AERO-CHECK - AN APPLICATION FOR AIR POLLUTION MONITORING

This project report is submitted to

Rashtrasant Tukadoji Maharaj Nagpur University

in the partial fulfilment of the requirement for the award of the degree

of

Bachelor of Engineering in Computer Technology

by

Mr. Akash S. Natkar (CT14057)

Mr. Yugant P. Kadu (CT14060)

Ms. Nilu Kant (CT14061)

Mr. Deepak Kumar Gupta (CT14068)

Mr. Satish W. Wanjari (CT14078)

Under the guidance of

Shri. Bhushan Deshpande

Assistant Professor

2016-17

DEPARTMENT OF COMPUTER TECHNOLOGY
KAVIKULGURU INSTITUTE OF TECHNOLOGY AND SCIENCE
RAMTEK – 441106

DEPARTMENT OF COMPUTER TECHNOLOGY
KAVIKULGURU INSTITUTE OF TECHNOLOGY AND SCIENCE
RAMTEK-441106

This is to certify that the project report entitled '**Aero-Check - An Application for Air Pollution Monitoring**' carried out by Mr. Akash S. Natkar (CT14057), Mr. Yugant P. Kadu (CT14060), Ms. Nilu Kant (CT14061), Mr. Deepak Kumar Gupta(CT14068) and Mr. Satish W. Wanjari (CT14078) of the B.E. third year of computer technology, during the academic year 2016-2017, in the partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering (Computer Technology)** offered by the **Rashtrasant Tukadoji Maharaj Nagpur University**, Nagpur**.**

Shri. Bhushan Deshpande
Guide

Shri. V. P. Mahatme
Head of the Department

Dr. B. Ram Rathan Lal
Principal

Date:
Place: Ramtek

DECLARATION

We, the projectees, of third year computer technology department for the session hereby declare that:

1. The work contain in this project has been done by us under the supervision of our guide.
2. The work has not been submitted to any other institute for any degree or diploma.
3. We have followed the guideline provided by institute in preparing the project report.
4. We have conformed to the norms and guidelines given in the Ethical Code of Conduct of the institute.
5. Whenever we have used material (data, theoretical analysis, figures and text) from other sources, we have given due credit to them by citing giving their details in the references.

Project-mates

ACKNOWLEDGEMENT

We are grateful to our respected guide **Shri. Bhushan Deshpande** for his kind, disciplined and invaluable guidance which inspired us to solve all the difficulties that came across during completion of project.

We express our special thanks to **Shri. V. P. Mahatme**, Head of the Department, for his kind support, valuable suggestion and allowing us to use all facilities that are available in the Department during this project.

Our sincere thanks are due to **Dr. B. Ram Rathan Lal**, Principal, for extending all the possible help and allowing us to use all resources that are available in the Institute.

We are also thankful to our **Parents** and **Friends** for their valuable cooperation and standing with us in all difficult conditions.

Project-mates

ABSTRACT

Air Pollution is becoming a major concern in India. Due to increase in traffic, factories, burning of fossil fuels, emission of hazardous gases like CO_2, O_3 and NH3 have been increased. One of factor to turn Taj Mahal's colour brownish-yellow is burning of fossil fuel. Smog, a mixture of smoke and fog, is reducing visibility and has increased rate of increased accident in North region. Last year, Delhi was ranked as most polluted city in the world followed by Beijing. Therefore, Aero-Check helps government authorities and other peoples to keep track on the quality of air.

The main objective of Aero-Check application is to view air quality information of different region of India on a Google map. User can switch between different cities to check level of pollution. A quality of air is calculated using Air Quality Index (AQI). An Air Quality Index is a number used by government agencies to communicate to the people for current status of the pollution. The National Air Quality Index (NAQI) has outlined 'One Number – One Description' for the common man to judge the air quality within his vicinity.

Aero-Check application will be developed using Android technology. Android is a mobile operating system developed by Google Inc., based on the Linux Kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. Android's user interface is mainly based on direct manipulation using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching to manipulate on-screen objects, along with a virtual keyboard for text input.

KEYWORDS: AQI, Linux, Android, Kernel, Google Inc., NAQI

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

Air pollution is a most serious problem of the current time all over the world especially in the large cities because of the huge level of industrialization. The release of such air pollutants in heavy concentrations such as smog, particulates, solid materials, etc. are getting settled over the city, causing air pollution and health hazards to the people. Lots of dirty wastes produced by people on daily basis especially in the big cities polluting the whole atmospheric air to a great extent.

Day by day the fresh air of the environment is getting polluted because of the mixing of particulates, biological molecules, and other harmful materials. Such polluted air is causing health problems, diseases and death. Air pollution is one of the most important environmental issues which requires to be noticed and solved by the efforts of all of us.

Exposure to nitrogen oxides (NO_2) and sulphur oxides (SO_2) can irritate the lungs, reduce lung function, and increase susceptibility to allergens in people with asthma. Both NO_X and SO_X are also precursors of fine particulate matter ($PM_{2.5}$).

Fine particulate matter and ground-level ozone (O_3) have been associated with eye, nose and throat irritations, shortness of breath, exacerbation of respiratory conditions, chronic obstructive pulmonary disease and asthma, exacerbation of allergies, increased risk of cardiovascular disease and premature death. The young, the elderly and those with acute illnesses are at greater risk.

Innovations made in information technology and the role of technology in various fields have wide ranging effects on every person living in this world and has its effects on numerous policies like economic productivity, lifestyle and has a great impact on the global market and society.

1.1 Purpose

The main objective of Aero-Check application is to view air quality information of different regions of India on a Google map. User can switch between different cities to check level of pollution. A quality of air is calculated using Air Quality Index (AQI). An Air Quality Index is a number used by government agencies

to communicate to the people for current status of the pollution. The National Air Quality Index (AQI) has outlined 'One Number – One Description' for the common man to judge the air quality within his vicinity. Different regions have different air quality.

1.2 Problem Statement

The problem statement is to provide the detailed information of air quality of different regions of India. In which the User can switch between different cities to check level of pollution. A quality of air is calculated using Air Quality Index (AQI). An Air Quality Index is a number used by government agencies to communicate to the people for current status of the pollution.

1.3 Aero-Check

Aero-Check is an android application for Air Pollution Monitoring**.** This application allows us to view air quality information of different region of India on a Google map. Here we will get the recent status about the air quality. This generic application designed for assisting the people of an initiate regarding information of different particulates present in air. It also provides support that people can get prior information about pollution in different regions of India. The data has been collected from CPCB (Central Pollution Control Board) of India.

Aero-Check shows air quality based on the following parameters

- Particulate matter and particulate microns (PM2.5 and PM10).
- The hazardous and polluting gases like CO, CO2, SO2, NO2, O3.

Based on the above parameters the air quality of each city is calculated. Aero-Check tells the pollution level of city which is represented on map with different colours. It has all information and statistics about Air Pollution and Air Quality of India.

