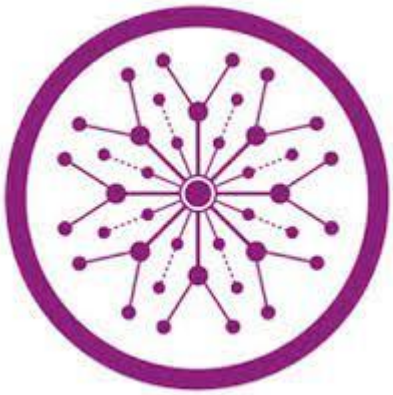
Air Quality Sensor Using Arduino

Final Year Project

Session 2019-2021

A project submitted in partial fulfillment of the degree of

Master in Information Technology



Department of Information Technology

Faculty of Computer Science & Information Technology

The Superior University Lahore

FALL 2021

Type (Nature of project)		[ ] **D**evelopment [ ] **R**esearch [ ] **R&D**		
Area of specialization				
FYP ID		FYP-MCSM-F20-033		
Final Year Project				
Sr.#	**Reg. #**	**Student Name**	**Email ID**	***Signature**
(i)	Mitm-f19-061	Ayesha Naseer	Mitm-f19-061@superior.edu,pk	

*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to work of others

Plagiarism Free Certificate

This is to certify that, I **Ayesha Naseer** S/D of **Muhammad Naseer**, group leader of FYP under registration no **Mitm-f19-061** **at** Information Technology Department, The Superior University Lahore. I declare that my FYP report is checked by my supervisor.

Date: Name of Group Leader: **Ayesha Naseer** Signature: _____

Name of Supervisor: Dr. Nadeem Jabbar

Designation: Senior Lecturer

Signature: _____

HoD: Dr. Asad Ali Naqvi

Signature: _____

Project Report

Air Quality Sensor Using Arduino

Change Record

Author(s)	Version	Date	Notes	Supervisor's Signature
Ayesha Naseer	1.0		Original Draft	
Ayesha Naseer	2.0		Changes Based on Feedback from Supervisor	
Ayesha Naseer	3.0		Changes Based on Feedback From Faculty	
Ayesha Naseer	4.0		Added Project Plan	
Ayesha Naseer	5.0		Changes Based on Feedback from Supervisor	

APPROVAL

PROJECT SUPERVISOR

Comments: _____

_____

Name:_____

Date:_____ Signature:_____

PROJECT MANAGER

Comments: _____

_____

Date:_____ Signature:_____

HEAD OF THE DEPARTMENT

Comments: _____

_____

Date:_____ Signature:_____

Dedication

First, We thank God for his blessing and to reconcile and repaid during these years that have passed us to get to this stage of science, knowledge and culture. We thank our respected teachers and parents on a continuous effort and giving and permanent support to reach achievement.

Acknowledgements

I would like to express my special thanks of gratitude to my teacher Nadeem Ch as well as our manager Javeed Iqbal who gave me the golden opportunity to do this wonderful project on the topic Air Quality Sensor Using Arduino, which also helped me in doing a lot of Research and I came to know about so many new things I am really thankful to them.

Secondly I would also like to thank my parents and friends who helped me a lot in finalizing this project within the limited time frame.

Executive Summary

This project is about how to make an air pollution monitoring system using an Arduino UNO, buzzer, LEDs and MQ135 gas sensor. When we keep some fuels which have harmful gases like CO2, NO2 etc. (when we keep an incense stick close to this sensor), the RED LED will glow and the buzzer starts ringing else if it is good quality air around, then green LED will glow. Sensor was giving us value of 90 when there was no gas near it and the safe level of air quality is 350 PPM and it should not exceed 1000 PPM. When it exceeds the limit of 1000 PPM, then it starts cause Headaches, sleepiness and stagnant, stale, stuffy air and if exceeds beyond 2000 PPM then it can cause increased heart rate and many other diseases. When the value will be less than 1000 PPM, then the LCD and webpage will display “Fresh Air”. Whenever the value will increase 1000 PPM, then the buzzer will start beeping and the LCD and webpage will display “Poor Air, Open Windows”. If it will increase 2000 then the buzzer will keep beeping and the LCD and webpage will display “Danger! Move to fresh Air”.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Chapter 1: Introduction

We are going to make an Air quality sensor using Arduino in which we will monitor the Air Quality over a webserver using internet and will trigger an alarm when the air quality goes down beyond a certain level, means when there are sufficient amount of harmful gases are present in the air like CO2, smoke, benzene and NH3. It will show the air quality in parts per million(PPM) on the LCD and as well as on webpage so that we can monitor it very easily and also display message like “Fresh air” etc . We will use MQ135 sensor which is the best choice for monitoring Air Quality as it can detects most harmful gases and can measure their amount accurately.

1.1. Background

Drager develops first portable tube for detecting carbon monoxide for the mining industry. The tubes were glass vials filled with a chemical reagent that reacted to a specific chemical or family of chemicals. A sample of air drawn into the tube changed the tube's color when the targeted chemical(s) were detected. Canaries in coal mines provided advanced warning of toxic gases. These living, mobile, handheld sensors saved countless lives of miners by detecting high concentrations of carbon dioxide, carbon monoxide, and methane. These small birds played an important part of mining safety and retired from mining the mid-1980s.

1.2. Motivations and Challenges

Air pollution index(API) is established to provide the public with information that is easy to understand about the air pollution level. API is calculated based on the new Ambient Air Quality . It is developed to follow the Pollutant Standard Index (PSI) system that has been regulated by United States Environmental Protection Agency (USEPA). The new standard added one new pollutant which is particulate matter with the size of less than 2.5µm (PM2.5) to the existing pollutants which are Carbon Monoxide (CO), Ozone (O3), Nitrogen Dioxide (NO2), Sulfur Dioxide (SO2) and particulate matter with the size of less than 10µm (PM10). Although PM2.5 is included in the new Ambient Air Quality Standards, the existing monitoring stations did not have the equipment to measure the pollutant yet.

1.3. Goals and Objectives

There is increasing public awareness of the real time air quality due to air pollution can cause severe effects to human health and environments. The Air Pollutant Index (API) is measured by Department of Environment (DOE) using stationary and expensive monitoring station called Continuous Air Quality Monitoring stations (CAQMs) that are only placed in areas that have high population densities and high industrial activities. particulate matter with the size of less than 2.5µm (PM2.5) in the API measurement system. we present a cost effective and portable air quality measurement system using Arduino Uno microcontroller and four low cost sensors. This device allows people to measure API in any place they want. It is capable to measure the concentration of carbon monoxide (CO), ground level ozone (O3) and particulate matters (PM10 & PM2.5) in the air and convert the readings to API value.

1.4. Literature Review/Existing Solutions

This project is about how to make an air pollution monitoring system using an Arduino UNO, buzzer, LEDs and MQ135 gas sensor. When we keep some fuels which have harmful gases like CO2, NO2 etc. (when we keep an incense stick close to this sensor), the RED LED will glow and the buzzer starts ringing else if it is good quality air around, then green LED will glow.

1.5. Gap Analysis

The combination of Internet of Things and Ambient-Assisted Living technologies creates a consistent approach for the development of Enhanced Living Environments to improve productivity in the day to day activities, as well as for the overall well-being. With new technologies, it is possible to design intelligent cyber-physical systems for easy installation at regular buildings. In order to improve the quality of occupational health, it is essential to monitor the majority of living environments on a real-time basis. It can be done by utilizing the latest approaches and technologies for data access. Alerts can be sent to the users to take appropriate steps for handling poor ambient air quality. It is essential to mention that a large population in the developing countries spends most of their time indoors. In such

situations, indoor air quality(IAQ) monitoring plays anessential role in creating Enhanced Living Environments with better occupational health. Moreover, the world population is aging with time. It is now essential to make efforts to improve the quality of life for the elderly while minimizing the cost of caregivers.

1.6. Proposed Solution

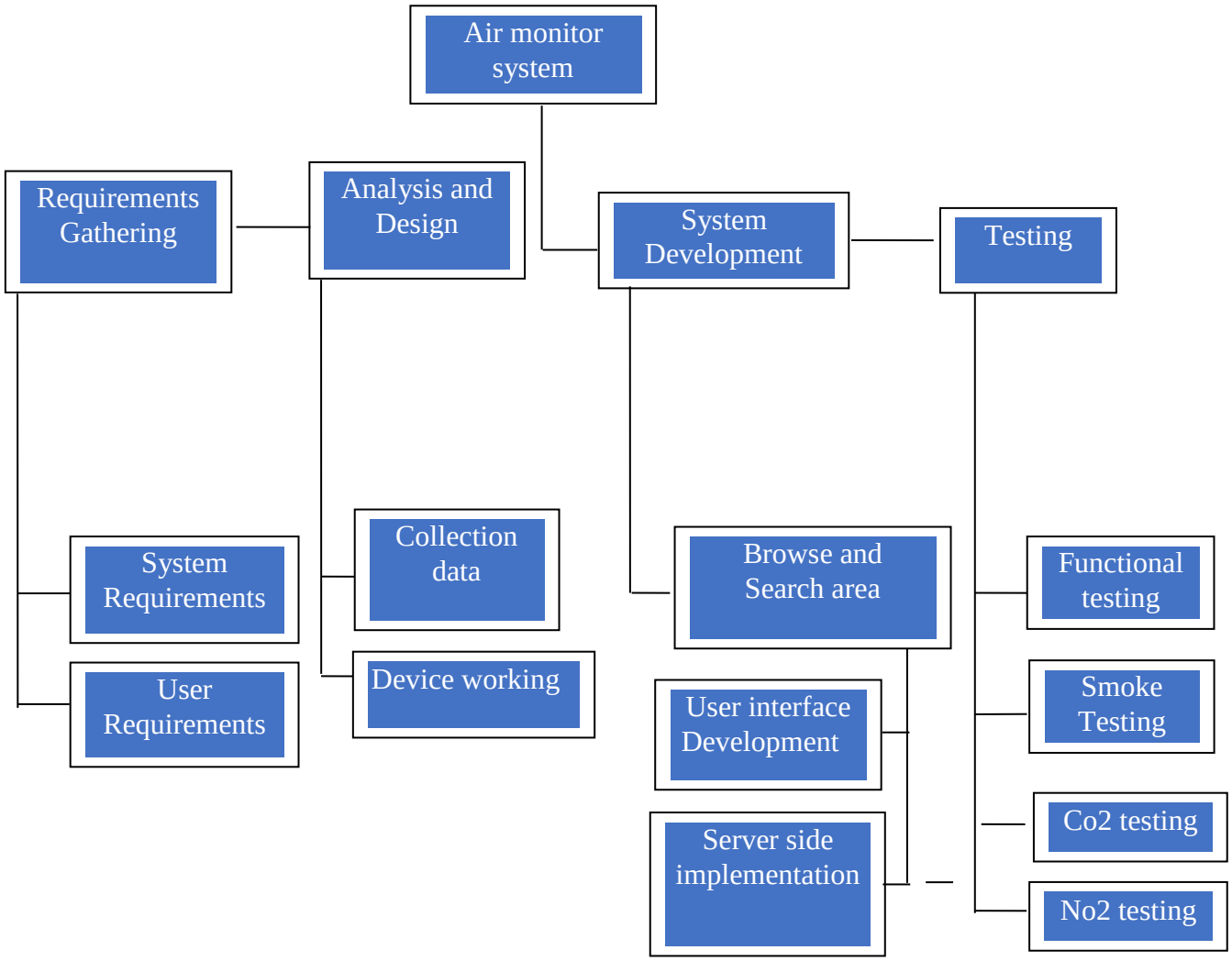
This project is aimed at developing an IOT device which can monitor air pollution in real time and log data to a remote server. Remote monitoring was facilitated using classical motes in the past, which has some pitfalls like limited memory, processing speed and complex programming strategies. By using Internet of Things and recording sensor data to a remote server, the limitations of memory in the monitoring devices and manual collection of data from the installed devices can be overcome. The IOT also helps monitoring the data in real time. The air pollution monitoring device developed in this project is based Arduino UNO. The Arduino board connects with Thing Speak platform using ESP8266 Wi-Fi Module(The ESP8266 is a really useful, cheap WiFi module for controlling devices over the Internet. It can work with a micro-controller like the Arduino or it can be programmed to work on its own. We can use other module such as ptcl modem etc but it is more expensive because more wire used in this module). As the cities usually have Wi-Fi hotspots at most of the places, so the device can be easily installed near any hotspot for its operation. The Thing Speak is a popular IOT platform which is easy to use and program. The sensor used for monitoring the air pollution is MQ-135(MQ135 gas sensor has high sensitivity to Ammonia, Sulfide and Benze steam, also sensitive to smoke and other harmful gases. It is with low cost and particularly suitable for Air quality monitoring application) gas sensor. The sensor data is also displayed on a character LCD interfaced in the monitoring IOT device.

