AStyle Guidelines for Final Year Project ReportsQI Predictor

Final Year Project – Final Report

Session 2016-2020

A 4th Year Student

A project submitted in partial fulfilment of the

COMSATS University Islamabad, Lahore Campus Degree

of

BSc. (Hons.)BS in Computer Science (CUI)



Department of Computer Science

COMSATS University Islamabad, Lahore Campus

12 May 2020

# Evaluation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Project Group Members** (To be filled by students) | | | | (To be filled by supervisor) |
| Sr.# | Reg. # | Student Name | \*Signature | Obtained Marks  (Total Marks: 10) |
| (i) | FA16-BCS-047 | Waqas Ahmad |  |  |
| (ii) | FA16-BCS-041 | Hamza Sadiq |  |  |
| (iii) | FA16-BCS-162 | Ali Hassan |  |  |

\*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.

Supervisor Name: Dr. Usama Ijaz Bajwa\_\_\_\_\_\_\_ Supervisor Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Remarks (if any): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Abstract**

Our project, AQI Predictor, is an app to predict the quality of Air Aka Air Quality Index (AQI) Level using pictures from the camera or gallery of any android device or through web. The app uses a pre-trained model that’s embedded within the app. The pre trained model is a Convolutional Neural Network that uses machine learning techniques to help us in predicting the Air Quality level correctly. The model is trained on a relevant dataset in order to achieve the accuracy. Along with the app, a website will be created that will be synchronized with the app and the data that will be collected in order to show a map of the areas the AQI has been calculated for. This will help people in deciding which areas have what kind of air quality. The quality of air will be predicted by the application providing us Air Quality Level which will further tell us more details such as if the air is harmful or good and how necessary precautions can be taken. These factors and results will help us and the authorities including the Government and relevant governmental and other bodies decide certain measures to make the air quality better.

**Acknowledgement**

It would be unfair and unkind not to acknowledge the guidance, intelligence and patience Dr. Usama Ijaz Bajwa has shared with us while working on this project. We are incredibly grateful to our supervisor for his vision and his persistence upon the values of hard work, helping others and not losing morale.

We would also like to thank our institute, Comsats University Islamabad, Lahore Campus, for their providing us a platform to work on as well as helping us in any way possible. This would not have been possible without it.

Our heartiest gratitude to Dr. Usama Ijaz Bajwa, colleagues and everybody who helped us in anyway with their thoughts, skills or any productive way.

Table of Contents

[Evaluation 2](#_Toc40235074)

[1 Introduction 10](#_Toc40235075)

[1.1 Introduction 10](#_Toc40235076)

[1.2 Objectives 11](#_Toc40235077)

[1.3 Problem Statement 12](#_Toc40235078)

[1.4 Assumptions and constraints 12](#_Toc40235079)

[1.4.1 Assumptions 12](#_Toc40235080)

[1.4.2 Constraints 12](#_Toc40235081)

[1.5 Project Scope 13](#_Toc40235082)

[2 Requirement Analysis 15](#_Toc40235083)

[2.1 Literature Review 15](#_Toc40235084)

[2.1.1 Air Visual 15](#_Toc40235085)

[2.1.2 Air Cognizer 17](#_Toc40235086)

[2.1.3 Research papers 17](#_Toc40235087)

[2.2 Stakeholders list 18](#_Toc40235088)

[2.2.1 Admin 18](#_Toc40235089)

[2.2.2 Users 19](#_Toc40235090)

[2.2.3 Government of Pakistan and relevant authorities 19](#_Toc40235091)

[2.2.4 Any private organization or students 19](#_Toc40235092)

[2.3 Requirements elicitation 19](#_Toc40235093)

[2.3.1 Functional Requirements 19](#_Toc40235094)

[2.3.2 Non-functional requirements 20](#_Toc40235095)

[2.4 Use case descriptions and designs 22](#_Toc40235096)

[2.4.1 App Side 22](#_Toc40235097)

[2.4.2 Design 23](#_Toc40235098)

[2.4.3 Server side description 24](#_Toc40235099)

[2.4.4 Server side design 25](#_Toc40235100)

[2.5 Software development life cycle model 26](#_Toc40235101)

[3 System Design 29](#_Toc40235102)

[3.1 Work breakdown structure 29](#_Toc40235103)

[3.2 Sequence Diagram 30](#_Toc40235104)

[3.2.1 Taking picture, sending to application and getting Air Quality level results: 30](#_Toc40235105)

[3.2.2 Training Sequence Diagram 31](#_Toc40235106)