1.4 Motivation

Air pollution may cause long term health issue as well as environmental damage. It may also trigger further symptoms in those with chronic condition such as Asthma. New Delhi has an air pollution problem. It is the 2nd most worst polluted city in the world, with an annual average PM2.5 measurement of 122. Average air-quality index reading for Delhi in October 2015 was considered "poor" according to the Central Pollution Control Board's air quality index bulletin. New Delhi's air pollution

hit dangerous levels as India celebrated Diwali, the Hindu festival of lights, it affected people badly. Air pollution in Delhi is killing around 80 people every day, according to an international study, said the Centre. Not only in Delhi but also in various regions air pollution is harming people. So these motivated us to make such an application which will prior inform people about the pollution level so that people may take effective steps to avoid such conditions.

1.5 Organization of the Report

Chapter 1: Include Introduction of the project including the main objective.

Chapter 2: Explains Literature survey which includes the literature study and reference for our project.

Chapter 3: Tells about Proposed Approach and System Architecture which includes brief implementation of the project.

Chapter 4: Describes Implementation which contains the implementation of different functions for the system.

Chapter 5: Gives the Result and Discussion.

Chapter 6: Contains Conclusion and Future Scope.

CHAPTER 2
LITERATURE REVIEW

Various parallel applications were studied that were similar to the project and came across many technologies that should be used to develop the project. This technologies were found helpful to make maps, graphs and database for Aero-Check project. In this chapter, parallel applications along with technologies which have been used are discussed like android, its features, development kit etc. While making Aero-Check, some applications were found similar. The detailed study of those applications were done and their details are listed below:

2.1 Air Quality: Real time AQI

Details of Air Quality application are as follows:

- **Application developer:** World Air Quality Index Project team.
- **Year of development:** 2008
- **Application size:** 9.96MB
- **Application website:** www.aqicn.org and www.waqi.com
- **Email address:** support@aqicn.org

The Air Quality application shows the real-Time Air Quality Index (AQI) for more than 60 countries in the World (Mainland China, Hong Kong, Taiwan, Singapore, Vietnam, India, Malaysia, Thailand, South Korea and Japan).

The Air Quality data sources varies depending on the cities:

- The US Embassy PM2.5 data is used for Shanghai, Beijing, Chengdu and Guangzhou.
- The China Ministry Environment Pollution Department PM10 data is used for all other cities in China.
- For Hong Kong and Taiwan, the respective Environment agencies PM2.5 data is used.
- For Singapore, the Singapore National Environment Agency PM2.5 data is used.
- The Central Pollution Control Board of India PM2.5 is used for all other cities in India.

- The Delhi Pollution Control Committee (Government of NCT of Delhi) data is used for Delhi region.

2.1.1 Home Page

The Air Quality application shows the real-Time Air Quality Index (AQI) for more than 60 countries in the World (Mainland China, Hong Kong, Taiwan, Singapore, Vietnam, India, Malaysia, Thailand, South Korea and Japan).

Fig. 2.1 Snapshot of air quality

2.1.2 Map

Air Quality shows the region in following form in the form of map. The Air Quality data sources varies depending on the cities:

- The US Embassy PM2.5 data is used for Shanghai, Beijing, Chengdu and Guangzhou.
- The China Ministry Environment Pollution Department PM10 data is used for all other cities in China.
- For Hong Kong and Taiwan, the respective Environment agencies PM2.5 data is used.
- For Singapore, the Singapore National Environment Agency PM2.5 data is used.

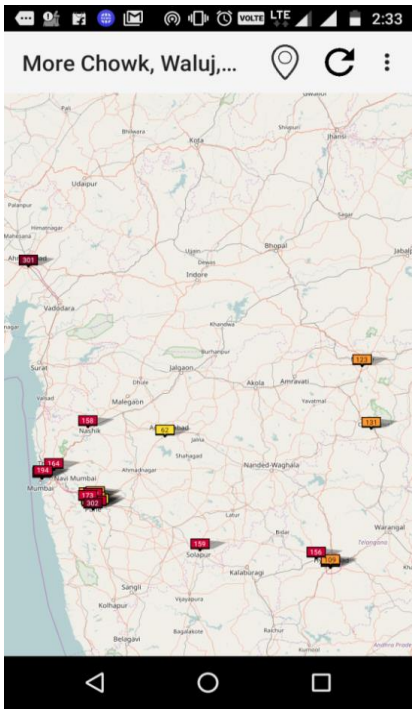

Fig. 2.2 Map of air quality

2.2 Air Quality Index Breezometer

Details of Air Quality Index Breezometer application are as follows:

- **Application developer:** BreezoMeter team.
- **Application size:** 14.97MB
- **Application website:** www.breezometer.com
- **Email address:**support@breezometer.com

BreezoMeter is an award winning, top rated Air Quality app. For the first time, user can see what user breathe (Service is available in Australia, China, India, United States, United Kingdom, Argentina, Brazil, Israel, Spain, Italy, Hong Kong, Austria, Belgium, Luxembourg, Switzerland, Netherlands, Turkey, New Zealand, Japan, Singapore, Germany, South Korea, Denmark, Ireland, Norway, Sweden, Poland, Taiwan, Finland, Gibraltar, France, Macedonia, Croatia, Mexico, Canada). But that's only the start, BreezoMeter Real Time Air Quality Analysis offers:

- Real time air quality maps - see in user's own eyes how outdoor air quality looks like at user street, block and even the all country.
- Actionable personalized health recommendations - know what user should do, whether user is at home, looking for an exercise route, planning a trip or watching the kids.

- Notifications on outdoor air quality changes - save user favorite locations and track their air quality 24/7.

2.2.1 Home Page

Real time air quality maps-see in user own eyes how outdoor air quality looks like at user own eyes how outdoor air quality looks like at user street, block and even the all country.

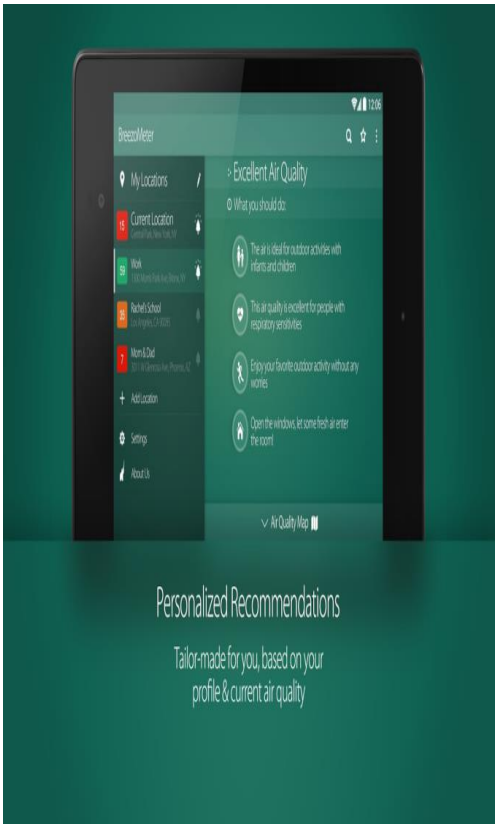

Fig. 2.3 Snapshot of application

2.2.2 Map Representation

BreezoMeter's secret sauce lies in technology that utilizes big data infrastructure to continuously gather air quality & weather measurements from literally thousands of sources among them more than 7000 official air quality monitoring stations worldwide.