The sensing of data and sending it to the Thing Speak server using Wi-Fi module is managed by the Arduino Sketch. The Arduino sketch is written, compiled and loaded to the Arduino board using Arduino IDE.

1.7. Project Plan

We are going to make an Air quality sensor using Arduino in which we will monitor the Air Quality over a webserver using internet and will trigger an alarm when the air quality goes down beyond a certain level, means when there are sufficient amount of harmful gases are present in the air like CO2, smoke, alcohol, benzene and NH3. It will show the air quality in PPM on the LCD and as well as on webpage so that we can monitor it very easily. We will use MQ135 sensor which is the best choice for monitoring Air Quality as it can detects most harmful gases and can measure their amount accurately. Connect the VCC and the ground pin of the sensor to the 5V and ground of the Arduino and the Analog pin of sensor to the A0 of the Arduino. Connect a buzzer to the pin 8 of the Arduino which will start to beep when the condition becomes true. Connect pin 1 (VEE) to the ground. Connect pin 2 (VDD or VCC) to the 5V.

1.7.1. Work Breakdown Structure

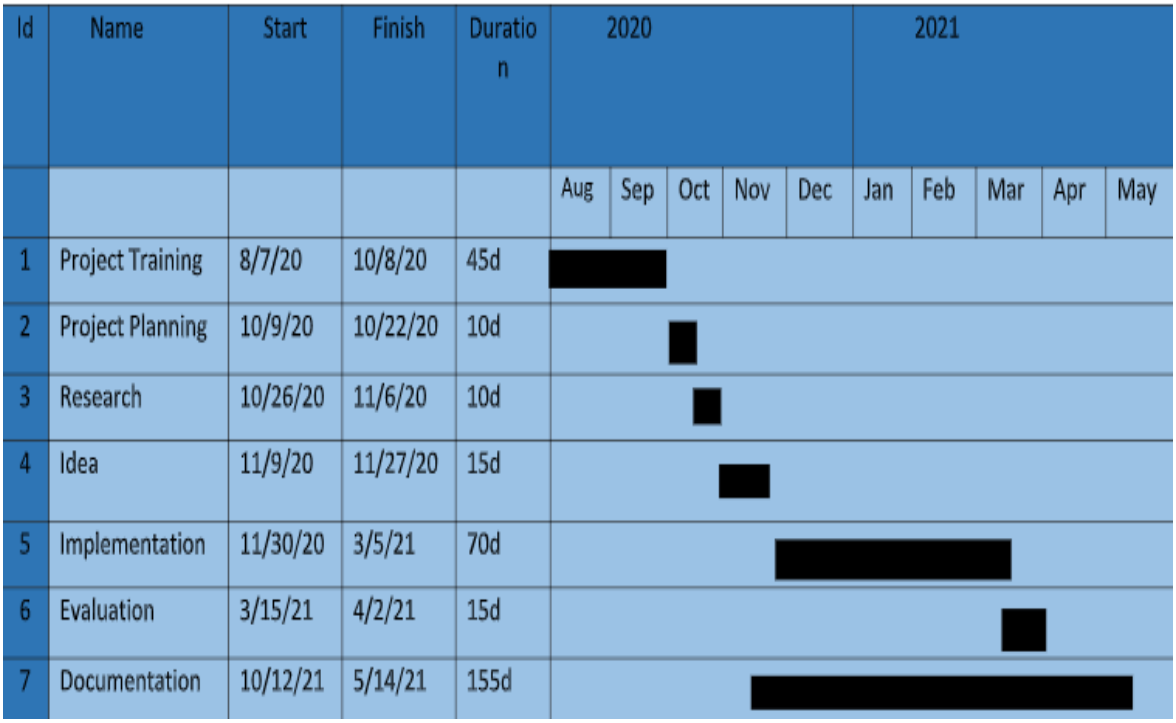


1.7.2. Roles & Responsibility Matrix

WBS #	WBS Deliverable	Activity #	Activity to Complete the Deliverable	Duration (# of Days)	Responsible Team Member(s) & Role(s)
1	Project Training	1	Training for the gain of skills in Air Quality Sensor tool "**Visual Studio**"	40	Ayesha Naseer
2	Project Planning	1 2 3	Setting tasks between the team members Creating WBS & Gantt Chart to manage time efficiently Documenting Project proposal	10	Ayesha Naseer
3	Research	1 2 3	Research Air Quality sensor market thoroughly and scope Research on trending Sensor Mechanic Research	7	Ayesha Naseer
4	Pollution control Generation	1 2 3	Conduct control pollution ideas generation from the research gathered Narrowing the ideas list Selecting the most suitable pollution control idea	7	Ayesha Naseer

5	Development	1 2 3 4 5	Prototyping Coding and documentation Testing and documentation Final Outcome Last check and documentation	55	Ayesha Naseer
6	Evaluatin	1 2 3 4	Evaluating the pollution control device Evaluating the values received by the sensor Evaluating the pollution control development tool (Visual studio or Dev c++) Evaluating the project as whole	10	Ayesha Naseer
7	Documentation	1	Formalizing the documentation	7	Ayesha Naseer

1.7.3. Gantt Chart



Id	Name	Start	Finish	Duration
1	Project Training	8/7/20	10/8/20	45d
2	Project Planning	10/9/20	10/22/20	10d
3	Research	10/26/20	11/6/20	10d
4	Idea	11/9/20	11/27/20	15d
5	Implementation	11/30/20	3/5/21	70d
6	Evaluation	3/15/21	4/2/21	15d
7	Documentation	10/12/21	5/14/21	155d

1.8. Report Outline

Introduction

Background

Motivations and Challenges

Goals and Objectives

Literature Review/Existing Solutions

Gap Analysis

Proposed Solution

Project Plan

Work Breakdown Structure

Roles & Responsibility Matrix

Gantt Chart

Chapter 2

Software Requirement Specifications

Chapter 2: Software Requirement Specifications

2.1. Introduction

2.1.1. Purpose

This project is aimed at developing an IOT device which can monitor air pollution in real time and log data to a remote server. Remote monitoring was facilitated using classical motes in the past, which has some pitfalls like limited memory, processing speed and complex programming strategies. By using Internet of Things and recording sensor data to a remote server, the limitations of memory in the monitoring devices and manual collection of data from the installed devices can be overcome. The IOT also helps monitoring the data in real time. The air pollution monitoring device developed in this project is based Arduino UNO. The Arduino board connects with Thing Speak platform using ESP8266 Wi-Fi Module. As the cities usually have Wi-Fi hotspots at most of the places, so the device can be easily installed near any hotspot for its operation. The Thing Speak is a popular IOT platform which is easy to use and program. The sensor used for monitoring the air pollution is MQ-135 gas sensor. The sensor data is also displayed on a character LCD interfaced in the monitoring IOT device.

The sensing of data and sending it to the Thing Speak server using Wi-Fi module is managed by the Arduino Sketch. The Arduino sketch is written, compiled and loaded to the Arduino board using Arduino IDE.

2.1.2. Document Conventions

It will be explained what we have realized with respect to the requirements, the problems found during the development of the system, what we have learned from the realization of the project and the possible improvements to the air quality sensor using arduino. Analyzed various types' research methodologies as well as software development methodologies. Build methodology was chosen as the suitable research methodology. Also provided justification for the choice of RAD as our software development methodology and provided reasons as to why other different

methodologies were not chosen. Tools that are to be used in the development of the Air quality sensor using Arduino were also discussed.

2.1.3. Intended Audience and Reading Suggestions

Following are the values that will help user to understand good or bad environment of his surrounding without reading SMS.

From 0 to 50.....good

from 51 to 100.....moderate

From 101 to 200.....unhealthy

From 201 to 300.....very unhealthy

Above from 300.....hazardous

Above from 500.....Emergency

unit :ppm (parts per million)

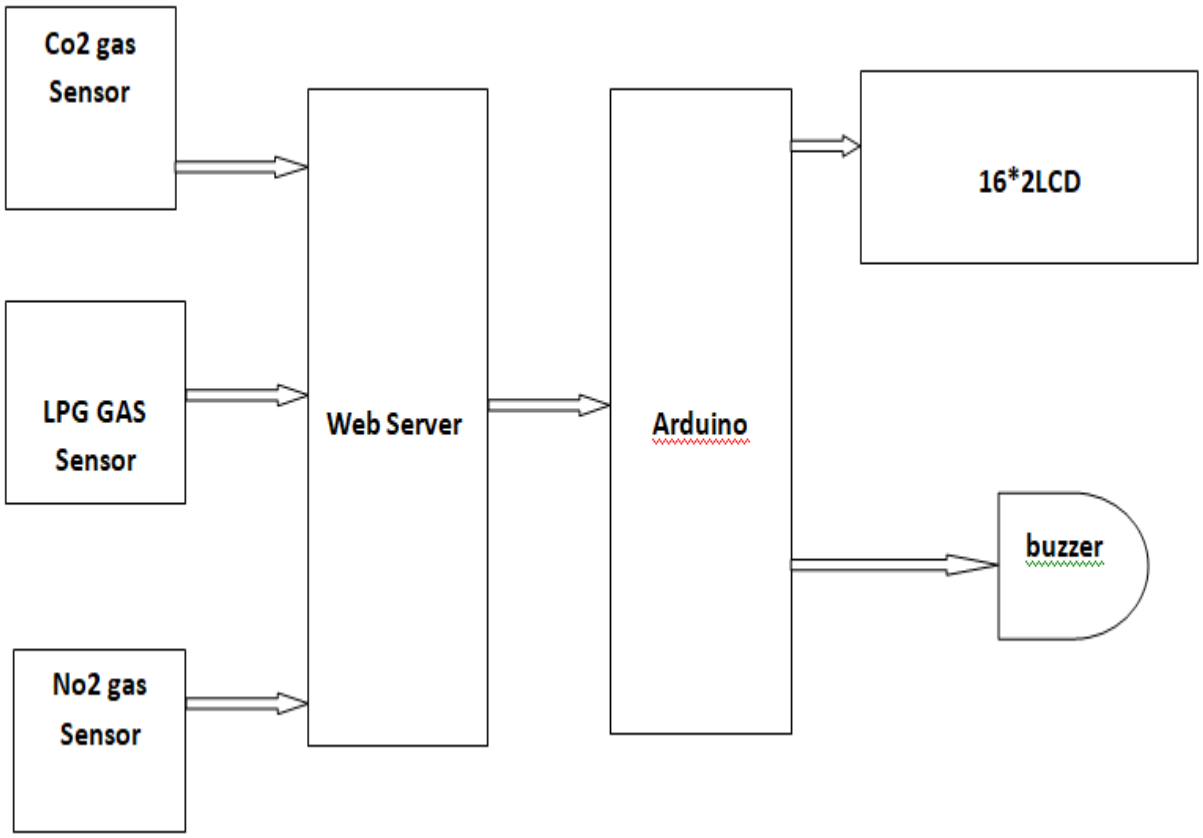
API	DESCRIPTOR
0-50	Good
51-100	Moderate
101-200	Unhealthy
201-300	Very unhealthy
>300	Hazardous
>500	Emergency

2.1.4. Product Scope

Sensor was giving us value of 90 when there was no gas near it and the safe level of air quality is 350 PPM and it should not exceed 1000 PPM. When it exceeds the limit of 1000 PPM, then it starts cause Headaches, sleepiness and stagnant, stale, stuffy air and if exceeds beyond 2000 PPM then it can cause increased heart rate and many other diseases. When the value will be less than 1000 PPM, then the LCD and webpage will display “Fresh Air”. Whenever the value will increase 1000 PPM, then the buzzer will start beeping and the LCD and webpage will display “Poor Air, Open Windows”. If it will increase 2000 then the buzzer will keep beeping and the LCD and webpage will display “Danger! Move to fresh Air”.

2.2. Overall Description

2.2.1. Product Perspective



2.2.2. User Classes and Characteristics

Air pollution sensors are devices that monitor the presence of air pollution in the surrounding area. They can be used for both indoor and outdoor environments.The EPA maintains a repository of air quality data through the Air Quality System (AQS).

