IoT Based Air Pollution Monitoring System

A minor project report

submitted in partial fulfillment of the requirement for the award of

B.Tech. degree in Computer Science and Engineering

by

Akashdeep Goel (2017BCS-009)
Sajal Verma (2017BCS-025)

Under the supervision of:

Prof. Aditya Trivedi
Dr. Santosh Singh Rathore

विश्वजीवनामृतं ज्ञानम्

ABV INDIAN INSTITUTE OF INFORMATION TECHNOLOGY AND MANAGEMENT GWALIOR-474015

2019

CANDIDATES DECLARATION

We hereby certify that the work, which is being presented in the report, entitled **IoT based Air Pollution Monitoring System**, in partial fulfillment of the requirement for the award of the Degree of **Bachelor of Technology** and submitted to the institution is an authentic record of our own work carried out during the period *May 2019* to *July 2019* under the supervision of **Pror. Aditya Trivedi** and **Dr. Santosh Singh Rathore**. We also cited the reference about the text(s)/ figure(s)/ table(s) from where they have been taken.

Date: Signatures of the Candidates

1. Akashdeep Goel
2. Sajal Verma

This is to certify that the above statement made by the candidates is correct to the best of our knowledge.

Date: Signatures of the Research Supervisors

1. Prof. Aditya Trivedi
2. Dr. Santosh Singh Rathore

ABSTRACT

The biggest challenge of every country is air environmental deterioration. Health problems have increased rapidly mostly in urban areas, where the major pollution is due to the industrialization and growing number of transportation vehicles, which lead to release of lot of polluting agents. The aim of the project is to make an IoT based air pollution monitoring system, which will measure the air pollution level and send it to the web server and activate an alarm when the air pollution level go beyond a threshold. ”The Internet of things (IoT) is an extension of Internet connectivity into physical devices and everyday objects. Embedded with electronics, Internet connectivity, and other forms of hardware (such as sensors), these devices can communicate and interact with others over the Internet, and they can be remotely monitored and controlled” . Initially the hardware was assembled and then the microcontroller was programmed using an arduino. The main idea is to alert the people when they are entering any area, which has high content of toxic gases or particulate matter so that person can take necessary actions to protect himself/herself. The hardware has sensors for the detection of PM2.5 level in air and also for the detection of other harmful gases such as carbon monoxide, carbon dioxide, oxides of nitrogen, etc. The output of the project would be a portable device, which can be installed at any location to trace air pollution level.

ACKNOWLEDGEMENT

When any project work is undertaken, one thing has to be made sure that it is a complete team effort. During the on-phase of our innovative project, we faced some anxious moments, some very tense ones, and some very euphonic ones; finally we made it to limelight. A successful project work is the result of teamwork, which contains not only the people who put their logic, but also who guide them. We are deeply grateful and express our deep sense of gratitude to our project supervisors, **Prof. Aditya Trivedi**, Professor, ABV-IIITM Gwalior and **Dr. Santosh Singh Rathore** Asst. Professor, ABV-IIITM for the instructions, suggestions, encouragement, and invaluable generous support throughout the project work. We thank them for giving their precious time and shall remain indebted to them throughout our life. We are greatly thankful to **Prof. S.G. Deshmukh** Director, ABV-IIITM Gwalior for providing all the help and facilities to carry out the research work. We are obliged to staff members of the institute, who provided us with the components. We are grateful for their cooperation during the period of our assignment. Finally, we must express our very profound gratitude to our loving parents for providing us active support and helping us throughout our studies, without which it would have been impossible to undertake and complete project work. This accomplishment would not have been possible without them. We shall remain indebted to them throughout our life.

(Akashdeep Goel) (Sajal Verma)
(2017BCS-009) (2017BCS-025)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AC: Alternating Current
CPU: Central Processing Unit
DC: Direct Current
GPRS: General Packet Radio Service
GSM: Global System for Mobile
IoT: Internet of things
LCD: Liquid Crystal Display
mA: Milliampere
PC: Personal Computer
PPM: Parts per Million
PM: Particulate Matter
SMS: Short Message Service
UART: Universal Asynchronous Receiver transmitter
USB: Universal Serial Bus
VCC: VOLTAGE collector to collector