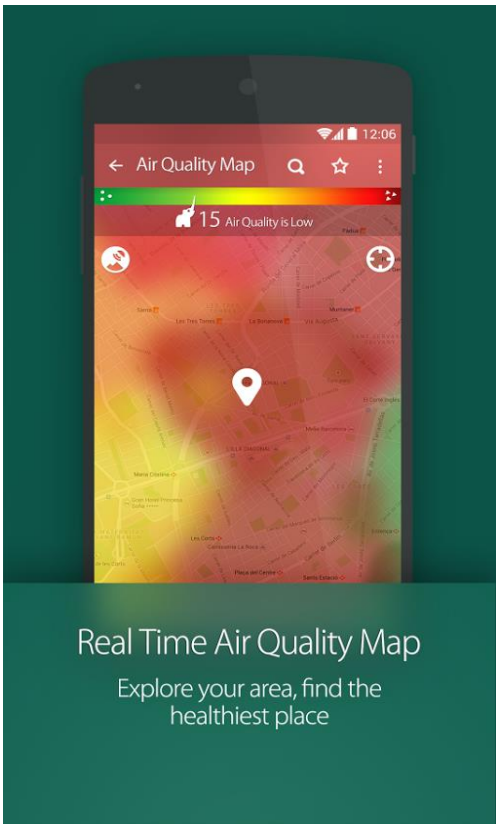

Fig. 2.4 Map representation

2.3 Air Quality India

Details of Air Quality India application are as follows:

- **Application developer:** Sohil and Oziom team.
- **Application size:** 42.52MB
- **Email address:**sohil@oziom.com
- Air Quality India application provides user with Real-time Air quality data from all across India. The data has been measured as per the standard norms are given by Central Pollution Control Board of India. Air Quality India App shows air quality based on the following parameters: Particulate matter and particulate microns (PM 2.5 and PM10).The hazardous and polluting gases in the atmosphere. Gases like CO, CO2, SOx, NOx, SO2, NO2.Humidity and temperature of the atmosphere.

Activity suggestion by Air Quality India app gives user suggestions based on pollution level in user's area. Is it a good time to go for morning exercise? Should user take pet for a walk? The activity suggestion by Air Quality India App gives user all the suggestions.

Air is a dynamic medium there is so much about it that one might be unaware about. Increase one's pool of knowledge with the Knowledge centre provided by Air Quality India App. It has all the facts, information, and statistics about Air Pollution and Air Quality across the globe. The knowledge centre acts as an encyclopaedia about air.

Air Quality App allows user to compare the pollution levels of various cities. With city comparison, user can make wiser choices when planning next vacation or shift to a new area.

Own device is added into the Air quality India app. If user have purchased DIY Air Quality Monitor, AirOwl, user can add it and configure with the Air Quality India App. Air Quality India App gives the chance to access the data and share the data of user's device to the public and contributes to fighting against pollution.

2.3.1 Home Page

Air Quality India application provides user with Real-time Air quality data from all across India. The data has been measured as per the standard norms are given by Central Pollution Control Board of India. Air Quality India App shows air quality based on the following parameters: Particulate matter and particulate microns (PM 2.5 and PM10).

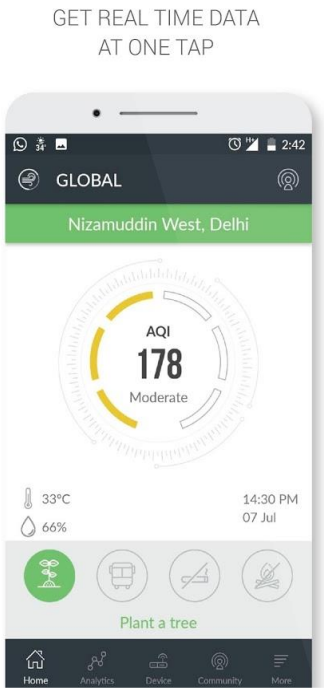

Fig. 2.5 AQI of the region

2.3.2 Graph Representation

This application gives AQI data as shown in figure and also activity suggestion which gives suggestions based on pollution level in the area.

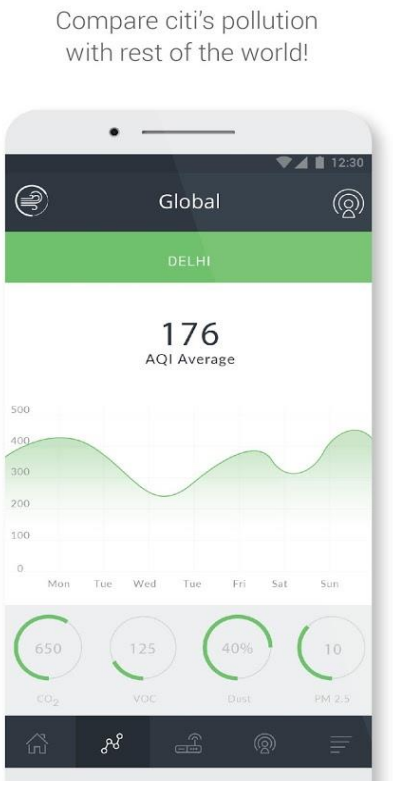

Fig. 2.6 Graph representation

2.4 Android

Android is a mobile operating system developed by Google, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. Android's user interface is mainly based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input. In addition to touchscreen devices, Google has further developed Android TV for televisions, Android Auto for cars, and Android Wear for wrist watches, each with a specialized user interface. Variants of Android are also used on notebooks, game consoles, digital cameras, and other electronics.

Android has the largest installed base of all operating systems (OS) of any kind. Android has been the bestselling OS on tablets since 2013, and on smartphones it is dominant by any metric.

Initially developed by Android, Inc., which Google bought in 2005, Android was unveiled in 2007 along with the founding of the Open Handset Alliance – a consortium of hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices. As of July 2013, the Google Play store has had over one million Android applications ("apps") published – including many "business-class apps" that rival competing mobile platforms – and as of May 2016 over 65 billion applications downloaded. An April–May 2013 survey of mobile application developers found that 71% of developers create applications for Android, and a 2015 survey found that 40% of full-time professional developers see Android as their priority target platform, which is comparable to Apple's iOS on 37% with both platforms far above others. In September 2015, Android had 1.4 billion monthly active devices.