2.2.3. Operating Environment

User can monitor the air quality at any where in the specific area like Lahore or any other. The sensor can be calibrated so that its analog output voltage is proportional to the concentration of polluting gases in PPM. The analog voltage sensed at the pin A0 of the Arduino is converted to a digital value by using the in-built ADC channel of the Arduino. The Arduino board has 10-bit ADC channels, so the digitized value ranges from 0 to 1023. The digitized value can be assumed proportional to the concentration of gases in PPM. The read value is first displayed on LCD screen and passed to the ESP8266 module wrapped in proper string through virtual serial function. The Wi-Fi module is configured to connect with the ThingSpeak IOT platform. ThingSpeak is an IOT analytics platform service that allows to aggregate, visualize and analyze live data streams in the cloud. ThingSpeak provides instant visualizations of data posted by the IOT devices to ThingSpeak server.

2.2.4. Design and Implementation Constraints

The air quality measurement system is designed using low cost sensors and microcontroller. It is compact and portable making it easy to measure air quality in any place. The air quality sensor using Arduino is developed in small programs called units, which are integrated in the next phase. Each and every one of these unit is developed and tested for its functionality which is known as Unit Testing.We have used Arduino UNO, MQ-135 air quality sensor, LCD display, breadboard, jumper wires, and potentiometer to develop an Arduino based air pollution detector which combined a small-sized, minimum-cost sensor to an Arduino microcontroller unit . The device is linked to a computer through a serial connection. From the sensor, the collected data through the Arduino microcontroller.

2.2.5. Assumptions and Dependencies

This phase tries to capture all the requirements of the system to be developed and document all of this in the requirements specification document. In the air quality sensor using Arduino this involves gathering requirements from the users .

The requirement specifications that are acquired from the first phase are studied in this phase and system design is carried out. System design phase describe the hardware to be used which in this project involves the use of laptops and a server and as an alternative the project makes use of a localhost server XAMPP. XAMPP can be hosted on a laptop and used to store the data. System Design assists in specifying system requirements and also helps in defining overall system architecture.

2.3. External Interface Requirements

2.3.1. User Interfaces

User can check value of its surrounding on the LCD of their device. User can also check lights (red or green) .Due to the complexity of factors determining the environmental quality, large variations are found between industries, residential and industrial areas on different location of a city or in different hydro and geoclimatic zones. The use of monitoring has evolved to determine trends in the quality of the aquatic, terrestrial and atmospheric environment and how they are affected by the release of contaminants, other anthropogenic activities or by waste treatment operations. As pollutants vary for each geographic area.

2.3.2. Hardware Interfaces

Air Quality Sensor using Arduino(Device) will show the air quality in PPM on the LCD and as well as on webpage so that we can monitor it very easily. MQ135 sensor which is the best choice for monitoring Air Quality as it can detects most harmful gases and can measure their amount accurately.Connect the VCC and the ground pin of the sensor to the 5V and ground of the Arduino and the Analog pin of sensor to the A0 of the Arduino. Connect a buzzer to the pin 8 of the Arduino which will start to beep when the condition becomes true. Connect pin 1 (VEE) to the ground. Connect pin 2 (VDD or VCC) to the 5V.

2.3.3. Software Interfaces

We have used Arduino UNO, MQ-135 air quality sensor, LCD display, breadboard, jumper wires, and potentiometer to develop an Arduino based air pollution detector which combined a small-sized, minimum-cost sensor to an Arduino microcontroller unit . The

device is linked to a computer through a serial connection. From the sensor, the collected data through the Arduino microcontroller. We will use SDK manager and HTML for design. The basic need of this project is arduino, wires, sensors.

2.3.4. Communications Interfaces

This project is aimed at developing an IOT device which can monitor air pollution in real time and log data to a remote server. Remote monitoring was facilitated using classical motes in the past, which has some pitfalls like limited memory, processing speed and complex programming strategies. By using Internet of Things and recording sensor data to a remote server, the limitations of memory in the monitoring devices and manual collection of data from the installed devices can be overcome.

2.4. System Features

By using Internet of Things and recording sensor data to a remote server, the limitations of memory in the monitoring devices and manual collection of data from the installed devices can be overcome. The IOT also helps monitoring the data in real time. The air pollution monitoring device developed in this project is based Arduino UNO. The Arduino board connects with Thing Speak platform using ESP8266 Wi-Fi Module. As the cities usually have Wi-Fi hotspots at most of the places, so the device can be easily installed near any hotspot for its operation. The Thing Speak is a popular IOT platform which is easy to use and program. The sensor used for monitoring the air pollution is MQ-135 gas sensor. The sensor data is also displayed on a character LCD interfaced in the monitoring IOT device.

2.4.1. System Feature 1

Monitor the pollution of air and send the data to the web server by using iot devices.

2.4.1.1. Description and Priority

Air pollution is one the most crucial factors affecting life and health of human, animals and plants. Air quality, weather and climate, and human health are closely linked. These interdependencies are becoming ever more evident and health professionals ever more reliant on meteorological and climate services to help anticipate and manage the health risks of poor air quality. Over the last century, poor air quality has become a critical

environmental, economic, and health problem around the world as industrial growth and economic development have caused massive increases in air pollutants. In this paper, a wireless solution is proposed for monitoring the level of hydrogen sulfide gas (H 2 S). The proposed system enables measurement of the levels of H 2 S, temperature, and humidity. The software part of the project has been developed under LabVIEW environment. The experimental results demonstrate the efficacy of our project in terms of fast detection and real time response.

2.4.1.2. Stimulus/Response Sequences

This involves running the system and evaluating the defect that can arise and actions are carried out to correct the defects. The methods for testing the system involves unit testing which tests the different components of the system that is the system's interfaces, data storage and how the different users activities are being carried out.

After testing each of the unit in the implementation phase they are integrated into the system. Post integration the entire system is tested for any faults and failures.

2.4.1.3. Functional Requirements

System must support two way communications between the client and server. System must be compact. System should be easy to deploy.

REQ-SF1-1: System needs to store the data and provide access to a location map interface. System must support accurate and continuous real time data collection.

REQ-SF1-2: System needs to support mobility.

REQ-SF1-3: System must use minimum power

2.4.2. System Feature 2

Decrease all the aspects that cause difficulties for human life or other living things.

2.4.2.1. Description and Priority

Gas sensors can output a measurement of the gases detected in a number of ways. These include percent LEL, percent volume, trace, leakage, consumption, density, and signature or spectra. The lower explosive limit (LEL) or lower flammable limit (LFL) of a combustible gas is defined as the smallest amount of the gas that will support a self-propagating flame when mixed with air (or oxygen) and ignited. In gas-detection systems, the amount of gas present is specified in terms of % LEL: 0% LEL being a combustible gas-free atmosphere and 100%

LEL being an atmosphere in which the gas is at its lower flammable limit. The relationship between % LEL and % by volume differs from gas to gas.

2.4.2.2. Stimulus/Response Sequences

An air pollution monitoring system consisting of gas sensors by which the gas levels of system can be known. Here use carbon monoxide (MQ-7), LPG sensor (MQ6) to shows the pollution level, latitude, longitude and UTC. The unit can be placed on the top of any moving device such as a public transportation vehicle. While the vehicle is moving, the micro controller generates a frame consisting of a acquired air pollutant levels from the sensor array and the physical location that is reported from the attached GPS module. The pollutants frame is then uploaded to the general packet radio service modem (GPRS modem) and transmitted to the pollution server via the public mobile network.

2.4.2.3. Functional Requirements

There are many commercial air quality sensor products and monitoring systems available in the market but the price is expensive and not affordable to employ. They consist of many features, functions and high-end equipment with complex technologies. Special skills and knowledge are required for system's handling and at times it is hard to set up for those with no expertise. Besides, the physical size of the commercial air quality system is usually huge and heavy and some of the existing systems do not provide the notification alert and data storage for further analysis. Thus a device with small scale size is necessary to increase the portability aspects of the arrangement so that the device can be used anywhere and at any time. These research questions are mapped with the research objectives of this study which are to develop a wireless and affordable Internet of Things (IoT)-based device that can monitor the air quality which can affect human's health, to integrate the monitoring system with a cloud storage so that the information can be viewed online on a web, and generate an alert notification e-mail when the air quality is in unhealthy condition. The proposed solution intends to solve the highlighted issues by using IoT technology, wireless communication and cloud services.

REQ-SF2-1: System should be easy to deploy. system must be field configurable.

REQ-SF2-2: System must support two way communications between the client and server.

REQ-SF2-3: System must be compact.

2.4.3. System Feature 3 (and so on)

2.5. Nonfunctional Requirements

2.5.1. Performance Requirements

In user design phase, model and prototype that represent all system processes, inputs, and outputs are developed. User design is a continuous interactive process that allows users to understand and modify a working model of the system that meets their needs. In this study, the air quality monitoring system model and prototype were developed based on system operations, inputs and outputs. The design for the proposed system is illustrated. It demonstrates the integration of the sensors, LEDs, Internet and cloud storage with the processing platform i.e. Raspberry Pi 2 microprocessor.

2.5.2. Safety Requirements

The input of the system is the air quality sensor MQ-135, temperature and humidity sensor DHT-22. The MQ-135 sensor is used to sense the air quality which comprises of chemical substances such as ammonia (NH3), nitrogen oxide (NOX), alcohol, benzene and other harmful gases. The concentration scope of the detection for benzene vapour and ammonia (NH3) are 10ppm to 1000ppm and 10ppm to 300ppm respectively, whereas the DHT-22 is used to sense the temperature and humidity in the air in degree Celsius and percentage. The output of the system is a notification email alert and online data display and storage.

Security Requirements

The main programming language used was Python. It was for code development of the system. It was integrated with several modules to realize the system's functionalities. Other related software requirements were Hypertext Markup Language (HTML) and Hypertext Preprocessor (PHP) for web development that incorporated with open source ThingSpeak cloud storage. ThingSpeak requires an Application Programming Interface (API) in order to provide web services for connecting things or objects such as storing and retrieving data from the sensors using Hypertext Transfer Protocol (HTTP) at port 80 over the Internet. As for the alert notification part, the system communicated with Google e-mail server for e-mail service of personal account.

2.5.3. Software Quality Attributes

The system uses inexpensive sensors to measure CO2, dust and temperature. Each sensor is followed with electronics transducer in order to amplify and filter out the noise signals. After that, Arduino Microcontroller is used as data acquisition system to convert the analog signals into digital signals. In order to track the system in real time, a communication solution has been introduced using Arduino Ethernet shield. To display the readings of sensors remotely, Blynk mobile application has been used, where Blynk is a platform used to design a mobile application to interact with IoT devices.

2.5.4. Business Rules

The Clean Air Act requires every state to establish a network of air monitoring stations for criteria pollutants, using criteria set by OAQPS for their location and operation. The monitoring stations in this network are called the State and Local Air Monitoring Stations (SLAMS). The states must provide OAQPS with an annual summary of monitoring results at each SLAMS monitor, and detailed results must be available to OAQPS upon request. To obtain more timely and detailed information about air quality in strategic locations across the nation, OAQPS established an additional network of monitors: the National Air Monitoring Stations (NAMS). NAMS sites, which are part of the SLAMS network, must meet more stringent monitor siting, equipment type, and quality assurance criteria. NAMS monitors also must submit detailed quarterly and annual monitoring results to OAQPS.

2.6. Other Requirements

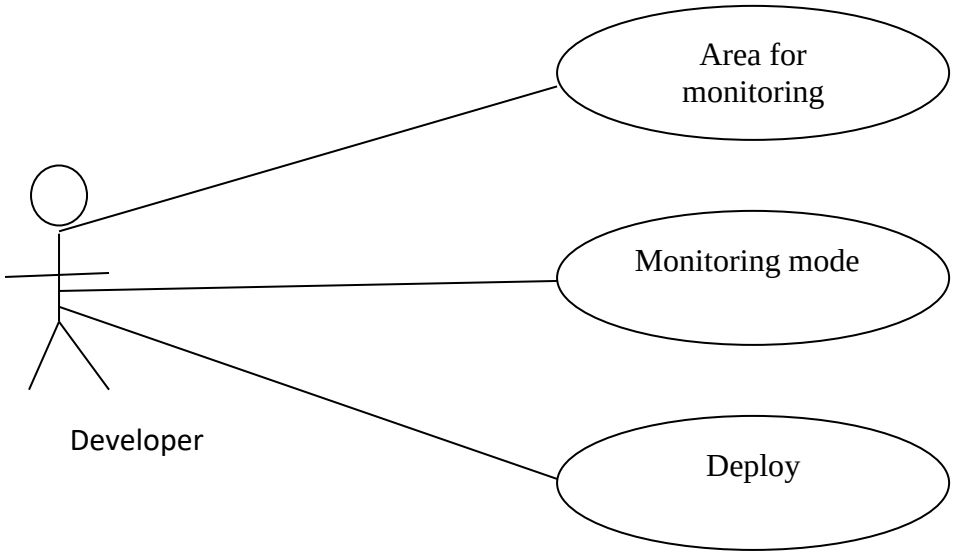
A wireless distributed mobile air pollution monitoring system was designed. The system utilizes city buses to collect pollutant gases such as CO, NO2 and SO2. The data shows wavelengths of interest and remove undesirable wavelengths from the beam. The NDIR technique is most sensitive and selective for small molecules whose spectral fine structure is resolved under ambient conditions. It is suitable for continuous monitoring of CO.