CHAPTER 1

Introduction

"The Internet of things (IoT) is the extension of Internet connectivity into physical devices and everyday objects. Embedded with electronics, Internet connectivity, and other forms of hardware (such as sensors), these devices can communicate and interact with others over the Internet, and they can be remotely monitored and controlled" [1]. Generally, MQ135 sensor is used for monitoring air quality as it detects most toxic gases and can measure their amount precisely. Additional hardware includes a PM 2.5 sensor, which can give the level of PM 2.5 particles in the air. The PM 2.5 sensor and the GSM module are both connected to the microcontroller using UART protocol. This project aims to develop a device that can monitor the environmental deterioration level from anywhere using system and we can easily see the air quality level in any portable device such as mobile phones. This device can be installed where-ever we want and can also activate some device when environmental deterioration goes beyond some level, like we can send alert SMS to the user. It will show the air quality in PPM on the LCD and as also on webpage so that air environmental deterioration level can be monitored very easily.

1.1 Motivation

Effects of environmental deterioration include mild allergic reactions such as suffocation or irritation of the throat, eyes and nose as well as some serious life threatening problems such as pneumonia, lung bronchitis heart diseases, and aggravated asthma. According to a study , due to air environmental deterioration 55,000 to 110,000 premature deaths per year occur in the United States alone. In Europe, this figure reaches to 350,000 and over 3,500,000 across the globe [2]. Every day, we listen to the news where millions of people die due to cancer, which is supposed to be a direct consequence of air pollution. If we want to survive in long run then we have to control this, and in order to control we must initially know about it. People need to be made aware about quality of air, which surrounds them at every moment.

1.2 Objectives

1. To develop an IoT based air pollution monitoring device that monitors the air quality over a web server using Internet.
2. To install an alert mechanism in the device that activates an alarm when the air quality goes down beyond a threshold means when there are sufficient amount of toxic gases present in the air.

1.3 Literature Review

The increase of manufacturing industries and the use of vehicles and in cities results in vital increase in the emission load of various toxins into air. As a result increase in environmental problems which will affect the human health in urban places [3]. The IoT concept was put forth in 1999 by a member of the Radio Frequency Identification (RFID) development community. The increasing growth of the mobile devices and development in the area of communication, cloud computing, embedded systems has made the IoT concept more relevant. In [4] it is stated that IoT is a converging technology to create the smart environment and the integrated ecosystem.
The shortcomings of the traditional instruments which are used to monitor air quality are their heavy weight, large size, and extraordinary expensiveness [5]. The programming approach that we have used is the Arduino IDE, which utilizes the C programming language. This gives you access to an enormous Arduino Library that is constantly growing thanks to open-source community [6].

1.4 Features of the device

The device would be portable and handy. The device would have an automatic message sending feature. As the device is connected to the servers and one can see the pollution level of any particular area from anywhere.

1.5 UART Protocol

In UART communication, two UARTs communicate directly with each other. The transmitting UART converts parallel data from a controlling device like a CPU into serial form, transmits it in serial to the receiving UART, which then converts the serial data back into parallel data for the receiving device. Only two wires are needed to transmit data between two UARTs. Data flows from the Tx pin of the transmitting UART to the Rx pin of the receiving UART. UARTs transmit data asynchronously, which means there is no clock signal to synchronize the output of bits from the transmitting UART to the sampling of bits by the receiving UART. Instead of a clock signal, the transmitting UART adds start and stop bits to the data packet being transferred. These bits define the beginning and end of the data packet so the receiving UART knows when to start reading the bits. When the receiving UART detects a start bit, it starts to read the incoming bits at a specific frequency known as the baud rate. Baud rate is a measure of the speed of data transfer, expressed in bits per second (bps). Both UARTs must operate at about the same baud rate. The baud rate between the transmitting and receiving UARTs can only differ by about 10 percent before the timing of bits gets too far off [7].

Table 1.1: **Configuration of PM 2.5 sensor(uses UART protocol)**

Wires used	2
Maximum speed	Any speed up to baud 115200,usually 9600 baud
Synchronous or asynchronous	Asynchronous
Serial or Parallel	Serial
Max of Masters	1
Max of slaves	1