Android's source code is released by Google under an open source license, although most Android devices ultimately ship with a combination of free and open source and proprietary software, including proprietary software required for accessing Google services. Android is popular with technology companies that require a ready-made, low-cost and customizable operating system for high-tech devices Its open nature has encouraged a large community of developers and enthusiasts to use the open-source code as a foundation for community-driven projects, which deliver updates to older devices, add new features for advanced users or bring Android to devices originally shipped with other operating systems. The success of Android has made it a target for patent (and copyright) litigation as part of the so-called "smartphone wars" between technology companies.

2.4.1 History

Android, Inc. was founded in Palo Alto, California in October 2003 by Andy Rubin (co-founder of Danger),[j]Rich Miner (co-founder of Wildfire Communications, Inc.), Nick Sears (once VP at T-Mobile), and Chris White (headed design and interface development at WebTV) to develop "smarter mobile devices that are more aware of its owner's [sic] location and preferences". The early intentions of the company were to develop an advanced operating system for digital cameras. Though, when it was realized that the market for the devices was not large enough, the company diverted its efforts toward producing a smartphone operating system that would rival Symbian and Microsoft Windows Mobile. Despite the past accomplishments of the founders

and early employees, Android Inc. operated secretly, revealing only that it was working on software for mobile phones. That same year, Rubin ran out of money. Steve Perlman, a close friend of Rubin, brought him $10,000 in cash in an envelope and refused a stake in the company.

In July 2005, Google acquired Android Inc. for at least $50 million. Its key employees, including Rubin, Miner and White, stayed at the company after the acquisition. Not much was known about Android Inc. at the time, but many assumed that Google was planning to enter the mobile phone market with this move. At Google, the team led by Rubin developed a mobile device platform powered by the Linux kernel. Google marketed the platform to handset makers and carriers on the promise of providing a flexible, upgradeable system. Google had lined up a series of hardware component and software partners and signaled to carriers that it was open to various degrees of cooperation on their part.

2.4.2 Features

Android has the largest installed base of all operating systems (OS) of any kind. Android has been the bestselling OS on tablets since 2013, and on smartphones it is dominant by any metric. The features of Android are listed below:-

- **Interface**

Android's default user interface is mainly based on manipulation. Using touch inputs that loosely correspond to real-world actions, like swiping, tapping, pinching, and reverse pinching to manipulate on-screen objects, along with a virtual keyboard. Game controllers and full-size physical keyboards are supported via Bluetooth or USB. The response to user input is designed to be immediate and provides a fluid touch interface, often using the vibration capabilities of the device to provide haptic feedback to the user. Internal hardware, such as accelerometers, gyroscopes and proximity sensorsare used by some applications to respond to additional user actions, for example adjusting the screen from portrait to landscape depending on how the device is oriented, or allowing the user to steer a vehicle in a racing game by rotating the device, simulating control of a steering wheel.

- **Applications**

Applications ("apps"), which extend the functionality of devices, are written using the Android software development kit (SDK) and, often, the Java programming language, which has complete access to the Android APIs. Java may be combined with C/C++,

together with a choice of non-default runtimes that allow better C++ support; the Go programming language is also supported since its version 1.4, which can also be used exclusively although with a restricted set of Android APIs. The SDK includes a comprehensive set of development tools, including a debugger, software libraries, a handset emulator based on QEMU, documentation, sample code, and tutorials. Initially, Google's supported integrated development environment (IDE) was Eclipse using the Android Development Tools (ADT) plugin; in December 2014, Google released Android Studio, based on IntelliJ IDEA, as its primary IDE for Android application development. Other development tools are available, including a native development kit (NDK) for applications or extensions in C or C++, Google App Inventor, a visual environment for novice programmers, and various cross platform mobile web applications frameworks. In January 2014, Google unveiled an framework based on Apache Cordova for porting Chrome HTML 5web applications to Android, wrapped in a native application shell.

- **Memory Management**

Since Android devices are usually battery-powered, Android is designed to manage processes to keep power consumption at a minimum. When an application is not in use the system suspends its operation so that, while available for immediate use rather than closed, it does not use battery power or CPU resources.

Android manages the applications stored in memory automatically: when memory is low, the system will begin invisibly and automatically closing inactive processes, starting with those that have been inactive for longest. Lifehacker reported in 2011 that third-party task killers were doing more harm than good.

- **Virtual Reality**

At Google I/O on May 2016, Google announced Daydream, a virtual reality platform that relies on a smartphone and provides VR capabilities through a virtual reality headset and controller designed by Google itself.] The platform is built into Android starting with Android Nougat, differentiating from standalone support for VR capabilities. The software is available for developers, and was released in 2016.

2.5 Integrated Development Environment (IDE)

An integrated development environment (IDE) is a software application that provides comprehensive facilities to computer programmers for software development. An IDE

normally consists of a source code editor, build automation tools and a debugger. Most modern IDEs have intelligent code completion. Some IDEs, such as NetBeans and Eclipse, contain a compiler, interpreter, or both; others, such as SharpDevelop and Lazarus, do not. The boundary between an integrated development environment and other parts of the broader software development environment is not well-defined. Sometimes a version control system, or various tools to simplify the construction of a Graphical User Interface (GUI), are integrated. Many modern IDEs also have a class browser, an object browser, and a class hierarchy diagram, for use in object-oriented software development.

Integrated development environments are designed to maximize programmer productivity by providing tight-knit components with similar user interfaces. IDEs present a single program in which all development is done. This program typically provides many features for authoring, modifying, compiling, deploying and debugging software. This contrasts with software development using unrelated tools, such as vi, GCC or make.

One aim of the IDE is to reduce the configuration necessary to piece together multiple development utilities, instead providing the same set of capabilities as a cohesive unit. Reducing that setup time can increase developer productivity, in cases where learning to use the IDE is faster than manually integrating all of the individual tools. Tighter integration of all development tasks has the potential to improve overall productivity beyond just helping with setup tasks. For example, code can be continuously parsed while it is being edited, providing instant feedback when syntax errors are introduced. That can speed learning a new programming language and its associated libraries.

Some IDEs are dedicated to a specific programming language, allowing a feature set that most closely matches the programming paradigms of the language. However, there are many multiple-language IDEs.

While most modern IDEs are graphical, text-based IDEs such as Turbo Pascal were in popular use before the widespread availability of windowing systems like Microsoft Windows and the X Window System (X11). They commonly use function keys or hotkeys to execute frequently used commands or macros.

2.6 Eclipse

Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications, but it may also be used to develop applications in other programming languages via plug-ins, including: Ada, ABAP, C, C++, COBOL, D, Fortran, Haskell, JavaScript, Julia, Lasso, Lua, NATURAL, Perl, PHP, Prolog, Python, R, Ruby (including Ruby on Rails framework), Rust, Scala, Clojure, Groovy, Scheme, and Erlang. It can also be used to develop documents with LaTeX (via a TeXlipse plug-in) and packages for the software Mathematica. Development environments include the Eclipse Java development tools (JDT) for Java and Scala, Eclipse CDT for C/C++, and Eclipse PDT for PHP, among others.