Chapter 3
Use Case Analysis

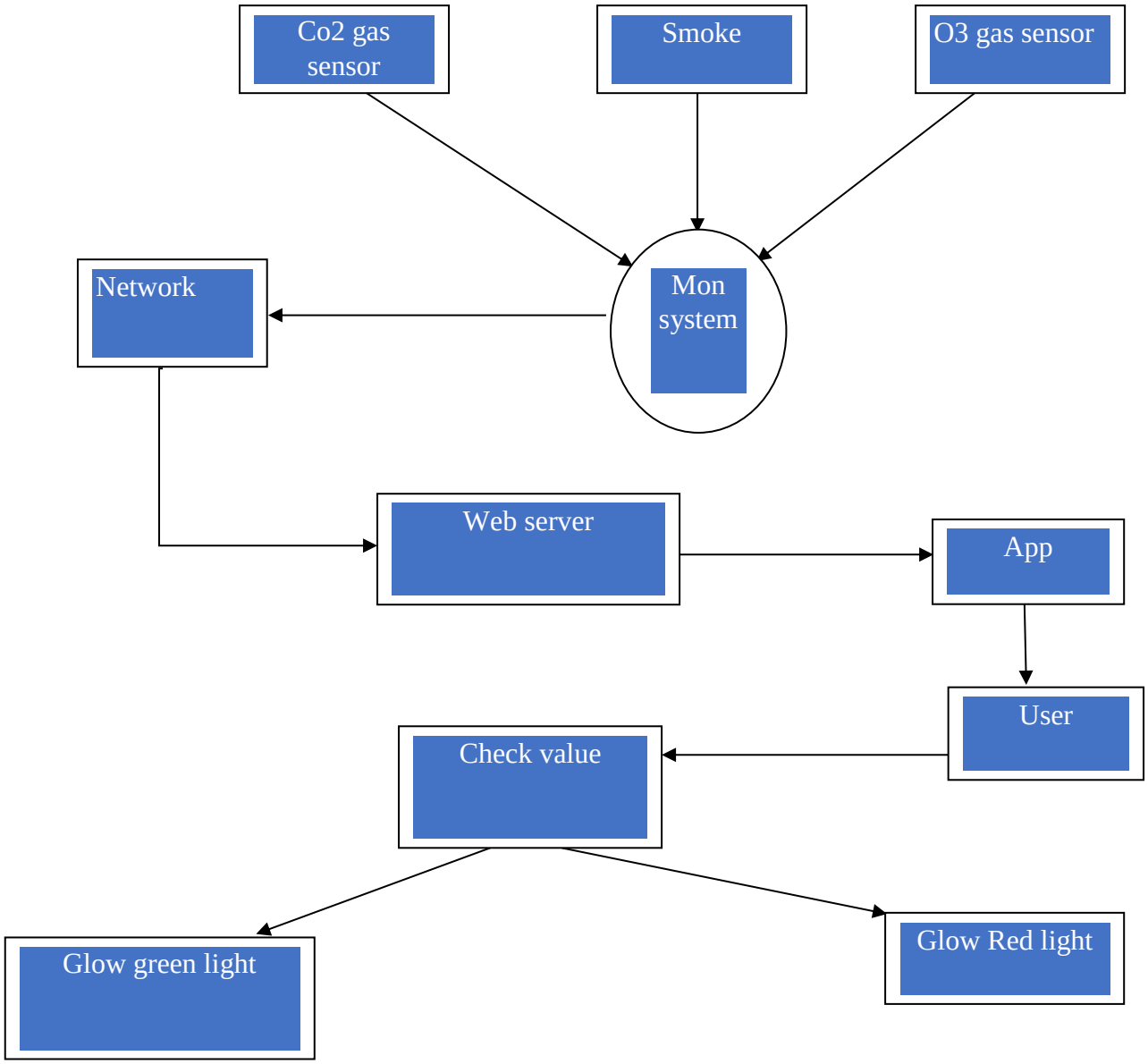
Chapter 3: Use Case Analysis

We are going to make an Air quality sensor using Arduino in which we will monitor the Air Quality over a webserver using internet and will trigger an alarm when the air quality goes down beyond a certain level, means when there are sufficient amount of harmful gases are present in the air like CO2, smoke, alcohol, benzene and NH3. It will show the air quality in PPM on the LCD and as well as on webpage so that we can monitor it very easily. We will use MQ135 sensor which is the best choice for monitoring Air Quality as it can detects most harmful gases and can measure their amount accurately.