1.6 System Architecture

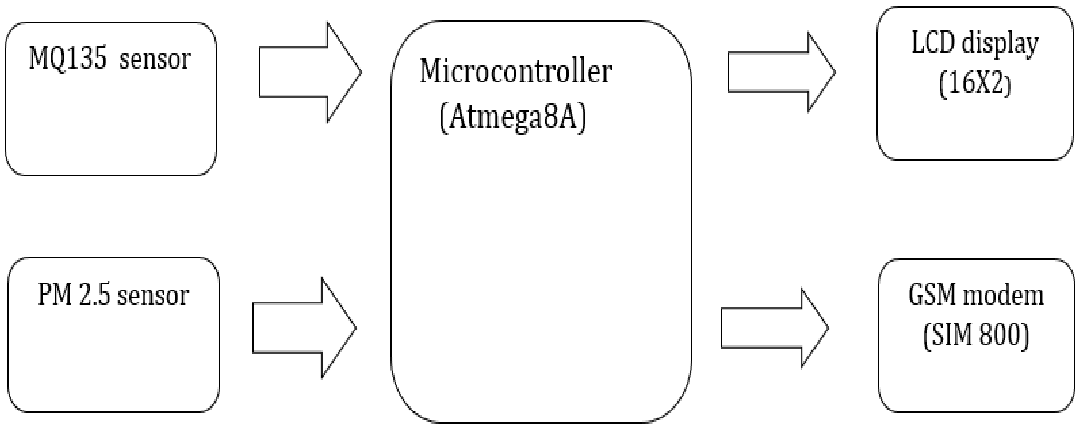

Figure 1.1: **Hardware representation of IoT based air pollution monitoring device**

1.7 Overview of Circuit Diagram

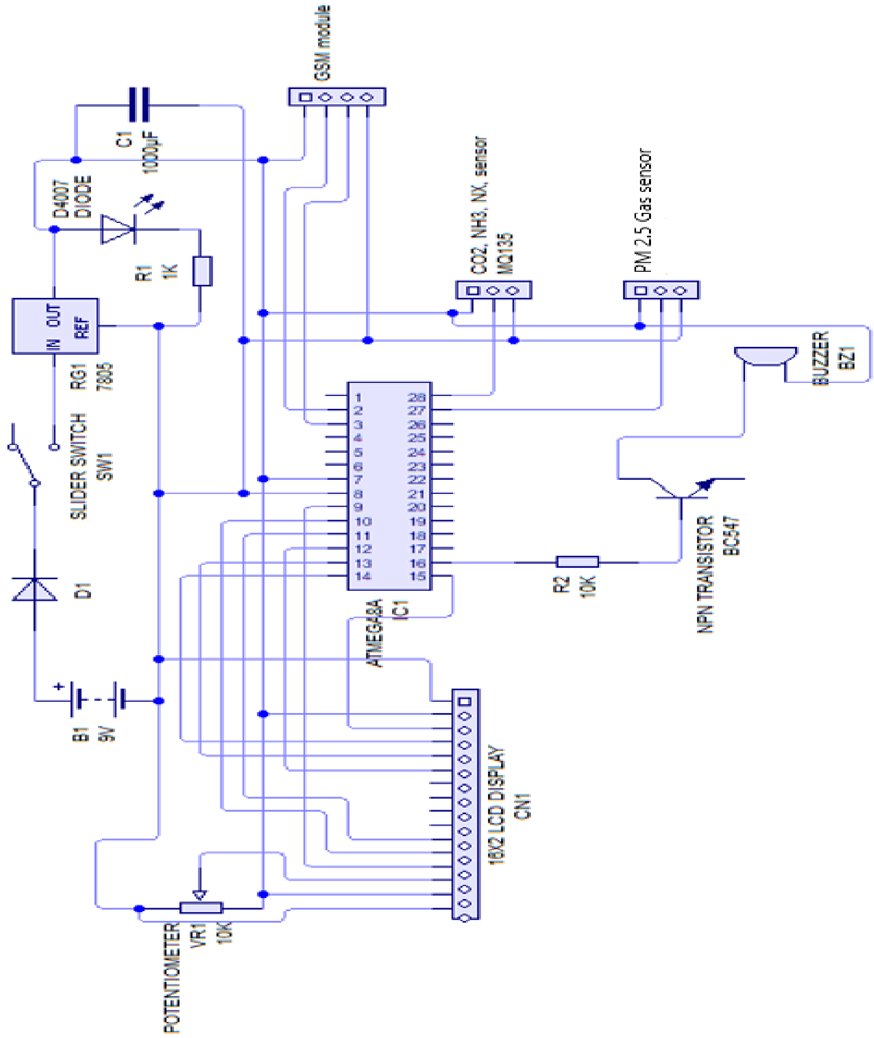

Figure 1.2: **Circuit diagram for hardware of IoT based air pollution monitoring device**

1.8 Report layout

Report layout: The report is divided into 4 chapters. The first chapter gives the basic idea about the project, topic and the motivation behind selecting the topic. Chapter 2 gives the insight about the individual working of the components, their specification their utility in the project and about the working of the hardware as a whole. It also gives insight about the challenges resolved in the project with the help of programming. Chapter 3 of the project describes how the project was implemented in different phases and tells about the challenges faced during each phase and the tasks completed in each phase. Chapter 4 tells the conclusion and future scope of the project.