The initial codebase originated from IBM Visual Age. The Eclipse software development kit (SDK), which includes the Java development tools, is meant for Java developers. Users can extend its abilities by installing plug-ins written for the Eclipse Platform, such as development toolkits for other programming languages, and can write and contribute their own plug-in modules. Since Equinox, plug-ins can be plugged-stopped dynamically and are termed (OSGI) bundles

Eclipse software development kit (SDK) is free and open-source software, released under the terms of the Eclipse Public License, although it is incompatible with the GNU General Public License. It was one of the first IDEs to run under GNU Class path and it runs without problems under IcedTea.

2.7 Development Kit

The development kit contains JDK (Java Development Kit), Android SDK(Software Development Kit),XML(Extensible Mark-up Language).

2.7.1 Java Development Kit (JDK)

The **Java Development Kit** (**JDK**) is an implementation of either one of the Java Platform, Standard Edition; Java Platform, Enterprise Edition or Java Platform, Micro Edition platforms released by Oracle Corporation in the form of a binary product aimed at Java developers on Solaris, Linux, Mac OS X or Windows. The JDK includes a private JVM and a few other resources to finish the development of a Java

Application. Since the introduction of the Java platform, it has been by far the most widely used Software Development Kit (SDK) On 17 November 2006, Sun announced that they would release it under the GNU General Public License (GPL), thus making it free software. This happened in large part on 8 May 2007, when Sun contributed the source code to the OpenJDK.

The JDK has as its primary components a collection of programming tools, including:

- appletviewer – this tool can be used to run and debug Java applets without a web browser
- apt – the annotation-processing tool
- extcheck – a utility that detects JAR file conflicts
- idlj – the IDL-to-Java compiler. This utility generates Java bindings from a given Java IDL file.
- jabswitch – the Java Access Bridge. Exposes assistive technologies on Microsoft Windows systems.
- java – the loader for Java applications. This tool is an interpreter and can interpret the class files generated by the javac compiler. Now a single launcher is used for both development and deployment. The old deployment launcher, jre, no longer comes with Sun JDK, and instead it has been replaced by this new java loader.
- javac – the Java compiler, which converts source code into Java bytecode
- javadoc – the documentation generator, which automatically generates documentation from source code comments
- jar – the archiver, which packages related class libraries into a single JAR file. This tool also helps manage JAR files.
- javafxpackager – tool to package and sign JavaFX applications
- jarsigner – the jar signing and verification tool
- javah – the C header and stub generator, used to write native methods
- javap – the class file disassembler

2.7.2 Android SDK

The Android software development kit (SDK) includes a comprehensive set of development tools.[4] These include a debugger, libraries, a handset emulator based on QEMU, documentation, sample code, and tutorials. Currently supported development platforms include computers running Linux (any modern desktop Linux distribution), Mac OS X 10.5.8 or later, and Windows 7 or later. As of March 2015, the SDK is not available on Android itself, but software development is possible by using specialized Android applications.

Until around the end of 2014, the officially supported integrated development environment (IDE) was Eclipse using the Android Development Tools (ADT) Plugin, though IntelliJ IDEA IDE (all editions) fully supports Android development out of the box, and NetBeans IDE also supports Android development via a plugin. As of 2015, Android Studio, made by Google and powered by IntelliJ, is the official IDE; however, developers are free to use others. Additionally, developers may use any text editor to edit Java and XML files, then use command line tools (Java Development Kit and Apache Ant are required) to create, build and debug Android applications as well as control attached Android devices (e.g., triggering a reboot, installing software package(s) remotely).

Enhancements to Android's SDK go hand in hand with the overall Android platform development. The SDK also supports older versions of the Android platform in case developers wish to target their applications at older devices. Development tools are downloadable components, so after one has downloaded the latest version and platform, older platforms and tools can also be downloaded for compatibility testing.

Android applications are packaged in .apk format and stored under /data/app folder on the Android OS (the folder is accessible only to the root user for security reasons). APK package contains .dexfiles[(compiled byte code files called Dalvikexecutables), resource files, etc.

2.7.3 XML (Extensible Markup Language)

In computing, **Extensible Markup Language** (**XML**) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The W3C's XML 1.0 Specification and several other related specifications—all of them free open standards—define XML.

The design goals of XML emphasize simplicity, generality, and usability across the Internet. It is a textual data format with strong support via Unicode for different human languages. Although the design of XML focuses on documents, the language is widely used for the representation of arbitrary data structures such as those used in web services. Several schema systems exist to aid in the definition of XML-based languages, while programmers have developed many application programming interfaces (APIs) to aid the processing of XML data.

2.8 Aims and Objectives

The main objective of Aero-Check application is to view air quality information of different regions of India on a Google map. User can switch between different cities to check level of pollution. A quality of air is calculated using Air Quality Index (AQI).

i. Design and implementation of android application which makes air pollution monitoring eligible.
ii. Easy access of application to users.
iii. Users can view the status of the air while travelling to different regions.

CHAPTER 3
PROPOSED APPROACH AND SYSTEM ARCHITECTURE

A system architecture is a conceptual model that defines the structure, behaviour and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviours of the system. In this chapter different modules of the application are discussed.

3.1 System Design

Module 1: Updating of AQI data

The AQI data will be fetched from Central Pollution Control Board (CPCB). This data will be then uploaded on server.

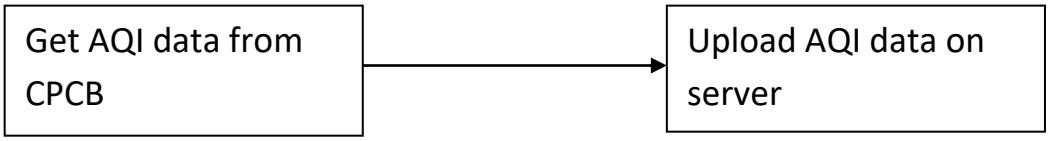

Fig. 3.1 Updating of AQI data

Module 2: Graphical representation of data

When user opens Aero-Check application then an application will fetch AQI data from the server. When user clicks on any city then graphical representation of AQI data will be shown on the graph.

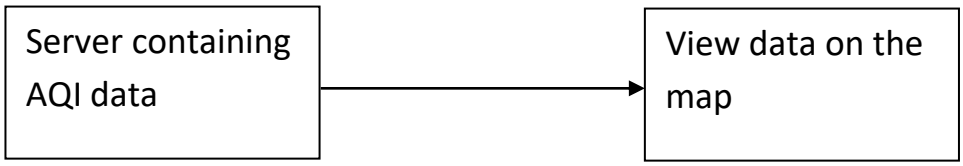

Fig. 3.2 Fetching of AQI data

3.1.1 Data Flow Diagram

Data flow diagrams are used in designing and documenting simple processes or programs. Like other types of diagrams, they help visualize what is going on and thereby help understand a process, and perhaps also find flaws, bottlenecks, and other less-obvious features within it. There are many different types of flowcharts, and each type has its own repertoire of boxes and notational conventions. The two most common types of boxes in a flowchart are:

- A processing step, usually called activity, and denoted as a rectangular box
- A decision, usually denoted as a diamond.