[3.2.3 Web Server Sequence Diagram 32](#_Toc40235107)

[3.3 High Fidelity Diagrams of the System 33](#_Toc40235108)

[3.3.1 Application: 34](#_Toc40235109)

[3.3.2 Website 38](#_Toc40235110)

[3.4 Database diagram 44](#_Toc40235111)

[3.5 Network Diagram (Gantt chart) 45](#_Toc40235112)

[4 Testing 47](#_Toc40235113)

[4.1.1 Air Quality Predictor (Android application) 47](#_Toc40235114)

[4.1.2 Air Quality Predictor (Website) 49](#_Toc40235115)

[4.1.3 Showing notifications to the user 51](#_Toc40235116)

[4.1.4 Admin/Server End 53](#_Toc40235117)

[5 Conclusion 56](#_Toc40235118)

[5.1 Project summary 56](#_Toc40235119)

[5.2 Future work 56](#_Toc40235120)

[6 References 58](#_Toc40235121)

List of Tables

[Table 1-FR01-Calculating AQI 19](#_Toc40235181)

[Table 2- FR02-Showing the AQI and information to the User 19](#_Toc40235182)

[Table 3-FR03-Uploading the data to the server 20](#_Toc40235183)

[Table 4-FR04-Checking out data and information 20](#_Toc40235184)

[Table 5-NFR01-User Friendly 20](#_Toc40235185)

[Table 6-NFR03-Simple and minimalistic design 20](#_Toc40235186)

[Table 7-NFR03-Portability 20](#_Toc40235187)

[Table 8-NFR04-Documentations and help 21](#_Toc40235188)

[Table 9-NFR05-Responsive and fast 21](#_Toc40235189)

[Table 10-NFR06-Accuracy 21](#_Toc40235190)

[Table 11-Use Case-App Side 22](#_Toc40235191)

[Table 12-Use Case-Server Side 24](#_Toc40235192)

[Table 13-Gantt Chart 45](#_Toc40235193)

[Table 14 – How the user can view the application 47](#_Toc40235194)

[Table 15 – How the AQ Level is predicted 48](#_Toc40235195)

[Table 16 – How the user views the website 49](#_Toc40235196)

[Table 17 – How the AQ Level is predicted on website 50](#_Toc40235197)

[Table 18 – Showing user the results 51](#_Toc40235198)

[Table 19 – Showing results to User on website 52](#_Toc40235199)

[Table 20 - Admin can view saved images on Drive 53](#_Toc40235200)

[Table 21 - Admin can see saved values in Database 54](#_Toc40235201)

**List of Figures**

[Figure 2‑1 – Air Visual – Displaying AQI 15](#_Toc40235242)

[Figure 2‑2 – Air Cognizer app 17](#_Toc40235243)

[Figure 2‑3 – Use case: App Side 23](#_Toc40235244)

[Figure 2‑4 Use case: Server side 25](#_Toc40235245)

[Figure 2‑5 – SDLC Model 26](#_Toc40235246)

[Figure 3‑1 – Work Break Down Structure Diagram 29](#_Toc40235247)

[Figure 3‑2 – Sequence diagram – Taking Picture 30](#_Toc40235248)

[Figure 3‑3 – Sequence Diagram – Training 31](#_Toc40235249)

[Figure 3‑4 – Sequence Diagram - Web Server 32](#_Toc40235250)

[Figure 3‑5 - Main/Home Page of the application 34](#_Toc40235251)

[Figure 3‑6 - Results page of the application 36](#_Toc40235252)

[Figure 3‑7- Website Home Page 39](#_Toc40235253)

[Figure 3‑8- AQI Teller Page of Website 39](#_Toc40235254)

**CHAPTER #1**

**INTRODUCTION**

# Introduction

## Introduction

Air Quality is a major and serious issue in the World, which has been picking pace in getting identified in World’s major cities in the past few years. A lot of International Organizations have started researches alongside independent scholars and research groups that have helped in identifying the problems and proposing small scale solutions as well as large scale solutions.

Declining air quality index has been in progress for a few years, and major cities like Paris, Lahore, New Delhi, Beijing, and so many others have suffered from this. For a while, it wasn’t acknowledged, but now it is, and it has been causing a lot of problems. Many International Organizations have worked to mitigate the issues arisen from the Air Quality Index (AQI). With all of the major problems, it takes a lot of time for the solutions, knowledge, and the problems to actually mature enough in order for it to work out.

It’s only natural to consider and think of a solution that is portable, inexpensive, and easily implementable. A mobile app would certainly solve such issues and makes lives easier if it can predict the Air Quality level of a place just through a picture.