CHAPTER 2

Working of the Hardware

This section first discusses the working of sensors used in the device and then it provides details of the working of the hardware as a unit.

2.1 Working of components

2.1.1 Atmega8A microcontroller:

Microcontrollers are made for embedded applications, in contrast to the microprocessors, which are used in personal computers or other general purpose systems. Atmega 8A microcontroller has 28 pins of type B,C,D. Type C pins are used for analog input(like for establishing connection between the microcontroller and sensors) and type B and type D are used to establish connection with the LCD display, buzzer etc. By default the microcontroller has only one set of pin which works on UART protocol [8].

Table 2.1: **Pin configuration of Atmega8A Micro controller**

Port type	Pin number
B	9,10,14,15,16,17,18,19
C	1,23,24,25,26,27,28
D	2,3,4,5,6,11,12

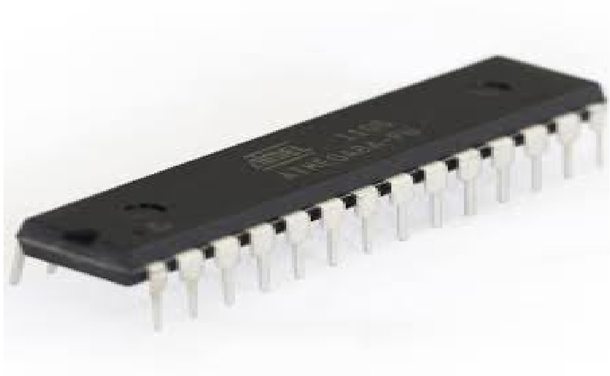

Figure 2.1: **Atmega8A Micro-controller**
(Source: https://www.microchip.com/wwwproducts/ATmega8A)

2.1.2 SIM800 GSM module:

Global Device for Mobile communication (GSM) module is used to establish communication between a system and a GSM device. It is an architecture used for mobile communication in most of the nations. GSM/GPRS module consists of a GSM modem assembled with power supply circuit and communication interfaces (like USB,RS-232, etc.) for systems. This device is responsible for sending the date to the web server and also for message sending. This device is connected to the micro-controller using UART interface [9].

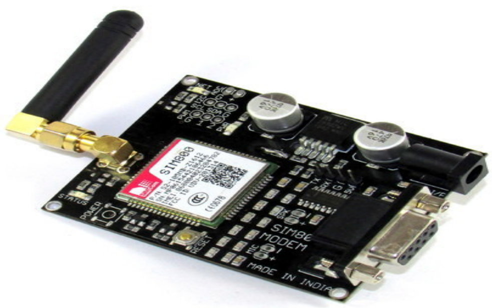

Figure 2.2: **SIM800 GSM module**

(Source: https://www.rhydolabz.com/wireless-gsm-gprs-c-130$_1$85/$sim - 800 - module - p - 2055.html$)

2.1.3 MQ135 (carbon dioxide, ammonia, alcohol, benzene sensor)

The MQ135 alcohol sensor is a Sno2 with a lower conductivity of clean air. When the explosive fumes exists, then the sensor conductivity grows more and more along with the growing gas concentration rising levels. By using simple electronic circuits, it converts the charge of conductivity to correspond output signal of gas concentration. Detection Range: 10 - 300 PPM. NH3, 10 - 1000 ppm Benzene, 10 - 300 Alcohol. Its circuit operates on 5 voltage, works both on AC and DC.MQ-135 sensor has 4 pins. First pin name is VCC, which is used to give power to sensor. Second pin name is Ground ,which is used to connect the module to ground, Third pin name is Digital out which is used to get digital output from the sensor and Last and forth pin name is Analog pin which is used to give analog output in the range of 0-5V .Its value depends upon intensity of the gases [10].