A flowchart is described as "cross-functional" when the page is divided into different swimlanes describing the control of different organizational units. A symbol appearing in a particular "lane" is within the control of that organizational unit. This technique allows the author to locate the responsibility for performing an action or making a decision correctly, showing the responsibility of each organizational unit for different parts of a single process

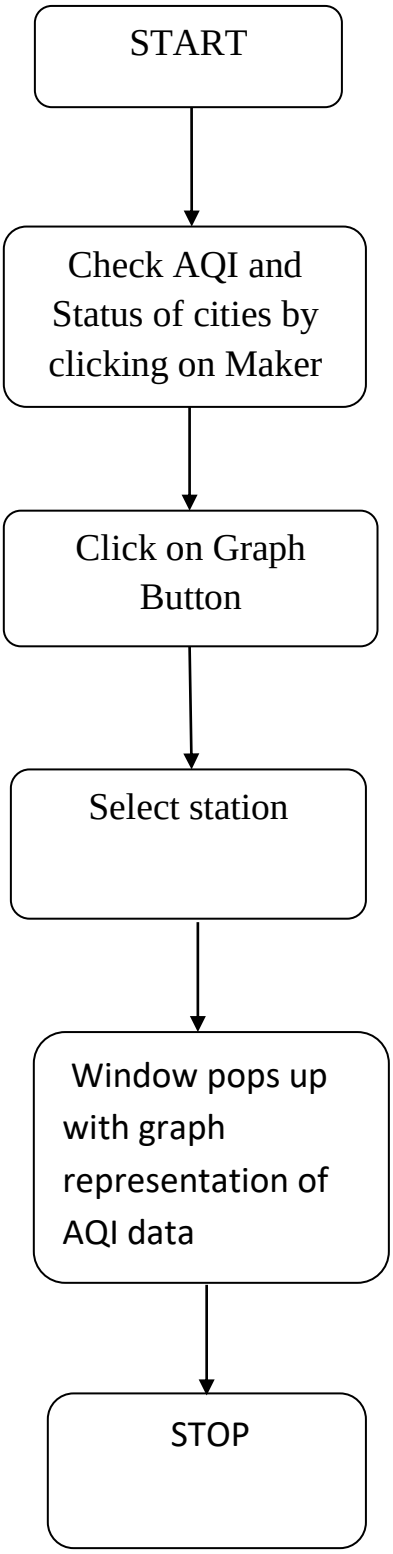

Fig. 3.3 Data flow diagram

3.2 Development and Deployment Requirements

Aero-Check application is developed using Android Studio and Android smartphone is used for deployment. The requirement for development and deployment of application are as follows:

3.2.1 Development Requirements

Software Requirements

- Operating System : Windows
- Programming Language : Java and J2EE
- Database Technology : SQLite
- Developing Environment : Android Studio

Hardware Requirements

Hardware requirements refer to the minimum capability of a system must have to execute this particular application.

- RAM : 3 GB + 1GB for Emulator
- Memory : 2 GB minimum disk space

3.2.2 Deployment Requirements

Smartphone Requirements

- Operating System : Android
- Android Version : Lollipop and above
- Memory : 1 GB
- RAM 1 GB

CHAPTER 4
IMPLEMENTATION

The application aims at giving prior information about air quality to users. Air is a dynamic medium there is so much about it that people might be unaware about. The implementation of this project requires, firstly the AQI data should be ready and providing it with a database. The mobile application should be equipped with an easy user friendly interface. The application should be promoted to applicator under various schemes and should be made freely available. Also using a mobile application with a database which is regularly updated will help the people updated with the level of pollution of the area.

Features of Aero-Check application is as follows

- Aero-Check is implemented using Google API.
- Aero-Check shows AQI value of different regions on Map.
- Aero-Check shows graphs of pollutants responsible for Air Pollution.

XML was used for basic implementation of buttons which plays an important role to use the application in an easy way. For proper linking of that XML buttons JAVA is used for creating objects. With the help of these objects every XML buttons are easily linked with each other and user can easily access it without any problem. These all i.e XML and JAVA and GOOGLE map API are the front end of Aero-Check. SQLite is used for creating back end.

4.1 onCreate( ) Method

onCreate( ) called when the activity is first created. This is where all of normal static set up was done: created views, bind data to lists, etc. This method also provides with a bundle containing the activity's previously frozen state, if there was one.

4.2 setContentView(View view)

Set the activity content to an explicit view. This view is placed directly into the activity's view hierarchy. It can itself be a complex view hierarchy. When calling this method, the layout parameters of the specified view are ignored. Both the width

and the height of the view are set by default to ViewGroup+LayoutParams.MatchParent.

4.3 Intent

An intent is an abstract description of an operation to be performed. It can be used with startActivity to launch an Activity, broadcastIntent to send it to any interested BroadcastReceiver components,
and startService(Intent) or bindService(Intent, ServiceConnection, int) to communicate with a background Service.

An Intent provides a facility for performing late runtime binding between the code in different applications. Its most significant use is in the launching of activities, where it can be thought of as the glue between activities. It is basically a passive data structure holding an abstract description of an action to be performed.

4.4 Layouts

A layout defines the visual structure for a user interface, such as the UI for an activity or app widget. A layout can be declared in two ways:

- **Declare UI elements in XML**. Android provides a straightforward XML vocabulary that corresponds to the View classes and subclasses, such as those for widgets and layouts.

- **Instantiate layout elements at runtime**. Users application can create View and ViewGroup objects (and manipulate their properties) programmatically.

The Android framework gives the flexibility to use either or both of these methods for declaring and managing user's application's UI. For example, user could declare its application's default layouts in XML, including the screen elements that will appear in them and their properties. One could then add code in its application that would modify the state of the screen objects, including those declared in XML, at run time.

- User should also try the Hierarchy Viewer tool, for debugging layouts — it reveals layout property values, draws wireframes with padding/margin indicators, and full rendered views while user debug on the emulator or device.

- The layoutopt tool lets user quickly analyse its layouts and hierarchies for inefficiencies or other problems.

The advantage to declaring user UI in XML is that it enables to better separate the presentation of users application from the code that controls its behavior. Users UI descriptions are external to users application code, which means that user can modify or adapt it without having to modify its source code and recompile. For example, XML layouts can be created for different screen orientations, different device screen sizes, and different languages. Additionally, declaring the layout in XML makes it easier to visualize the structure of our UI, so it's easier to debug problems. As such, this document focuses on teaching us how to declare our layout in XML. If anyone interested in instantiating View objects at runtime, refer to the ViewGroup and View class references.

In general, the XML vocabulary for declaring UI elements closely follows the structure and naming of the classes and methods, where element names correspond to class names and attribute names correspond to methods. In fact, the correspondence is often so direct that can be guessed what XML attribute corresponds to a class method, or guess what class corresponds to a given XML element. However, note that not all vocabulary is identical. In some cases, there are slight naming differences. For example, the EditText element has a text attribute that corresponds to EditText.setText().