Now in Pakistan, there is no application or software available which can judge the Air Quality level of our environment and help us take specific measures. We decided to build an app that can judge the Air Quality Level of the local area just by a picture. It is not only frugal but also user-friendly as compared to hardware-based solutions. Most of the hardware devices which measure AQI need prior knowledge of the device to operate it and are expensive to purchase and operate.

The Air Quality level is a term used to describe the categories the AQI is divided into. There are three levels:

1. Good: 0-74
2. Average: 75-115
3. Bad/Hazardous: >115

Our application judges the AQI of a place by categorizing it into one of three levels and then displaying the level to the user.

The front end of our main system (the app) is developed in Java using the Android SDK. It has a simple interface that gives users the option to either capture a picture or use one from their gallery to process. The interface can, of course, be changed later on to add more features and sophistication. It works in conjunction with the model and the database that will be linked to it at the backend. The app processes the picture to calculate the Air Quality level from the picture by handing the picture over to the model that’s trained on the dataset we will have given it. The data is saved in our database, increasing our dataset for the model to be trained further for higher accuracy if required. The app works on the principle of crowdsourcing. The users being the source for our constantly increasing dataset.

Coming onto the model, we trained the model using the concepts of Neural Networks. We have extracted features from the dataset and create feature vectors from the trained images. This set of features is used to train the model. The model trained itself on this vast dataset and then predicts the AQ level. It returns an integer value that tells us which level it is:

* 0 is average
* 1 is bad/hazardous
* 2 is good.

By getting the user’s location, we can also tell the temperature as well as humidity level of that specific location by obtaining information from another source.

Secondly, we created a website that works in synchronization with all the data that the database has at the backend. The site is linked to our database. It will show all the data and results that will be calculated by the app, which can be useful for the average citizen or relevant government institutes/employees. The website also embeds a map which will show the AQIs of different areas that have been predicted by the app.

This accomplishment will help the Government to tackle the areas and alleviate the problems of bad air quality by doing proper plantation or initiating a drive for plantation. Govt. can also hinder the working of factories which emit harmful chemicals in that area until air quality reaches a normal level.

## Objectives

There are many countries that have been affected by the poor quality of air that is the result of many external factors such as poor waste management, burning of various materials that harm the air and excessive smoke produced by vehicles and factories, etc. There are no cheap solutions as to measuring and preserving the air quality.

Our main objective was to create an android app that will measure the Air Quality level just by taking a picture. A pre-trained model is used alongside which will help the app in determining the air quality level.

The app functions just by taking a picture and using it in the app for processing. Firstly it is checked if sky exists in the picture or not by using panoptic segmentation model to detect sky. If sky exists, then the app uses the model to process the picture which will result in generating an Air Quality level value which will tell us if the Air Quality is good, bad or average.

This project will then help people to determine Air Quality levels of any location they want. The values with the location will then be uploaded to our server that will widen the dataset. This can further be used in future for retraining the model to increase the accuracy. This dataset will integrate seamlessly with the website as well providing the website and its visitors with the live predicted values of wherever it’s being used. This can later be used for many causes such as determining if the Air Quality of a specific area is harmful or not.

## Problem Statement

An AQI Predictor system is developed to find out the Air Quality level of any place by just a picture that can be taken by any Android device or through the website. It is developed using deep learning neural networks.

## Assumptions and constraints

The assumptions and constraints of the system are as follows:

### Assumptions

* The user must have an Android smartphone that can run the app.
* The user’s smartphone must have a functioning camera.
* The user must have an internet connection if accessing through the website.

### Constraints

* Our Android application will run on Android version Android 6.0.1 or higher.
* The photo that the user will use to calculate Air Quality level must have sky.
* It needs daylight and it will not work during the night.

## Project Scope

Hazardous AQI is a serious issue today especially in Pakistan. The Government and other relevant authorities haven’t been able to manage to bring down the AQI of major cities to a good level. Our system will help the Government and other relevant authorities in checking the Air Quality Level of any certain area in the picture to help them decide how the Air quality of the place is and what necessary measures can be taken in order to control the Air Quality. The information and dataset created by our system can be used by the Government of Pakistan and relevant bodies for further research and studies to reach conclusions.