Figure 2.3: **MQ135 Gas sensor**
(Source: https://components101.com/sensors/mq135-gas-sensor-for-air-quality)

2.1.4 PM 2.5 Sensor:

Aerosol particles that are less than 2.5 micrometer in width or diameter are termed as PM 2.5 particles. This sensor detects the presence of such particles in air and give the value in PPM i.e. parts per million. These are the major cause of air pollution. The mass concentration value that the sensor can sense ranges from 1 to 1000ug/m3. The lower limit of detection i.e. the minimum diameter of the particle must be 0.3 micrometer. The ideal operating voltage is 5V and current is 50mA. The sensor gives best result when the temperature is between -10C to 60C. Also the device is expected to work for 8 years if used 24 hours a day [11].

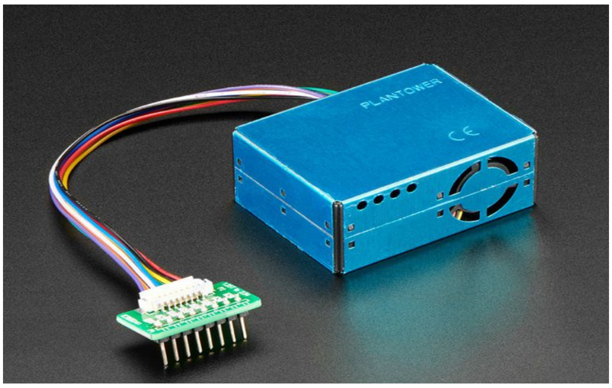

Figure 2.4: **PM 2.5 sensor**
(Source: https://www.sensirion.com/en/environmental-sensors/particulate-matter-sensors-pm25/)

2.2 Procedure

All the above mentioned hardware components are connected with microcontroller. The sensors give the analog value as input to the microcontroller which converts it to the corresponding digital output value using programming and will display it on the LCD. The power adapter provides the potential difference of 9V and the required potential drop is 5V for proper working of the components, hence a voltage regulator is used to provide the required drop. GSM module requires 4V of potential drop, hence a separate potential regulator have been used for it. The PM 2.5 sensor works on UART interface and is connected to the microcontroller. The value will be transmitted using the GSM module and the alarm will be triggered and a message will be sent when the value crosses a certain threshold. The GSM module is connected to the microcontroller using UART interface and by default there is only one set of pins which are configured as UART in the microcontroller, hence there was a need to explicitly make another set of UART pins in the microcontroller, which was achieved with the help of programming. Additionally, the capacitors and resistors are used to give the required current output to the sensors. Also the appropriate baud rate for the sensors and microcontroller used is 9600, hence the GSM module is configured at the baud rate of 9600.

CHAPTER 3

Project Implementation and Results

This chapter gives the insight about the phase wise development of the project ,conclusions drawn and about the scope for future developments.

3.1 Project Implementation

We have implemented the project in three phases.The first phase includes Topic search, Problem identification, Solution to the problem and Gathering required knowledge. Second phase includes Hardware identification, Hardware procurement and Hardware assemble. Third phase includes Coding implementation and System testing.

3.1.1 Resource Gathering

First we did research to find the topic of the project.Focus was on searching a topic/idea that is connected to real day to day life of people across the globe, which can be utilized for betterment of human life and which can be realized by us.Then we identified the problem, the problem is that there are several instruments, which measure the quality of air but the main problem is the mobility of device and the need to spread message, when the air quality goes beyond a certain value. After problem identification, the solution was found by us, which is an IoT based air quality monitoring system, which can trigger an alarm and send message when the air quality goes beyond a certain value.Then we gathered the required information regarding the hardware components.

3.1.2 Hardware Procurement and Assembling

In this phase, first we search for the hardware components which can be used in our project. Then we ordered these hardware components and then procured these hardware components from various sources. Finally we assembled the hardware components.

3.1.3 Coding Implementation and System Testing

First the coding part was implemented in the microcontroller. The microcontroller was attached to an Arduino, which was further attached to a PC and the coding part was implemented. With the help of codes the analog value is converted to the digital one and also it does sync the sensors with the GSM module.Then we did the system testing, we tested the system in various environmental conditions. Finally the report formulation was done.

