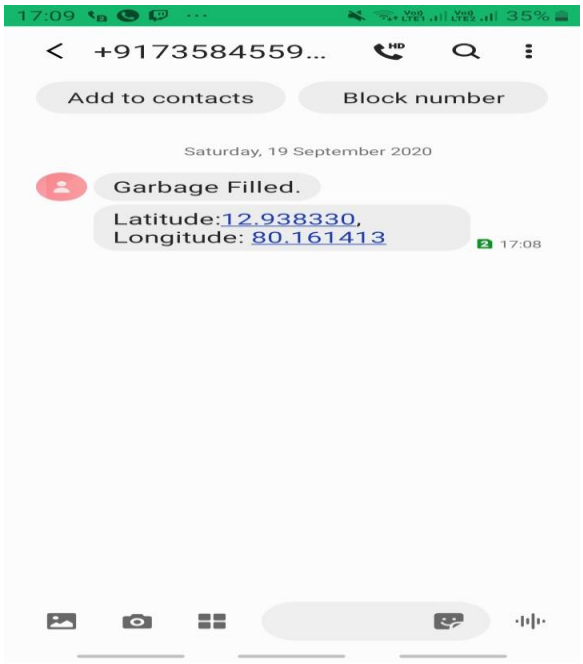
3.6 CODE

```
#include <SoftwareSerial.h>
SoftwareSerial mySerial(10, 11);
#include <LiquidCrystal.h>
const int trigPin = 9;
const int echoPin = 8;
// defines variables
long duration;
int distance;
// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(7, 6, 5, 4, 3, 2);
//String S,S1,s2,s3;
String json,s,s1,s2,s3,s4;
int swt,b,b1;
int sensor;
String e="Latitude:";
String e1="Longitude:";
String e2;
```