**CHAPTER #2**

**REQUIREMENT ANALYSIS**

# Requirement Analysis

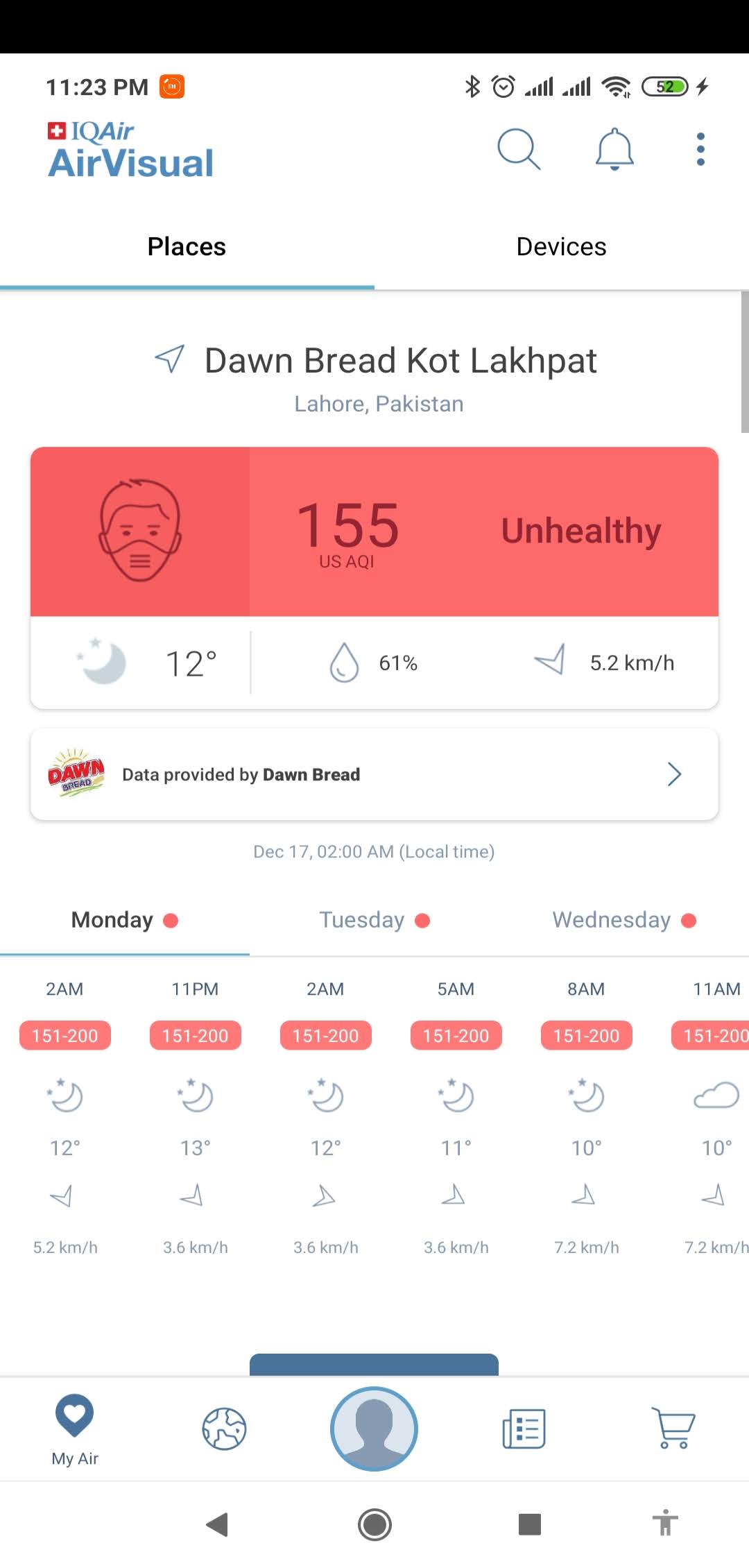
## Literature Review

In Pakistan there is no such mobile or web application that tells its user AQI by just clicking a photo. Here are some related works from around the globe:

### Air Visual

Air Visual (AirVisual, 2019) is a mobile application as well as website that takes PM2.5 values from different sources. It displays to the user PM2.5 value and AQI level of a specific city. It is a platform that only displays values taken through sources rather than an AQI calculating application. These sources are sensors that help in determining PM2.5 values from the specific areas. Air Visual also sells their products in the form of AQI calculation sensors and Air Purifiers.

Screenshots of the application are provided below:



x

Figure 2‑1 – Air Visual – Displaying AQI

Figure 2-2 shows the list of cities ranked in order of their Air Quality Index from worse to better.

### Air Cognizer

Air Cognizer (AirCognizer, 2019) is a mobile application which works in the same manner as our application will. It is limited to Indian region because it was developed by Indian developers and students. It’s not available in Pakistan. It also takes a picture as input through user’s phone camera or gallery and then uses an embedded model in order to determine AQI and display it to the user. It uses similar methodology with our application.