3.2 Code Snippets

```
#include <SoftwareSerial.h>                          // header file for creating extra UART pins
#include <hpma115S0.h>                               // PM2.5 header file
#include <LiquidCrystal.h>                           // LCD header file
#define buzzerOn() digitalWrite(A0, HIGH)            // redefined buzzer on command
#define buzzerOff() digitalWrite(A0, LOW)            // redefined buzzer off commnad

#define token "A1E-9MX88FI9FzACosaZYKTkqS8Plo7CD4"  // Server ID
#define device "demo"                                // Server's Device ID

LiquidCrystal lcd(7,9,10,11,12,13);                  // object for lcd with pin no
SoftwareSerial hpma1(A5,A4);                           // object for extra UART with rx tx pin
HPMA115S0 hpma115S0(hpma1);                            // object for pm2.5 sensor

unsigned int pm2_5, pm10, count=0;                           //variable
float gas;

void setup()
{
  Serial.begin(9600);                                // begin Serial protocol with 9600 baud rate
  hpma1.begin(9600);                                   // begin extra UART protocol with 9600 baud rate
  lcd.begin(16,2);                                   // begin lcd with lcd size
  pinMode(A0, OUTPUT);
   lcd.print("Wait.....");
   delay(5000);
   hpma115S0.Init();                                    // initialize PM2.5 sensor
   hpma115S0.StartParticleMeasurement();                // Start measurement
   lcd.clear();
   lcd.print("DONE.....");
   buzzerOn();
   delay(1000);
   buzzerOff();
 }

 void loop()
 {
  for(int i=0; i<50; i++)
  {
   lcd.clear();
   int adc= analogRead(A3);                                      // read digitalValue
   gas=adc*100.0/1023;                                           // convert in %
   lcd.setCursor(0,0);
   lcd.print("GAS:");
   lcd.print(gas);                                               // print value on lcd
```

Figure 3.1: **Starting Code for values initialization**

```
  lcd.print("%");
  lcd.setCursor(0,1);
  if (hpma115S0.ReadParticleMeasurement(&pm2_5, &pm10))      // reading particle measurement pm2.5 and pm10
  {
    lcd.print("PM 2.5: " + String(pm2_5) + " ug/m3" );       // print value on lcd
  }

  if((pm2_5>100 || gas>20) && count==0)
  {
    msgSend();
    count=1;
  }

  if((pm2_5<50 || gas<10)) count=0;

   delay(1000);
 }
 send2server();                                              // calling server Function
}
void send2server()
{
 int num;
 String le;
 String var;
 var="{\"gas\":"+(String)gas+", \"pm2.5\":"+(String)pm2_5+"}";
 num=var.length();
 le=String(num);

lcd.setCursor(15,0);
lcd.print("U");

Serial.println("AT+CIPSHUT");
delay(3000);

Serial.println("AT+CIPSTART=\"tcp\",\"things.ubidots.com\",\"80\""); //start up the connection
delay(8000);

Serial.println("AT+CIPSEND"); //begin send data to remote server
delay(4000);

Serial.print("POST /api/v1.6/devices/"); Serial.print(device);
delay(100);

Serial.println("/ HTTP/1.1");
```

Figure 3.2: **Main logic**

```
delay(100);

Serial.println("Content-Type: application/json");
delay(100);

Serial.println("Content-Length: "+le);
delay(100);

Serial.print("X-Auth-Token: ");
delay(100);

Serial.println(token);
delay(100);

Serial.println("Host: things.ubidots.com");
delay(100);

Serial.println();
delay(100);

Serial.println(var);
delay(100);
Serial.println();
delay(100);

Serial.println((char)26);
delay(7000);
Serial.println();

Serial.println("AT+CIPCLOSE"); //close the communication
delay(1000);
lcd.setCursor(19,0);
lcd.print(" ");
}

void msgSend()
{
  Serial.println("ATE0");
  delay(1000);
  Serial.println("AT+CMGS=\"9755204250\"");
  delay(500);
  Serial.println("!!! ALERT !!!");
  Serial.println("PM 2.5: " + String(pm2_5) + " ug/m3");
  Serial.println("GAS:" +String(gas));
  delay(500);
  Serial.println((char)26);
  delay(2000);
}
```