3.1. Use Case Model



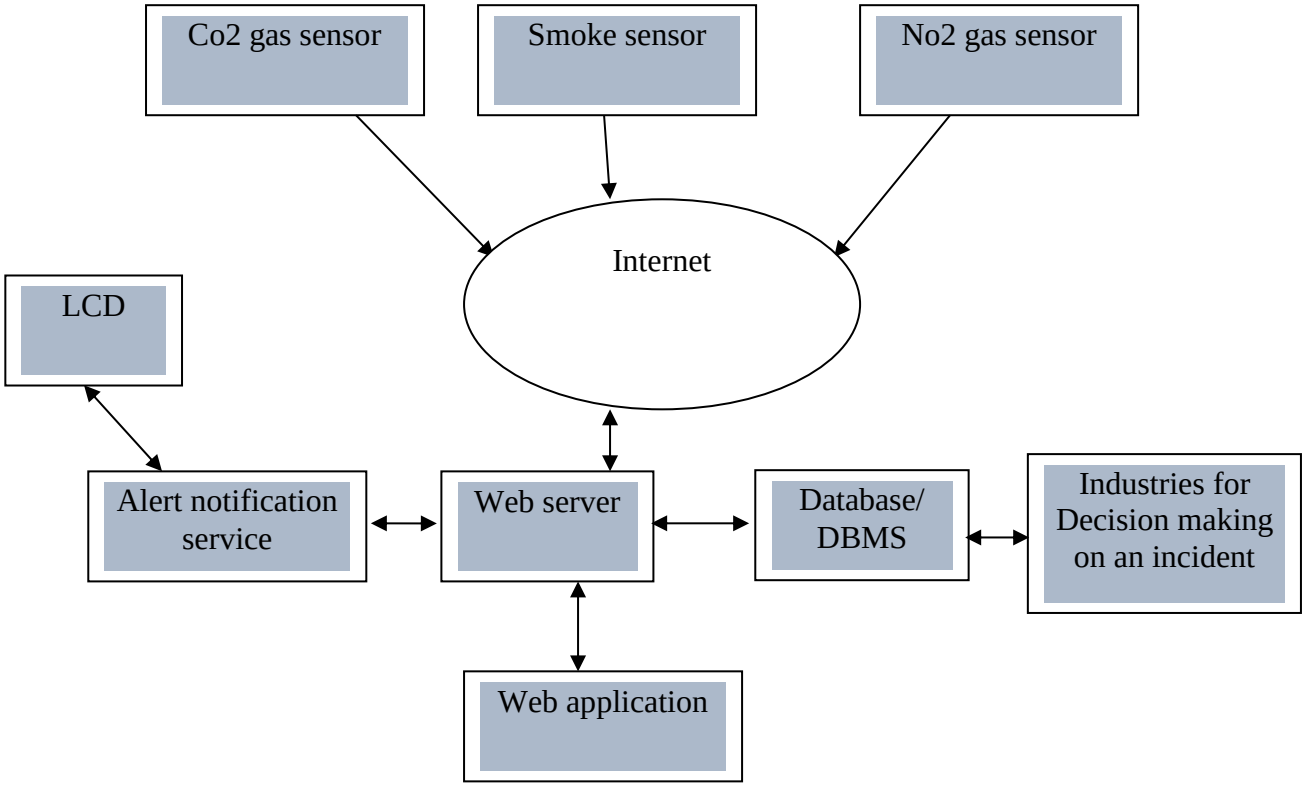
3.2. Fully Dressed Use Cases



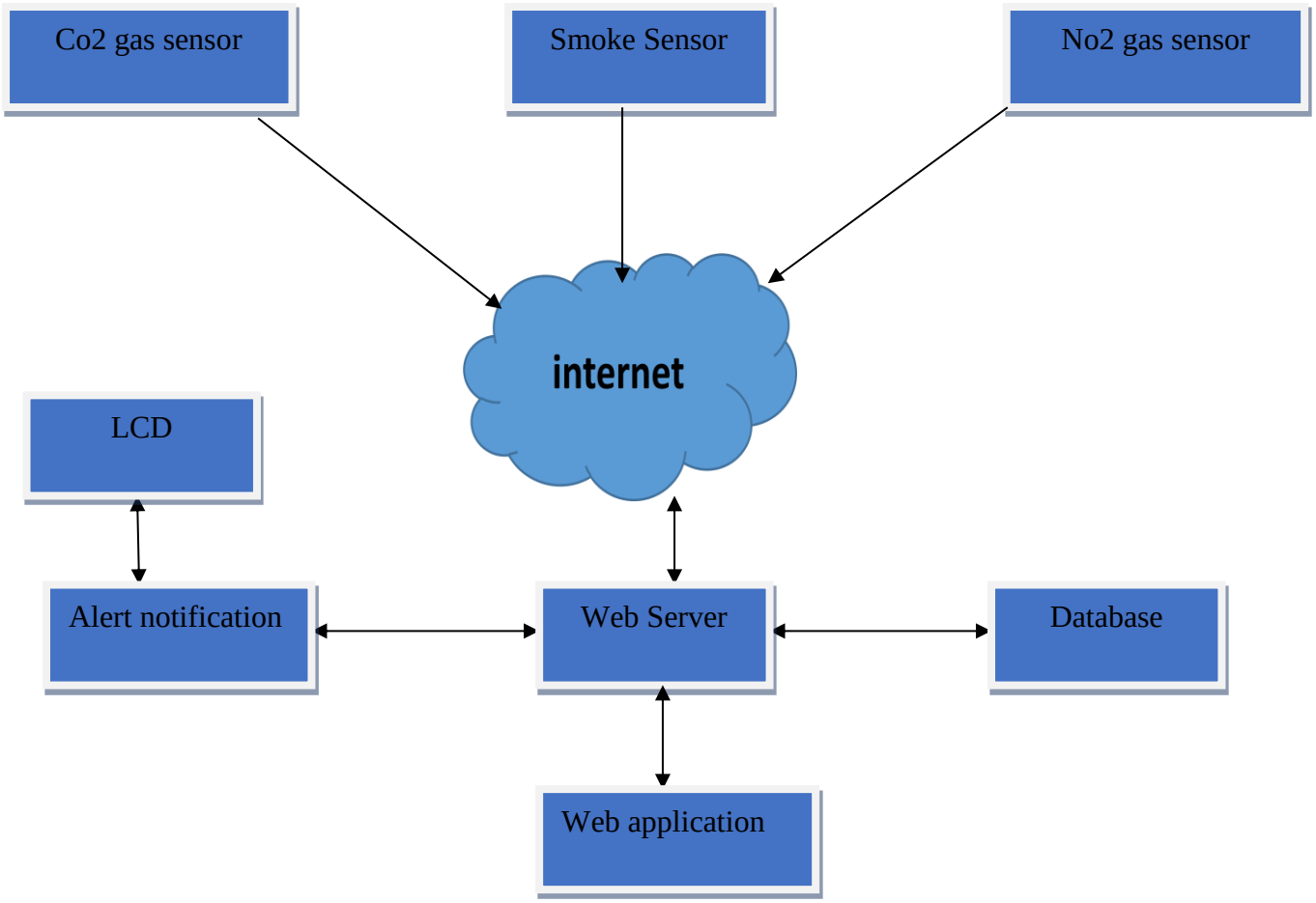
Chapter 4

System Design

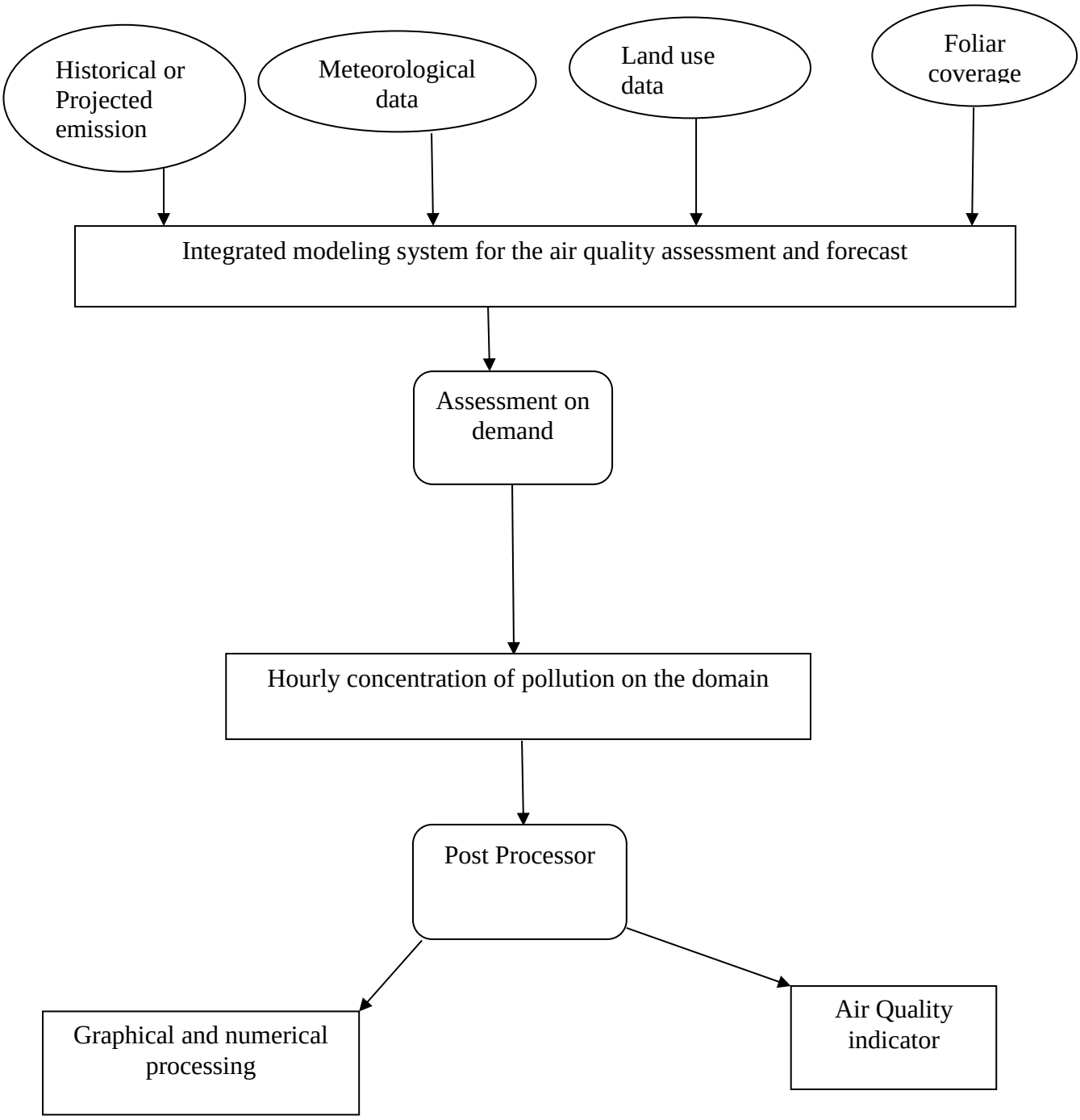
Chapter 4: System Design



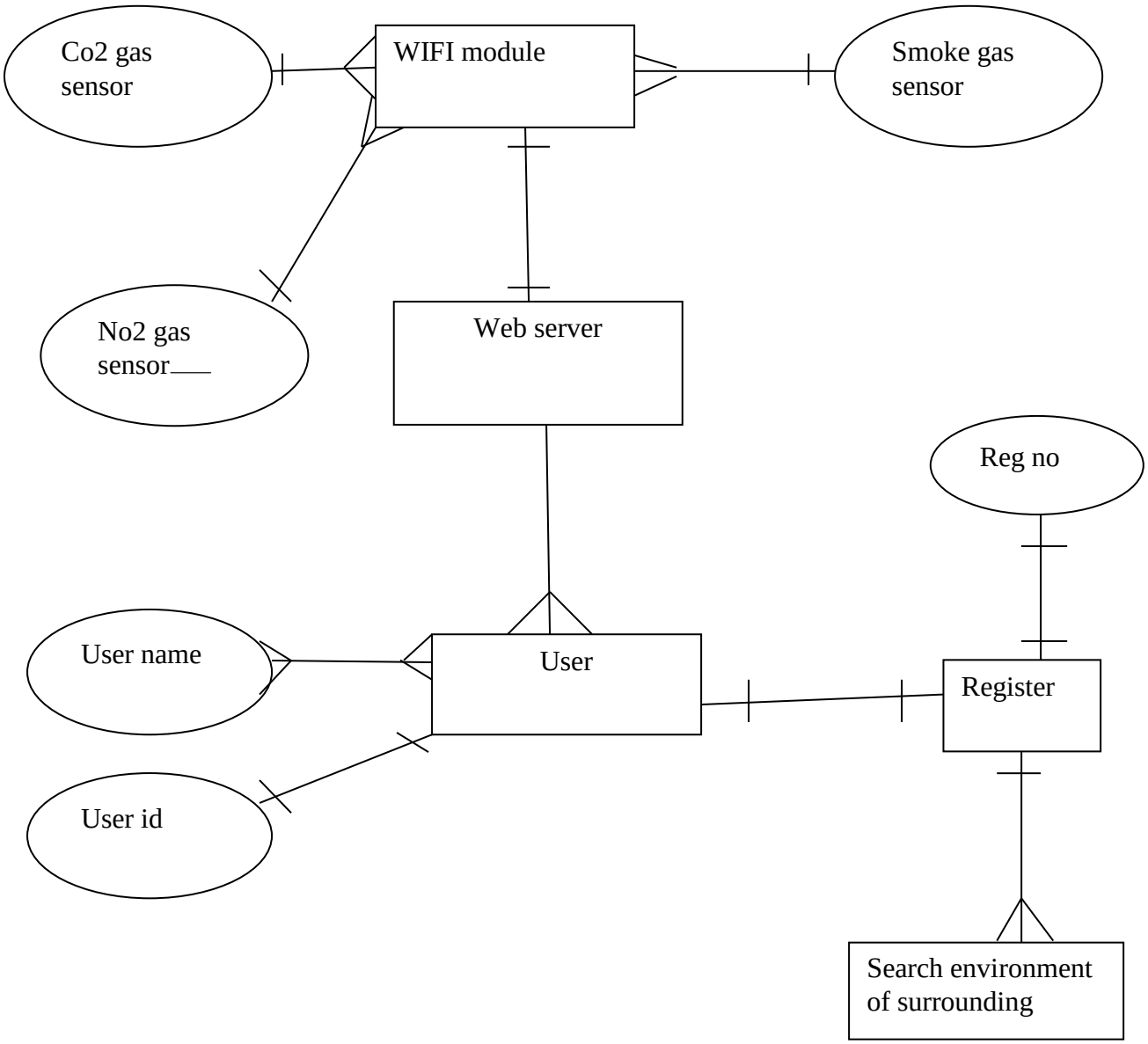
4.1. Architecture Diagram



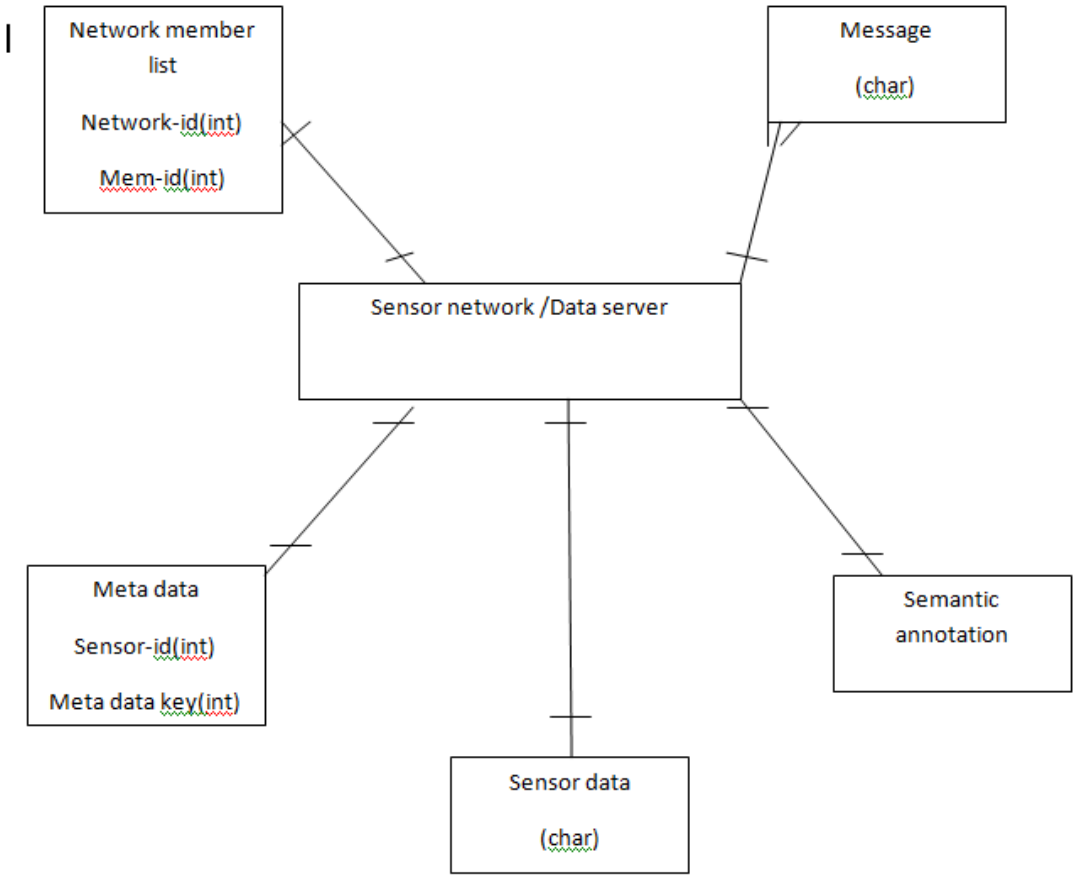
4.2. Domain Model



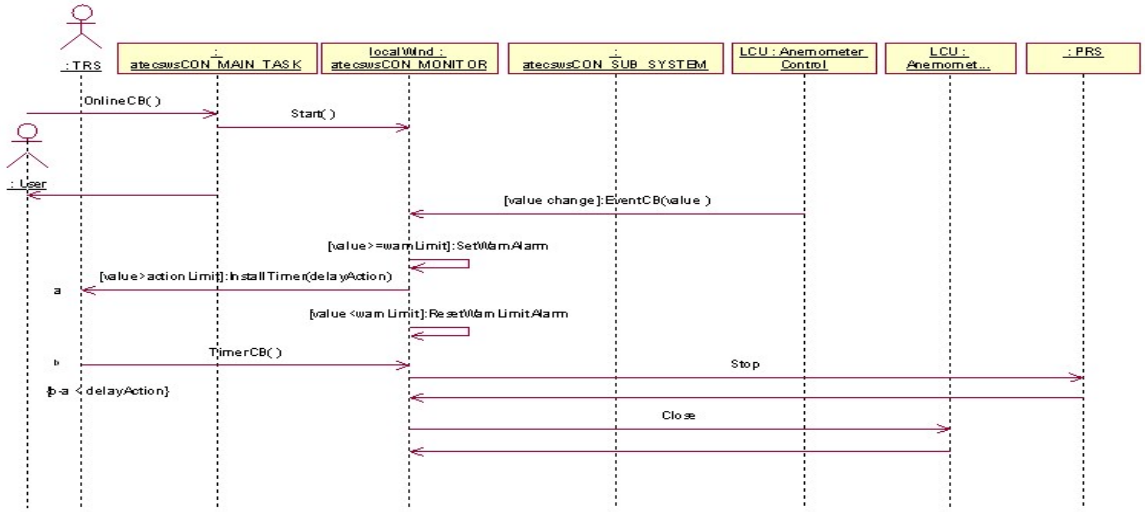
4.1. Entity Relationship Diagram with data dictionary