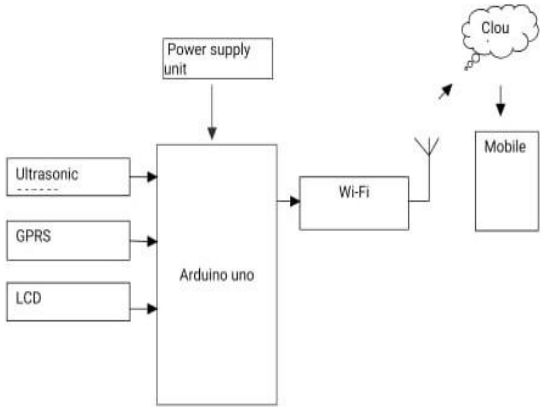
```
float a,a1;
char buf[50],buf1[50];
void setup() {
   lcd.begin(16, 2);
  Serial.begin(9600);
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin, INPUT);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Garbage Monitoring");
  lcd.setCursor(0, 1);
  lcd.print("     System");
  mySerial.begin(9600);
 // mySerial.begin(9600);
  //pinMode(Dev1, INPUT);
  //pinMode(Dev2, INPUT);
  delay(1000);
  mySerial.println("AT\r");
  delay(1000);
  mySerial.println("ATE0\r");
  delay(1000);

mySerial.println("AT+SAPBR=3,1,\"Contype\",\"GPRS\"\r");
  delay(1000);
  mySerial.println("AT+SAPBR=3,1,\"APN\",\"internet\"\r");
  delay(1000);
  mySerial.println("AT+SAPBR =1,1\r");
  delay(1000);
  mySerial.println("AT+SAPBR=2,1\r");
  delay(1000);
  mySerial.println("AT+CLBSCFG=0,1\r");
  delay(1000);
  mySerial.println("AT+CLBSCFG=0,2\r");
  delay(1000);
}

void loop() {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel time in
microseconds
  duration = pulseIn(echoPin, HIGH);

// Calculating the distance
  distance= duration*0.034/2;
  Serial.println(distance);
  if(distance>23)
  {
  digitalWrite(4, HIGH);
  //Serial.println("pin4 High");
  digitalWrite(2, LOW);
  digitalWrite(3,LOW);
  //digitalWrite(4, LOW);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Garbage Empty");
  lcd.setCursor(0, 1);
  lcd.print("Diatance:");
  lcd.setCursor(9, 1);
  lcd.print(distance);
  delay(1000);
  }
  if(distance>10&&distance<18)
  {
  digitalWrite(3, HIGH);
  //Serial.println("pin3 High");
   digitalWrite(2, LOW);
  //digitalWrite(3,LOW);
  digitalWrite(4, LOW);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Garbage Half Filled");
  lcd.setCursor(0, 1);
  lcd.print("Diatance:");
  lcd.setCursor(9, 1);
  lcd.print(distance);
  delay(1000);
  }
  if(distance<5)
  {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Garbage filled");
  lcd.setCursor(0, 1);
  lcd.print("   ");
  s4=1;
  while(s4=="1")
  {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("SMS SENDING.....");
  mySerial.println("AT+CLBS=4,1\r");
  s = mySerial.readString();
  //Serial.println(s);
  delay(3000);
  int slen = s.indexOf(":");
  int elen = s.indexOf("/");
  s1 = s.substring(slen + 1, elen);
```

```
  s2 = s1.substring(3, 13);
  s3 = s1.substring(13, 23);
  s2.toCharArray(buf, sizeof(buf));
  a = atof(buf);
  b = atoi(buf);
  s3.toCharArray(buf1, sizeof(buf1));
  a1 = atof(buf1);
  b1 = atoi(buf1);
  if (b < 10 && b1 < 10)
  {
  }
  else if (b > 10 && b1 > 10)
  {
    //json = "{\"Latitude\":" + String(a) + ",\"Longitude\":" +
String(a1) + ",\"Sensor_value\":" + String(sensor) + "}";
    //Serial.println(a, 6);
    //Serial.println(a1, 6);
     e+=s2;
     e1+=s3;
     e2+=e;
     e2+=e1;
     char buff[200];
     e2.toCharArray(buff, sizeof(buff));
//sendSMS("9791169791",e+char(buf)+e1+char(buf1));
    sendSMS("9791169791",buff);
    sendSMS("9791169791","Garbage filled");
      e="Latitude:";
      e1="Longitude:";
      e2="";
     s4="";
  }
  s="";
  }
  }
  delay(1000);
}
void sendSMS(String number,char *message) {
  mySerial.println("AT\r");
 // String s=Serial.readString();
 // Serial.println(s);
  delay(1000);
  mySerial.println("ATE0\r");
//  s=Serial.readString();
 // Serial.println(s);
  delay(1000);
  mySerial.println("AT+CMGF=1\r");
 // s=Serial.readString();
 // Serial.println(s);
  delay(1000);
  mySerial.println("AT+CMGS=\""+number+"\"\r");
 // s=Serial.readString();
 // Serial.println(s);
  delay(1000);
  mySerial.println(message);
  //s=Serial.readString();
 // Serial.println(s);
  delay(1000);
  mySerial.println((char)26);
//  s=Serial.readString();
  //Serial.println(s);
  delay(1000);
 // Serial.println();
  delay(5000);
 }
```

3.7 OUTPUT



3.8 BLOCK DIAGRAM



IV. CONCLUSION

In this project, we propose a new solution to enhance waste collection efficiently using the Arduino Uno with Arduino Ethernet Shield technology and ultrasonic sensor systems. In this proposed system, the garbage overflow of garbage can be avoided and managed efficiently. This will intimate or send SMS to the authorized person through firebase platform. The garbage managing system and the facility of collecting the garbage presently doesn't fit to the current requirement. Hence better facilities of collecting garbage and transportation should be provided. Since, this system provides the information when the bin gets completely filled with garbage, it reduces the number of times the arrival of a vehicle which collects the garbage. This method finally helps in keeping the environment clean. Thus, the waste collection is made more efficient.

REFERENCES

[1] A SMART WASTE MANAGEMENT SYSTEM USING IOT AND BLOCKCHAIN TECHNOLOGY
2017 Manish Lamichhanes
[2] Smart Bins for Smart City Pune Municipal Corporation.
[3] World Health Organization. MHealth: New Horizons for Health Through Mobile Technologies. Accessed: Jan. 2019. [Online].
[4] M. Gerla, E.-K. Lee, G. Pau, and U. Lee, "Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds," in Proc. IEEE World Internet Things (WF-IoT), Mar. 2014, pp. 241–246.