4.5 Manifest

Every application must have an AndroidManifest.xml file (with precisely that name) in its root directory. The manifest file provides essential information about app to the Android system, which the system must have before it can run any of the app's code.

Among other things, the manifest file does the following:

- It names the Java package for the application. The package name serves as a unique identifier for the application.
- It describes the components of the application, which include the activities, services, broadcast receivers, and content providers that compose the application. It also names the classes that implement each of the components and publishes their capabilities, such as the Intent messages that they can handle. These declarations inform the Android system of the components and the conditions in which they can be launched.

- It determines the processes that host the application components.
- It declares the permissions that the application must have in order to access protected parts of the API and interact with other applications. It also declares the permissions that others are required to have in order to interact with the application's components.
- It lists the Instrumentation classes that provide profiling and other information as the application runs. These declarations are present in the manifest only while the application is being developed and are removed before the application is published.
- It declares the minimum level of the Android API that the application requires.
- It lists the libraries that the application must be linked against.

4.6 Database

Administrator will be directed to the server. Administrator is the person who will look after the every activity carried out by the server and different permission requested by the client. It will provide the authorization to the registered client for the further assessment. Administrator can enable or disable any client for accessing the service.

In this module the design and implementation of database is carried out.SQL tool was for database creation. All the necessary data about the air quality, different particulates present in air.

SQLite database was used for this project .The name of database is ‘Aero’ and the name of table is ‘info’ which has 6 attributes i.e. Id, Name Of Station, PM10, SO2, NO2, PM2.5, CO, O3 and AQI .Following commands were used for accessing the database.

A] **To create a database:**

db = openOrCreateDatabase(“Aero”, MODE_PRIVATE, null);

B] **To create a table:**

db.execSQL(“create table if not exists info(id integer, city varchar, pm10 integer,so2 integer, no2 integer, pm2.5 integer, nh3 integer, co integer, o3 integer, aqi integer)”;

C] **To insert a value:**

db.execSQL("insert into info values('1', ' NSIT Dwarka', '0', '25', '49', '176', '0', '31', '53', '176');

D] **To fetch a value:**

db.rawQuery("select * from info,null");

Table 4.1. Structure of database

Attribute	**Type**
Id	Integer
City	Varchar
PM10	Integer
SO2	Integer
NO2	Integer
PM2.5	Integer
NH3	Integer
CO	Integer
O3	Integer
AQI	Integer

Table 4.2. Database of pollutants recorded on 5/03/2017

ID	City	PM10	SO2	NO2	PM2.5	NH3	CO	O3	AQI
1	NSIT Dwarka		25	49	176		31	53	176
2	Shadipur		21	33	188		108	57	190
3	IHBAS		12	81	-	6	34		
4	Mandir Mark-DPCC	128	13	55	162	6	72	38	165
5	AnandVihar-DPCC	236	30	121	205	12	106	14	241
6	R.K.Puram-DPCC	207	40	65	205	5	60	253	241
7	Punjabi Bagh-DPCC	163	34	77	215	8	44	169	216
8	ITO	139		53	206	16			205
9	DTU			22	253	1			252

10	Sirifort			77	186	10			186
11	Civil Lines, Nagpur	104	24	50	113	7	31	192	206
12	More chowk,waluj, Aurangabad	116	4	91	68	5	56		247
13	Bandra West, Mumbai	102	21	35	61		47	44	105
14	K.T.H.M. College, Nashik	68	8	62	102	9	35	155	151
15	Karve Road, Pune	67	41	51	62		6	82	95
16	PimpleshwarMandir, Dombivli,Thane	145	45	31	154	2	55		153
17	Airoli,NaviMumbai	70	24				39		70
18	Chandrapur	127	11	16	131	8	39	8	133
19	Solapur	124	24	44	89		106	41	124
20	GVMC,RamNagar,Visakhapatnam	75	14	38	57	2	32	25	73
21	Tirumala,Tirupati	7	14	100	17	2	18	7	101
22	IGSC Planetarium Complex,Patna		47	134	282		44	30	263
23	MuzaffarpurCollectorate		15	24	240		47	2	241
24	Gaya Collectorate		3	8	330		30	3	330
25	Maninagar,Ahmedabad		58	77	234		23	84	253
26	Sector 16A,Faridabad		5	32	160		41	49	260
27	Sector 6,Panchkula		10	33	60		28	105	112
28	VikasSadan,Gurgaon		10	22	138		26	75	143
29	MD University,Rohtak		3	23	67		14	38	67
30	BTM layout,Bengaluru		7	20	31		23	45	65
31	Peenya,Bengaluru		9	36	59		7	23	56
32	BWSSB Kadabesanahalli		1	24	23		24	26	39
33	City Railway Station-KSPCB	73	21	67			36		74
34	SaneguruvaHalli-KSPCB	40	6	52			28		51
35	VK Industrial Area,Jaipur	119	22	58	114	1	49		120
36	Collectorate,Jodhpur	142	17	38	186		37	39	186
37	Alandur Bus Depot		3	19	48		35	17	52
38	IIT		3	15	37		52	22	51
39	Manali		4	24	38	17	23		42
40	Sanathnagar,Hyderabad		6	36	57		24	60	87

41	Zoo Park,BahadurpuraWest,Hyderabad	92	13	32	83	5	25		108
42	IDA Pashamylaram-TSPCB,Hyderabad	78	17	39	60	4	25		78
43	Bollaram Industrial Area,Hyderabad	99	37	29	86	9	34	61	99
44	ICRISAT Patancheru,Hyderabad	68	17	14	54	5	34	46	68
45	Sanjay Palace,Agra		7	35	149		29	41	149
46	Nehru Nagar,Kanpur		3	51	252		67	6	240
47	Lalbagh,WestLucknow		9	50	165		65	191	168
48	Central School,Lucknow		6	25	205		45	100	213
49	TalkatoraDistrict Industries Center		13	144	247		41	48	248
50	ArdhaliBazar,Varanasi	195	7	27	275		27	15	273
51	Victoria,Kolkata	49	14	31			24	25	431
52	RabindraBharatiUniversity,Kolkata	98	8	36			42	21	98
53	Howrah	66		45			35	22	60
54	Haldia						27	26	
55	SindhuKanhu Indoor Stadium,Durgapur	185	21	54			19	32	185

Table 4.3. Air pollution impacts

AQI	**Associated Health Impacts**
Good (0-50)	Minimal impact
Satisfactory (51-100)	May cause minor breathing discomfort to sensitive people.
Moderately polluted (101–200)	May cause breathing discomfort to people with lung disease such as asthma, and discomfort to people with heart disease, children and older adults
Poor (201-300)	May cause breathing discomfort to people on prolonged exposure, and discomfort to

	people with heart disease
Very poor (301-400)	May cause respiratory illness to the people on prolonged exposure. Effect may be more pronounced in people with lung and heart diseases.
Severe (401-500)	May cause respiratory impact even on healthy people, and serious health impacts on people with lung/heart disease. The health impacts may be experienced even during light physical activity.