4.2. Class Diagram



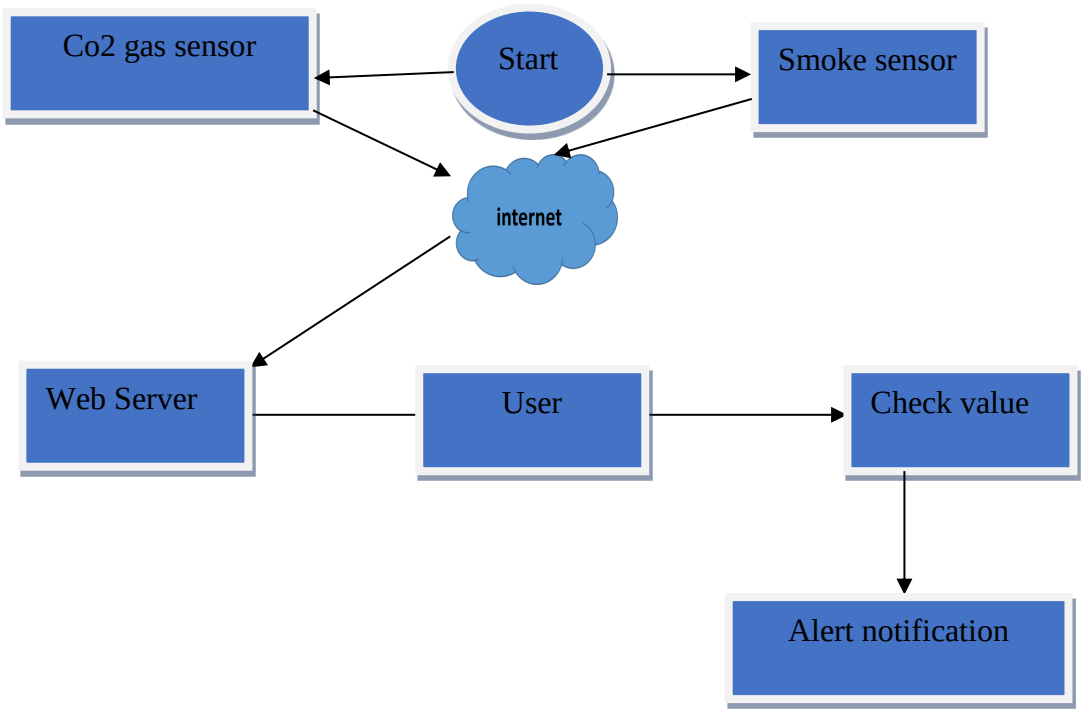
4.3. Sequence / Collaboration Diagram



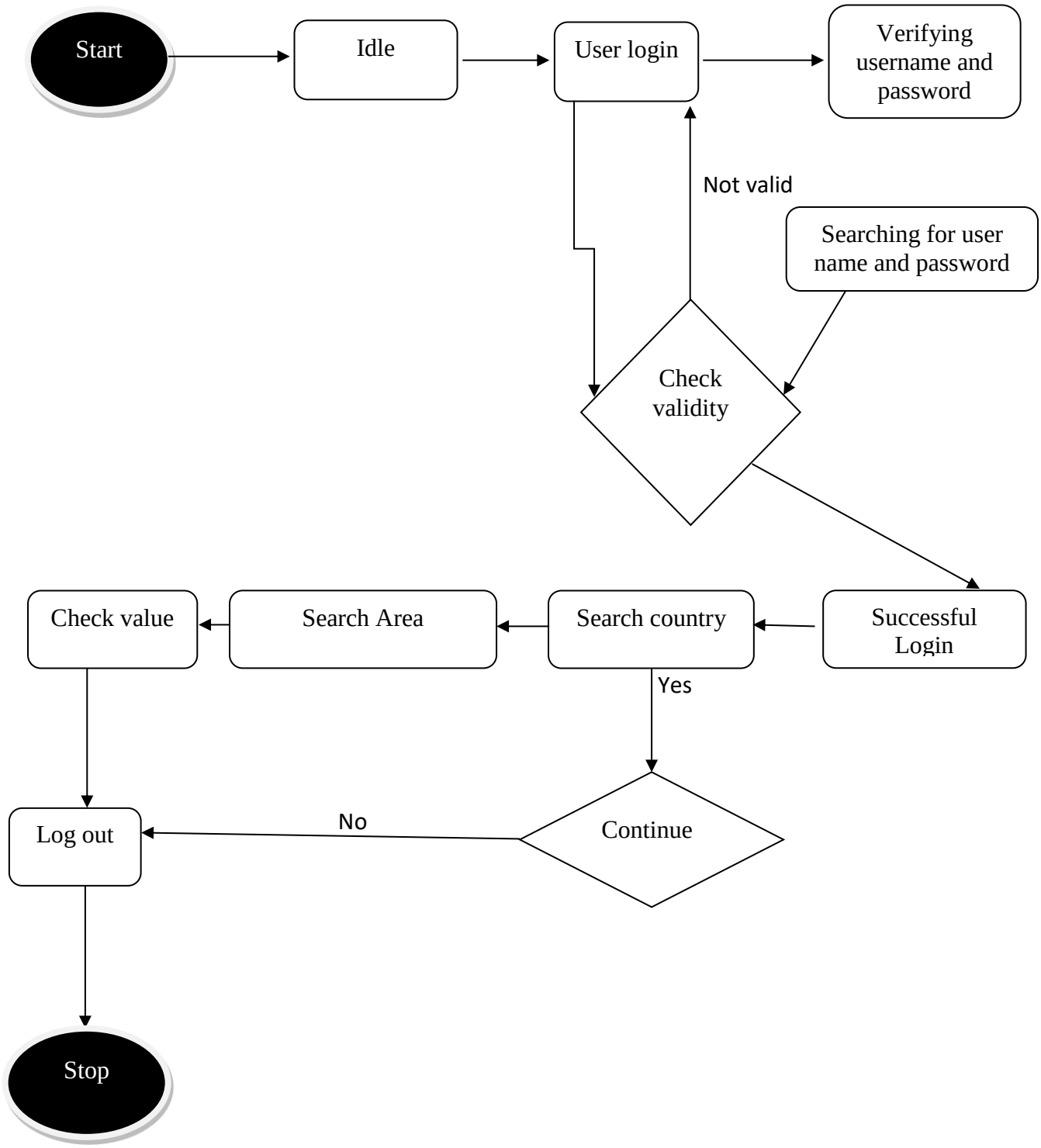
4.4. Operation contracts

Use Cases often fully describe the behavior of a system l But they may not be enough l Operation Contracts describe how the internal state of the concepts in the Domain Model may change l Operation Contracts are described in terms of preconditions and post conditions.