Figure 3.3: **Code for message sending and Server communication**

3.3 Timeline

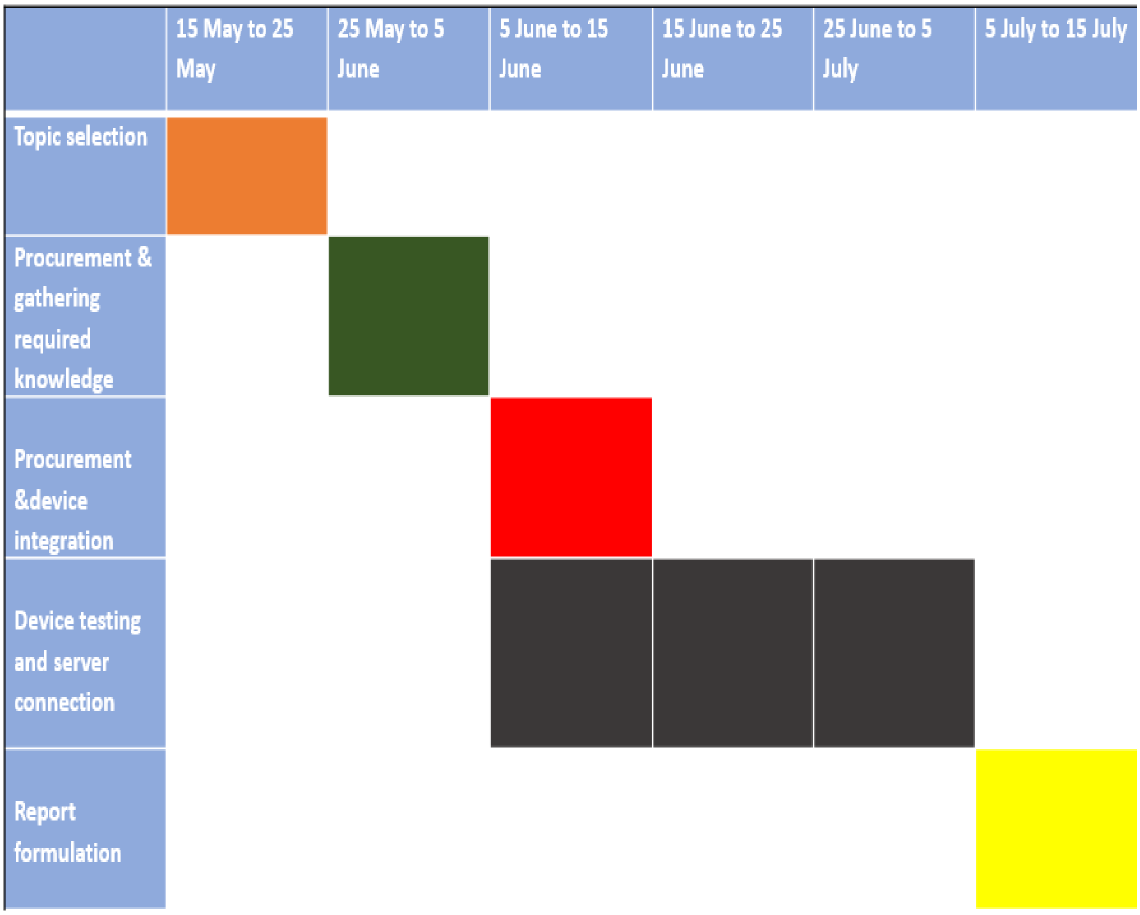

Figure 3.4: **Timeline**

3.4 Results

After completing of the assembling of hardware and implementation of the coding the system was linked to the web server and system testing was done at different time and location.

Table 3.1: **Gas values extracted from the server**

Human readable date	demo_gas
2019-07-15 15:29:00	5.57
2019-07-15 15:27:32	5.77
2019-07-15 14:10:51	10.75
2019-07-05 16:11:49	4.01
2019-07-05 16:10:29	4.01
2019-07-05 16:09:10	4.11
2019-07-05 16:07:50	4.11
2019-07-05 16:06:31	4.11
2019-07-05 15:27:00	4.99
2019-07-05 15:25:41	5.28
2019-07-05 15:24:21	5.77
2019-07-05 15:23:02	23.75
2019-07-05 15:19:30	15.64
2019-07-05 15:18:07	4.4
2019-07-05 15:13:33	4.4
2019-07-05 15:12:13	4.4
2019-07-05 14:51:14	4.99
2019-07-05 14:49:55	5.38
2019-07-01 20:07:33	15.84
2019-07-01 19:16:56	3.32
2019-07-01 19:15:45	3.32
2019-07-01 19:13:14	3.32
2019-06-15 19:51:38	6.26
2019-06-15 19:50:23	6.45
2019-06-15 19:49:08	6.65