4.7 Bar Graph

Bar Graph is used in Aero-Check to display the pollutant's contribution in particular region. Bar Graph is implemented using MPAndroidChart API. MPAndroidChart is a powerful & easy to use chart library for Android. It runs on API level 8 and upwards.

Usage

Gradle dependency

- Following repositories are added in build.gradle:

```
allprojects{
        repositories{
                maven{ url "https://jitpack.io" }
        }
}
```

- Following dependencies are added in build.gradle:

```
dependencies {
        compile 'com.github.PhilJay:MPAndroidChart:v3.0.1'
}
```

4.8 Map Activity

Map plays major role in Aero-Check application. It displays markers with status of air quality.

- Install the Google Play services SDK.
- Create a Google Maps project.
- Get a Google Maps API key.
- Following code is required to add marker and move the camera.

```
   public void onMapReady(GoogleMap googleMap) {
      mMap = googleMap
      LatLng sydney= new LatLng(-34, 151);
      mMap.addMarker(new MarkerOptions().position(nagpur).title("Marker
in Sydney"));
      mMap.moveCamera(CameraUpdateFactory.newLatLng(sydney));
   }
}
```

CHAPTER 5
RESULT AND DISCUSSION

In the previous chapter the information related to the application is provided which consists of each and every details of the project. The software tools required to build the project along with its functional and non-functional requirements. Study about java and xml for coding required for the implementation.

Following are the screen snapshots of some of application parts shown on the android smartphone at different interval of time:

5.1 Home Page

When user clicks on the Aero-Check application icon on their Android smartphone, Fig. 5.1 is displayed which is Home Page or the first interface of application. Total 55 cities with markers are displayed on the screen.

Fig. 5.1 Home page

5.2 Air Quality Status

When user clicks on any of this markers e.g. Ahmedabad, the AQI values and Status of air pollution is displayed i.e. 253 and Poor respectively as shown in Fig. 5.2:

Fig. 5.2 Air quality status

5.3 Stations

A Graph Button is present at the top right of the screen. When user clicks on this Graph Button, List View with 55 Cities is displayed as shown in Fig. 5.3.

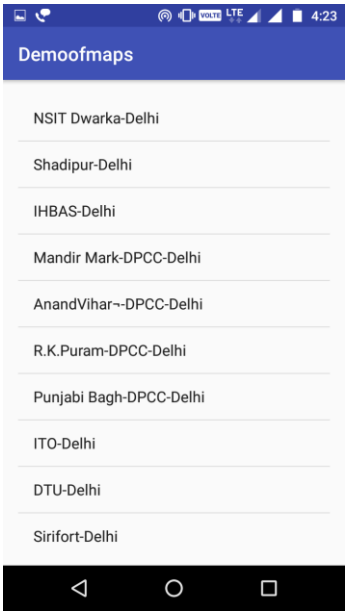

Fig. 5.3 Stations

5.4 Graph View

After clicking on any of the Cities shown in List View, the graph is shown with data of different particulate contents present in air as shown in Fig. 5.4.

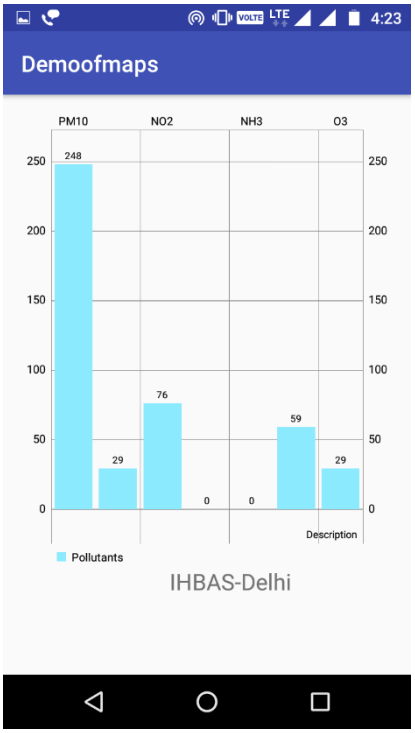

Fig. 5.4 Graph view

CHAPTER 6
CONCLUSION

Aero-Check application gives information about Air Quality of the area by acquiring AQI data from Central Pollution Control Board (CPCB). CPCB have installed 530 Air Quality stations in 175 cities in 26 States and 5 Union territories under National Air quality Monitoring Program. This application is not only used by common man but also by Government agencies can keep track air quality to monitor industries and traffics and run pollution prevention programs and energy conservation efforts. Like Weather reports, News agencies can report Air Quality of respective regions. Common man can view the status of the air while travelling to different regions.

Android has growing selection of third Party applications. While android operating system is mostly used operating system for mobile technology. Aero-Check is so implemented using Android.

6.1 Limitations of the Study

Every project has got some limitations. Aero-Check has also some limitations. Aero-Check does not have any centralised database .Moreover it does not deal with any real time data. Aero-Check is limited to India only. In India it shows details of air quality of specific cities only where CPCB has already installed sensors. There is no alert system in this application. It does not do analysis of data just display it on graph.

6.2 Future Scope of Work

Every project has future scope for further development. To overcome the limitations of the Aero-Check application this project can be extended in future. This application can be made real time if CPCB allows to access their data. If the Government increases more number of stations then this application can be more efficient and can give information about more number of regions. And also if the sensors are installed in taxis and public transport then the places where stations are not installed still their AQI can also be shown in the map.

References

1. George Joseph (2005), “Fundamentals of Remote Sensing”, Universities Press (India) Private Limited, Second Edition, ISBN-13: 978-81-7371-535-8
2. Dr. B.C.Panda (2009), “Remote Sensing Principles and Applications”, Viva books Private Limited, ISBN-13: 978-81-7649-630-8
3. John Horton (2015), “Android Programming for Beginners”, Pact Publishing Ltd, ISBN-13: 978-1-78588-326-5
4. Herbert Schild (2016), “Java – The Complete Reference”, McGraw Hill Education, Ninth Edition, ISBN-13: 978-1-25900-246-5
5. www.timesofindia.indiatimes.com/taj_mahal_effect accessed on 02/03/2017 at 8:15 pm
6. www.stackoverflow.com/questions/31915026/how-can-i-make-a-listview-in-android-studio accessed on 03/03/2017 at 6:15 pm
7. https://github.com/PhilJay/MPAndroidChart accessed on 06/03/2017 at 6:15 pm
8. developer.android.com/training/maps/index.html accessed on 03/03/2017 at 8:15pm
9. www.cpcb.nic.in/aqi_report.php accessed on 05/03/2017 at 2:30 pm
10. www.wikipedia.org/Air_quality_index accessed on 02/03/2017 at 8:00 pm