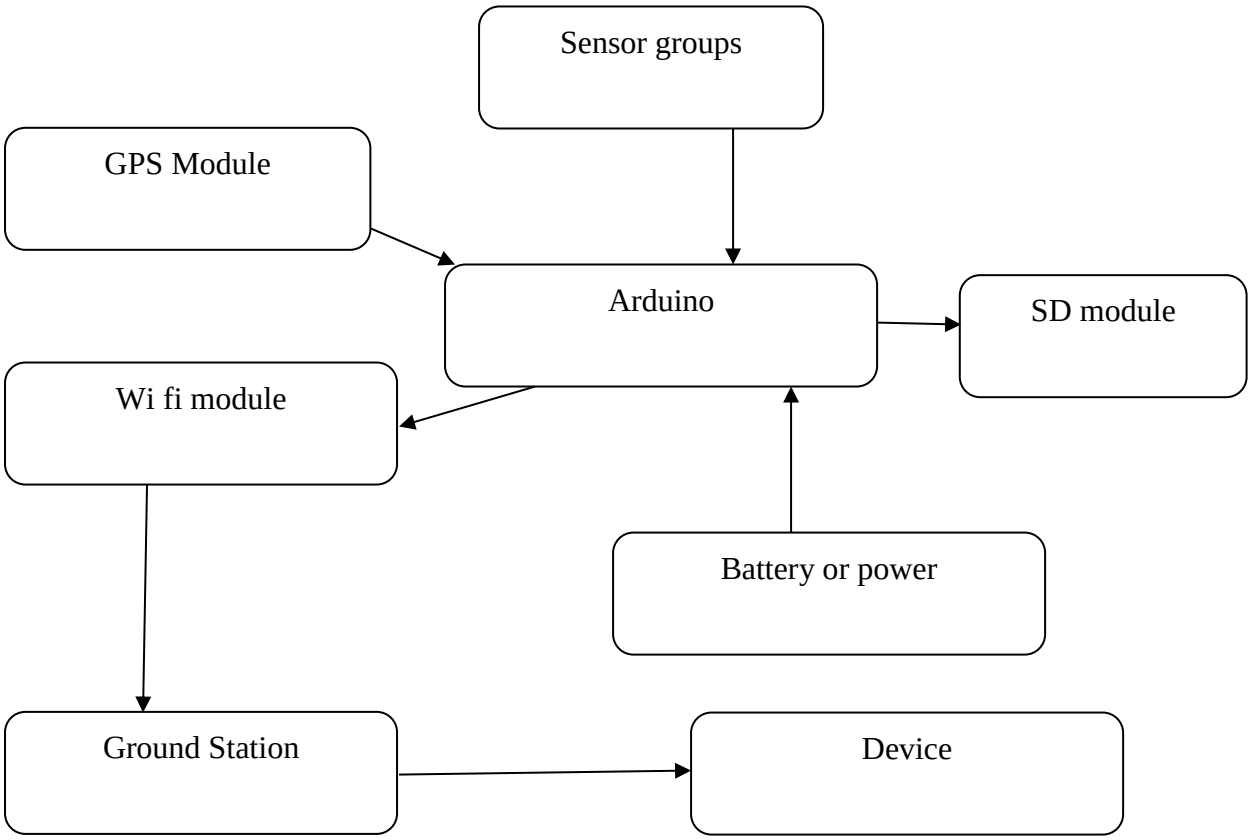
4.5. Activity Diagram



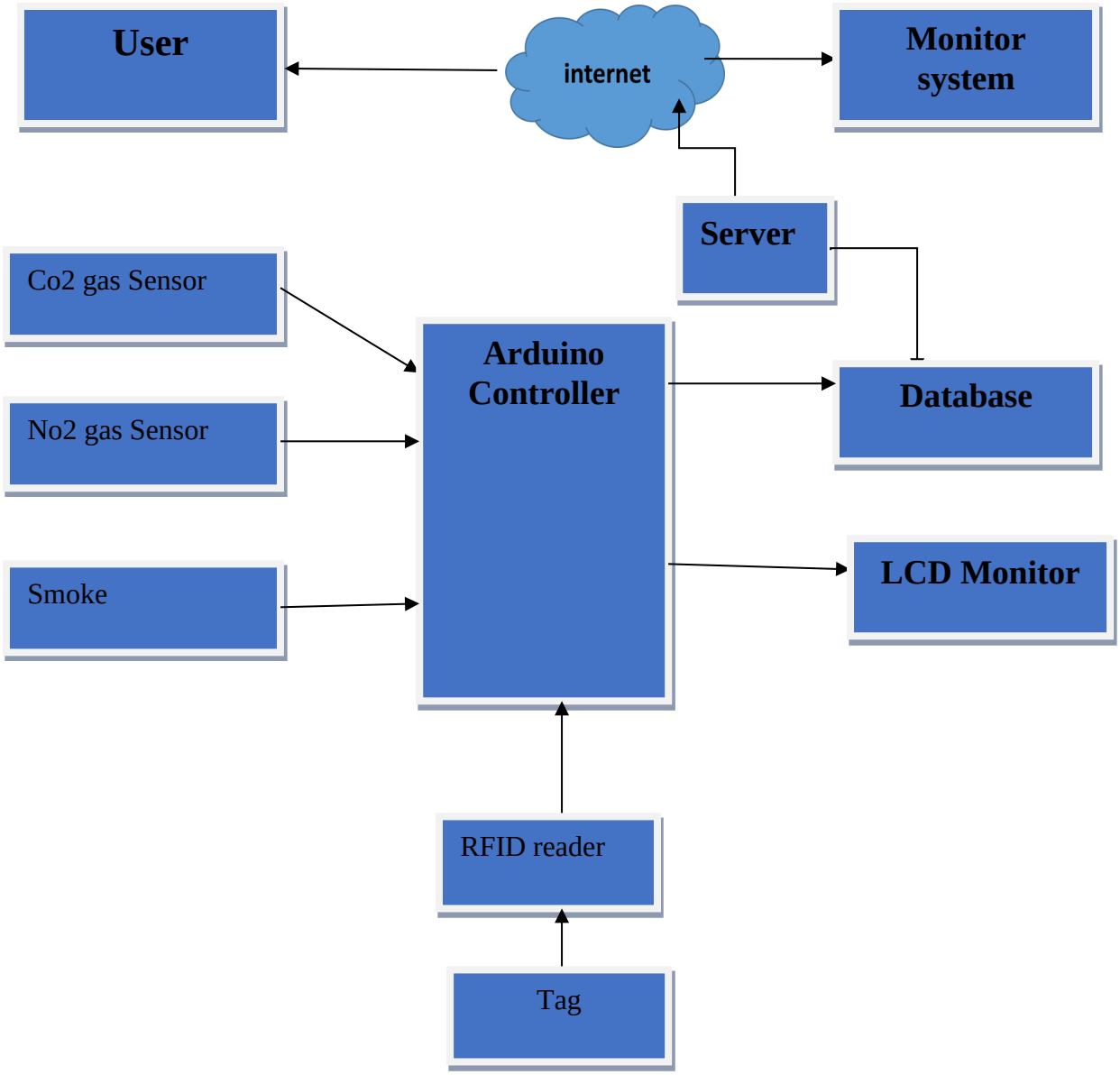
4.6. State Transition Diagram



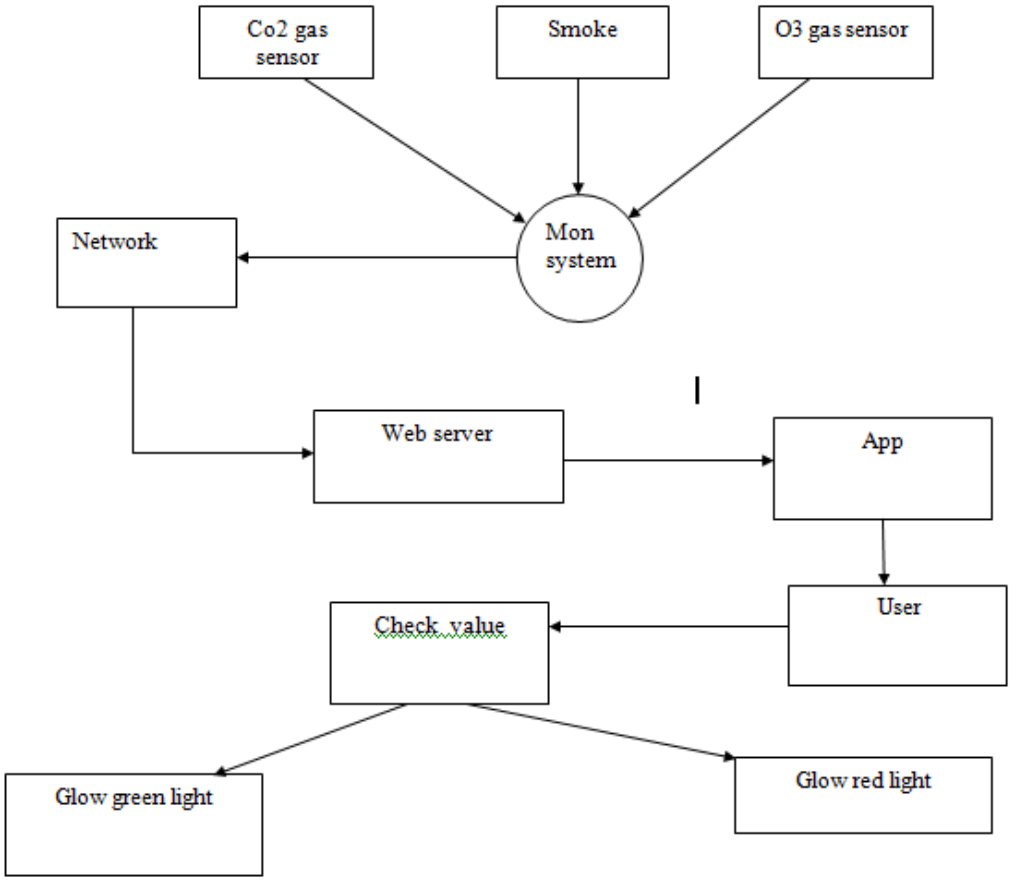
4.7. Component Diagram



4.1. Deployment Diagram



4.2. Data Flow diagram



Chapter 5
Implementation

Chapter 5: Implementation

We are going to make an Air quality sensor using Arduino in which we will monitor the Air Quality over a webserver using internet and will trigger an alarm when the air quality goes down beyond a certain level, means when there are sufficient amount of harmful gases are present in the air like CO2, smoke, alcohol, benzene and NH3. It will show the air quality in PPM on the LCD and as well as on webpage so that we can monitor it very easily. We will use MQ135 sensor which is the best choice for monitoring Air Quality as it can detects most harmful gases and can measure their amount accurately.

5.1. Important Flow Control/Pseudo codes

The air quality detector modules are installs with sensor modules that can detect a wide range of Volatile Organic Compounds (VOCs) and are intended for indoor air quality monitoring. They are able to respond to carbon monoxide, alcohol, acetone, thinner, formaldehyde, and other slightly toxic gases.These modules are often cheaper and smaller in size than PM2 5 sensors.

5.2. Components, Libraries, Web Services and stubs

- Gas sensors (MQ-2, 3, 7)
- Air quality sensor (MQ-135)
- Android device
- Mobile application development platform
- IoT-cloud
- Jumper wires
- Programming skills
- Arduino IDE/Arduino Web IDE
- Solar panel for green power source

5.3. Deployment Environment

In Deployment environment we work on android studio for build our android based application

5.4. Tools and Techniques

We have used Arduino UNO, MQ-135 air quality sensor, LCD display, breadboard, jumper wires, and potentiometer to develop an Arduino based air pollution detector which combined a small-sized, minimum-cost sensor to an Arduino microcontroller unit . The device is linked to a computer through a serial connection. From the sensor, the collected data through the Arduino microcontroller.

Tools:

Android Studio
HTML
Breadboard

5.5. Best Practices / Coding Standards

Arduino codding on Dev c++ or android studio

Use C++ language

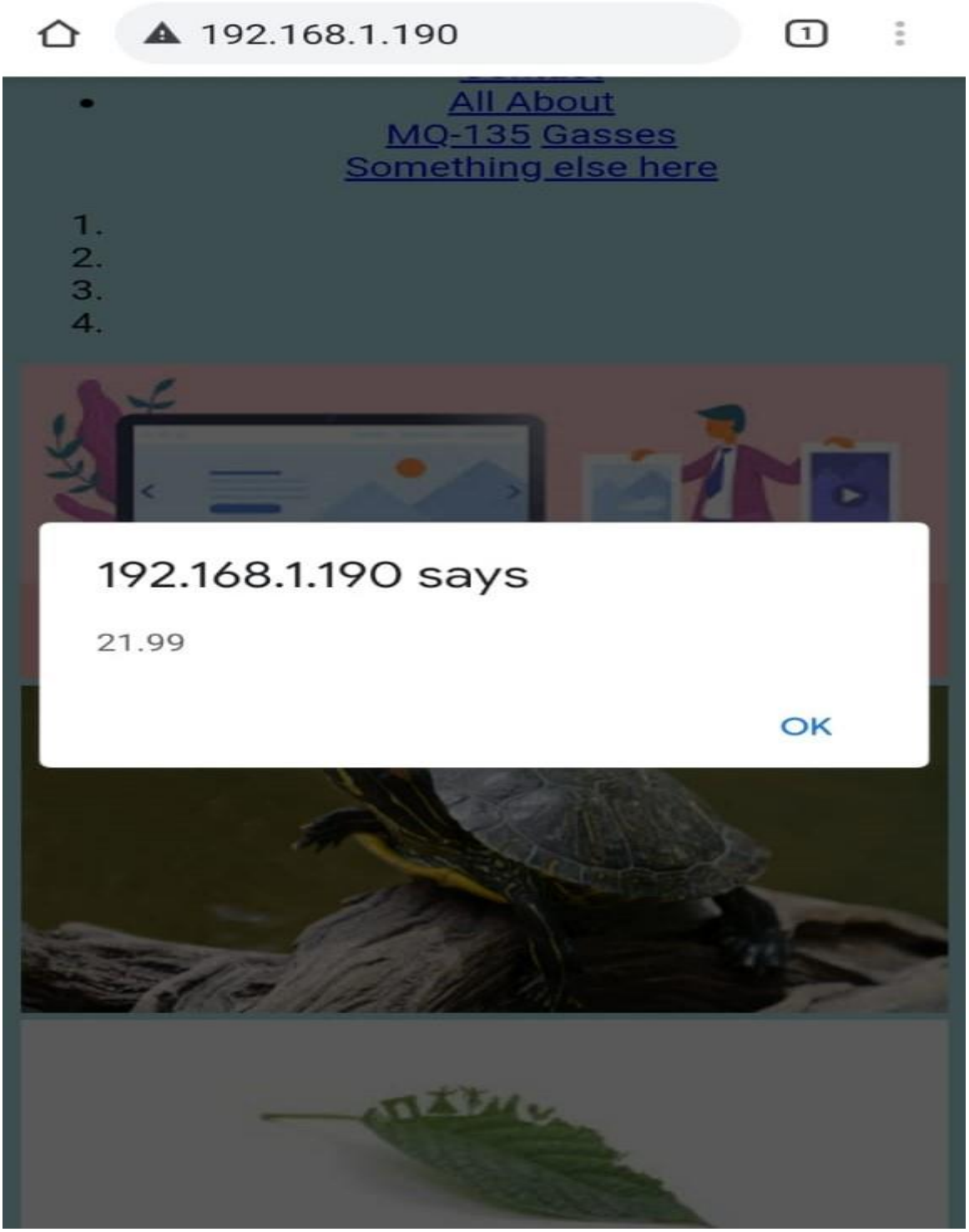
5.6. Version Control

we just working on version 1 after the passage of time we changed the version. From the start of the project number of versions updated and finalized now version 1.5 mentioned here

Chapter 6

Testing and Evaluation

Chapter 6: Testing and Evaluation



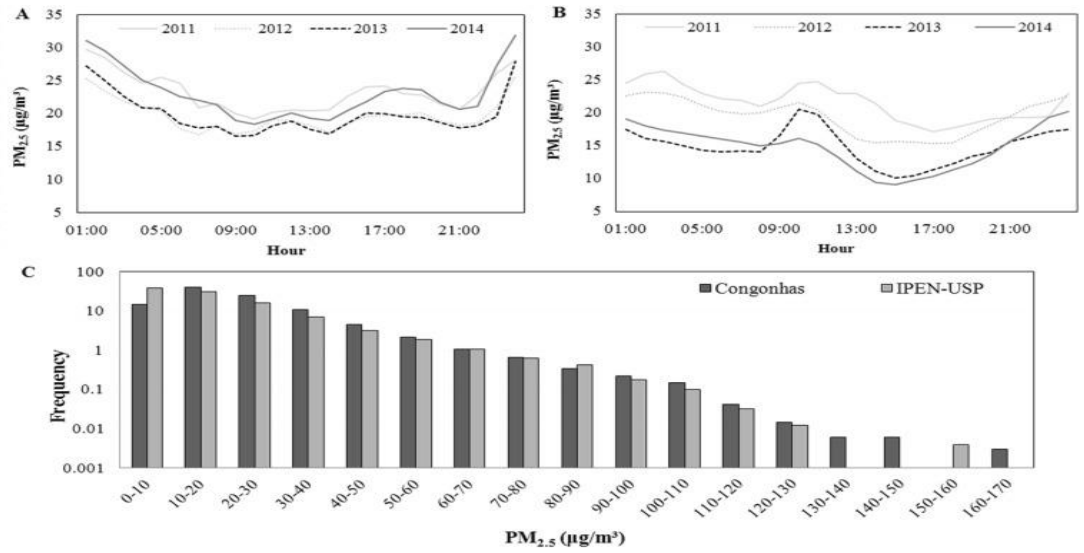
6.1. Use Case Testing

A Use Case in Testing is a brief description of a particular use of the software application by an actor or user. Use cases are made on the basis of user actions and the response of the software application to those user actions. It is widely used in developing test cases at system or acceptance level.