Table 3.2: **PM2.5 values extracted from the server**

Human readable date	demo_pm2.5
2019-07-05 15:30:59	11.0
2019-07-05 15:29:39	11.0
2019-07-05 15:28:20	11.0
2019-07-05 15:27:00	11.0
2019-07-05 15:25:41	11.0
2019-07-05 15:24:21	11.0
2019-07-05 15:23:02	11.0
2019-07-05 15:19:30	11.0
2019-07-05 15:18:07	11.0
2019-07-05 15:13:33	16.0
2019-07-05 15:12:13	16.0
2019-07-05 15:10:53	16.0
2019-07-05 15:08:33	21.0
2019-07-05 14:59:40	0.0
2019-07-05 14:58:21	0.0
2019-07-05 14:52:34	0.0
2019-07-05 14:51:14	0.0
2019-07-05 14:49:55	0.0
2019-07-01 20:07:33	43.0
2019-07-01 20:05:38	51.0
2019-07-01 19:22:00	31.0
2019-07-01 19:18:10	37.0
2019-07-01 19:16:56	31.0
2019-07-01 19:15:45	33.0
2019-07-01 19:13:14	38.0
2019-06-15 19:51:38	64.0
2019-06-15 19:50:23	69.0
2019-06-15 19:49:08	63.0

CHAPTER 4

Conclusion and Future Scope

4.1 Conlusion

In this project an IoT based air pollution monitoring system was implemented using the presence of harmful gases and particulate matter in the air as parameters. In this we tried to implement a device which can sense the not only the presence of harmful gases and particulate matter but also alert the user when it reaches threshold. The contribution of the project is as follows:
1. The proposed device is having multiple sensors that too for different types of pollutants present in the air which makes it an overall broad spectrum device as compared to the devices made earlier.
2. The real time data of the amount of pollutants present in the air can be seen from anywhere.
3. We get notified when the level of pollutants crosses the threshold level and the user gets a message so that he/she can take necessary actions.

4.2 Scope For Future Development:

This section tells us about the various aspects of the project, which when addressed can make this project very helpful to the society and humanity.

4.2.1 Linking of the device with air purifiers

This device can be very helpful when it is linked to the air purifier, as the working of the purifier i.e when it has to stop – when it has to start, what type of gas has to be treated etc, can all be made dynamic and much more effective.

4.2.2 Scalability

This project is implement on a small scale and can be extended on large scale.

4.2.3 Interlinking

Multiple such devices can be build and can be fixed at different places and can be linked with each other, which will make it more efficient, increase its spectrum and make it more reliable in case of individual device failure.

REFERENCES

[1] Acharjya, D. P., Geetha, M. K., and Sanyal, S. (Eds.), Internet of Things: novel advances and envisioned applications, Vol. 25, Heidelberg: Springer, 2017.

[2] Unenvironment, “URL:https://www.unenvironment.org/news-and-stories/press-release/premature-deaths-environmental-degradation-threat-global-public” [Accessed on 1 June 2019]

[3] intinternetofthings, “URL:https://www.itu.intlinternetofthings.com”

[4] "IoT-Converging Technologies for smart Environments and Integrated Ecosystems", Ovidiu Vermesan and Peter Fiess., International Conference on Industrial Instrumentation and Control (ICIC), 2015.

[5] Andrea Zanella, Nicola Bui, "Internet of Things for smart cities", IEEE Internet of Things Journal, vol. 1, no. 1, 2014.

[6] Schmidt, M. (2015). Arduino: a quick-start guide. Pragmatic Bookshelf.

[7] Basics of UART Communication, "URL:http://www.circuitbasics.com/basics-uart-communication/" [Accessed on 1 June 2019]

[8] Microchip,"URL:https://www.microchip.com/wwwproducts/en/ATmega8A" [Accessed on 7 June 2019]

[9] Rhydolabz,"URL:https://www.rhydolabz.com/wireless-gsm-gprs-c-130185/sim-800-module-p-2055.html" [Accessed on 10 June 2019]

[10] Rhydolabz, "URL:https://www.rhydolabz.com/sensors-gas-sensors-c-137140/air-quality-sensor-mq135-p-1115.html" [Accessed on 10 June 2019]

[11] learnadafruit,"URL:https://learn.adafruit.com/pm25-air-quality-sensor"[Accessed on 10 June 2019]