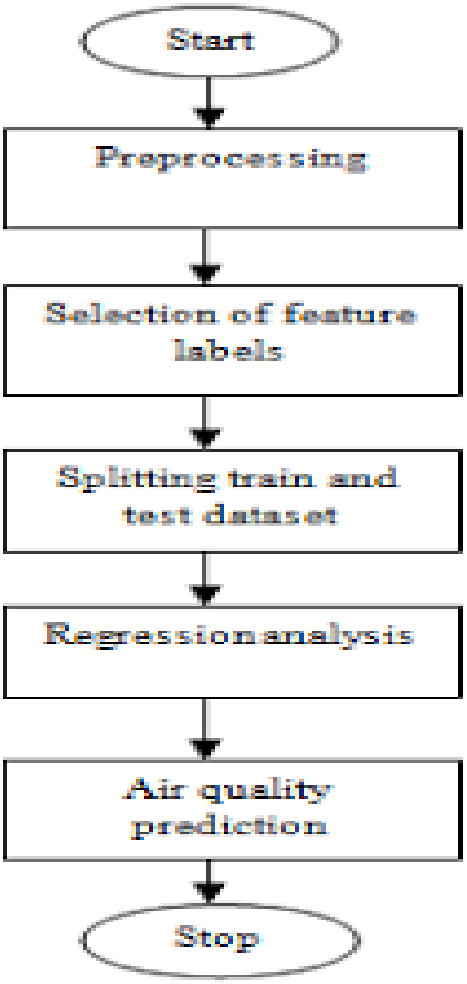
6.2. Equivalence partitioning

Equivalence partitioning or equivalence class partitioning (ECP) is a software testing technique that divides the input data of a software unit into partitions of equivalent data from which test cases can be derived. In principle, test cases are designed to cover each partition at least once.

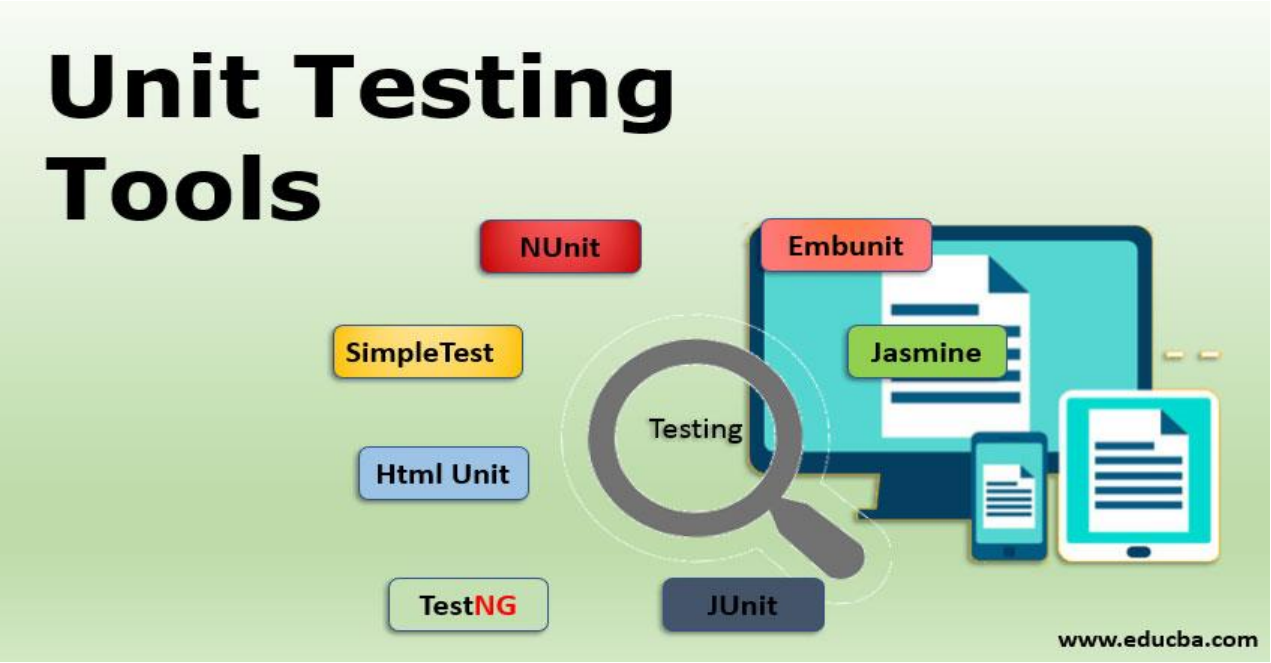
6.3. Boundary value analysis



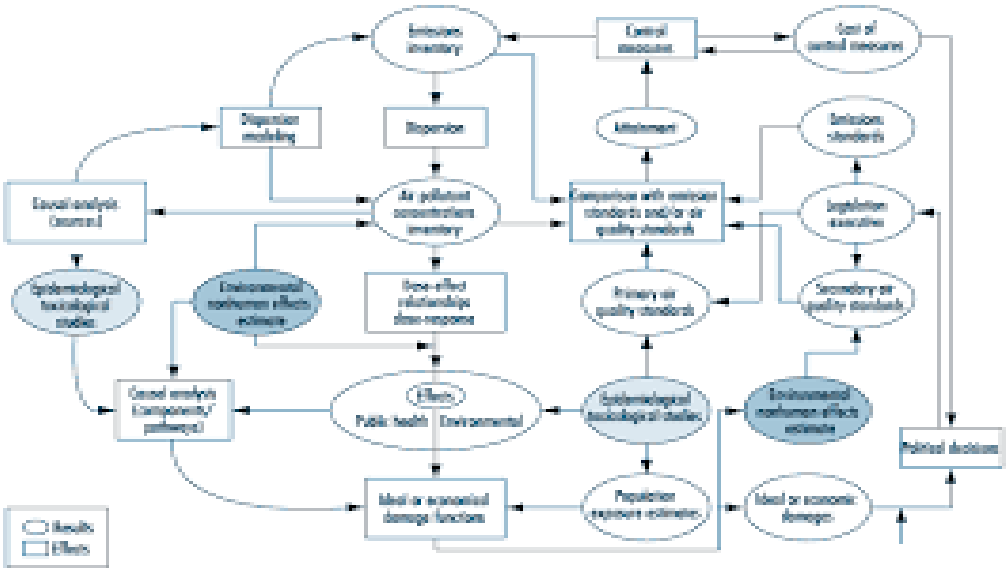
6.4. Data flow testing



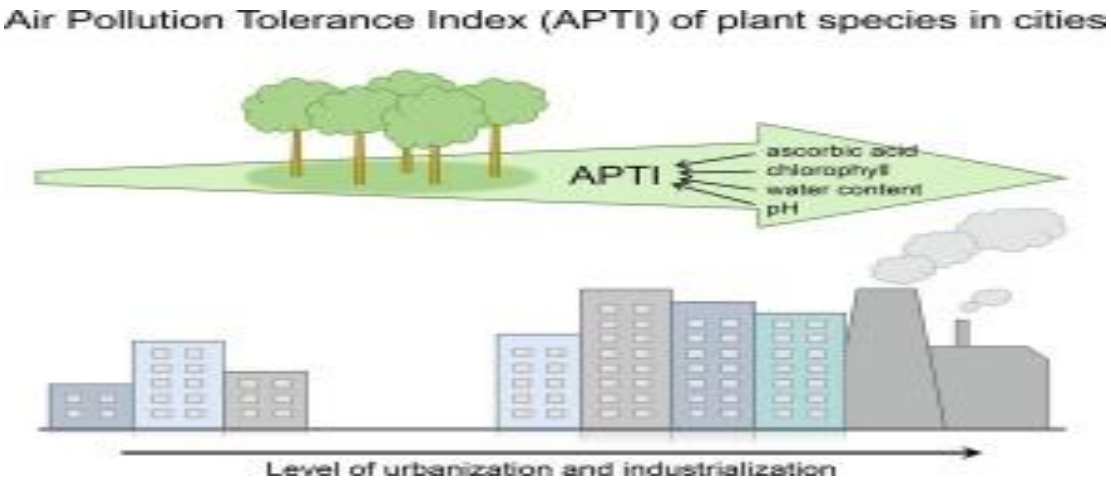
6.5. Unit testing



6.6. Integration testing



6.7. Stress Testing



6.8. Performance testing

Chapter 7

Summary, Conclusion and Future Enhancements

Chapter 7: Summary, Conclusion & Future Enhancements

7.1. Project Summary

This project is about to monitor the quality of our environment. When we keep some fuels which have harmful gases like CO2, NO2 etc. (when we keep an incense stick close to this sensor), the RED LED will glow and the buzzer starts ringing else if it is good quality air around, then green LED will glow. Sensor was giving us value of 90 when there was no gas near it and the safe level of air quality is 350 PPM and it should not exceed 1000 PPM. When it exceeds the limit of 1000 PPM, then it starts cause Headaches, sleepiness and stagnant, stale, stuffy air and if exceeds beyond 2000 PPM then it can cause increased heart rate and many other diseases. When the value will be less than 1000 PPM, then the LCD and webpage will display “Fresh Air”. Whenever the value will increase 1000 PPM, then the buzzer will start beeping and the LCD and webpage will display “Poor Air, Open Windows”. If it will increase 2000 then the buzzer will keep beeping and the LCD and webpage will display “Danger! Move to fresh Air”.

7.2. Achievements and Improvements

This device can informed about the harmful gases . An increasing body of research has shown that air pollution—even in relatively low doses—also affects educational outcomes across several distinct age groups and varying lengths of exposure. This implies that a narrow focus on traditional health outcomes, such as morbidity and mortality, may understate the true benefit of reducing pollution, as air pollution also affects scholastic achievement and human capital formation. In future we will update it and try to give best output more effective.

7.3. Critical Review

The aim of this project is that we will make a device that detect the gases from environment and show its values and quality on a local website using 3g mobile network. If the quality of gases is good no message will be display on web page. If the quality is poor or moderate a message will be shown in message box that is “moderate or pure”. We can see the values of gases on web page through wifi module and ip address.

7.4. Lessons Learnt

We learnt from this project how MQ 135 works , learnt about those gases which is sense by this sensor.

7.5. Future Enhancements/Recommendations

In future we will try to control his environment by remote i.e (Exhausting fan or Heater etc). We will set alarm if the values of gases cross a specific limit.

Reference and Bibliography

Reference and Bibliography

References

[1]Lanjewar U M and Shah J J 2012 Int. J. Adv. Comput. Res. 2 p 19

[2] Kaur N, Mahajan R, Bagai D and Student P G 2016 Int. j. innov. res. sci. eng. technol. 5 p 9635-9646

[3] Hanafi N H, Hassim M H, Noor Z Z, Ten J Y, Aris N M and Jalil A A 2019 E3S Web. Conf. 90 p 01009

[4] Krishna, V. Siva, and S. Arun. "Embedded System Based Air Pollution Detection in Vehicles." (2015).

[5] Piedrahita, R., et al. "The next generation of low-cost personal air quality sensors for quantitative exposure monitoring." Atmospheric Measurement Techniques 7.10 (2014): 3325.

[6] Kim, Sunyoung, and Eric Paulos. "inAir: measuring and visualizing indoor air quality." Proceedings of the 11th international conference on Ubiquitous computing. ACM, 2009.

[7] Al-Dahoud, A., Fezari, M., Jannoud, I., & AL-Rawashdeh, T. (2015, March). Monitoring Metropolitan City Air-quality Using Wireless Sensor Nodes based on ARDUINO and XBEE. In *Proceedings of the International Conference on Circuits, Systems, Signal Processing, Communications and Computers (CSSCC), Vienna, Austria, March* (Vol. 1517).

[8] Rajkumar, M. Newlin, M. S. Sruthi, and V. Venkatesa Kumar. "IOT Based Smart System for Controlling Co2 Emission." (2017).

[9] Razak, R., Iktibar musibah jerebu 1997. Sinar Harian Online, 2013.

[10] ASMC, Annual Hotspot Count. ASEAN Specialised Meteorological Centre. Available at: http://asmc.asean.org/asmc-haze-hotspot-annual#Hotspot [Accessed December 1, 2015].

[11] France-presse, A., Worst haze in a decade blankets southern Thailand, 2015.

Index

Index

[A]

[B]

[C]