Screenshot of its main page: below:

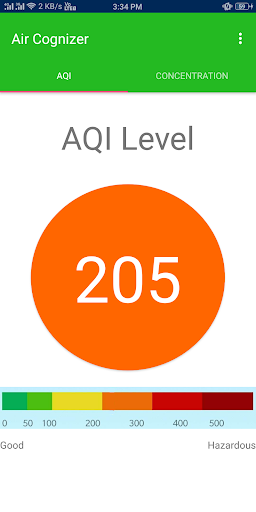


Figure 2‑2 – Air Cognizer app

Figure 2-3 shows that the Air Cognizer app displays the AQI of the place in the picture.

### Research papers

There are a few research papers available on the internet which are relevant to our project.

#### Air pollution forecasting from sky images with shallow and deep classifiers

This research paper (Vahdatpour, Sajedi, & Ramezani, 2018) talks about how to extract information from images that can be used to detect or predict Air Quality Index of the location/atmosphere in that image. In this paper, a method was proposed that predicted the level of air pollution of any location by using an image that was taken from a smartphone’s camera. At the end there were two classifications methods that were implemented in order to gather air pollution information from an image. The second method uses a Convolutional Neural Network (CNN) which is designed to get an image that has sky in it and provides the results in form of level of air pollution. There were some evaluation tests conducted and acceptable accuracy was found through these methods.

#### Air Quality Measurement Based on Double-Channel Convolutional Neural Network Ensemble learning

This research paper (WANG, et al., 2019) proposes a solution to solve the single channelled convolutional neural networks that are used to train images. While the Single channelled CNNs take one feature at a time, the double channel CNNs can take the combined feature extracted from different parts of an image. The “double-channel weighted convolutional network ensemble learning algorithm” was proposed in this paper. In this double channel CNN, each channel is used to train different parts of the image to extract different features. And afterwards, there’s a feature weights self-learning method proposed which weighs and combines the extracted feature vectors to calculate the AQI.

#### Image-based Air Quality Analysis Using Deep Convolutional Neural Network

This research paper (A. Chakma, 2017) discusses how to classify and categorize images into different levels based on their PM2.5 concentrations. The proposed method tells how to implement this in model and get results based on different categories of PM2.5 values which can be used to determine AQI.

## Stakeholders list

The stakeholders list of our system is as follows:

### Admin

The admin will be the top level manager who will manage the whole system, any issues and the data that comes in and goes out.

### Users

The system will help users to predict the Air Quality level using pictures and helping them in suggesting measures.

### Government of Pakistan and relevant authorities

This system will help the Government of Pakistan and the Climate Change ministry in checking the Air Quality level of places and then taking certain measures. Currently they have to use sensors for AQI measurements which are both expensive and rarely available.

### Any private organization or students

Lastly, any other organizations, students or researchers who want to use our application and data for further research and studying can benefit from our system.

## Requirements elicitation

The requirements of the system from the stakeholders’ point of view are as follows:

### Functional Requirements

The functional requirements of the system are mentioned below:

#### FR01 – Calculating the AQI

Table 1-FR01-Calculating AQI

|  |  |
| --- | --- |
| FR01 | The application will take the picture through the camera app on the user’s android phone or through the gallery, then it will calculate its AQI level. |

Table 1 shows as the app takes a picture through user’s input (camera or gallery), the app will calculate the Air Quality level of the picture.

#### FR02 – Showing the AQI and information to the User

Table 2- FR02-Showing the AQI and information to the User

|  |  |
| --- | --- |
| FR02 | Using the trained model embedded in the app, the Air Quality level is calculated. Then the Air Quality level is shown alongside any other information or suggestion to the user. |

Table 2 shows that the Air Quality level has been calculated, then the Air Quality level and additional information are shown to the user on their mobile’s screen.

#### FR03 – Uploading the data to the server

Table 3-FR03-Uploading the data to the server

|  |  |
| --- | --- |
| FR03 | After the first two FRs, whenever the user’s phone connects to the internet through Wi-Fi or mobile data, the data is uploaded to the server. |

Table 3 shows that after the Air Quality level has been calculated and shown to the user, the data is then uploaded on our server when the user’s device is connected to the internet.

#### FR04 – Checking out data and information

Table 4-FR04-Checking out data and information

|  |  |
| --- | --- |
| FR04 | The user can view and use the information that will be uploaded on the server through the website and in future, through app as well. |

Table 4 shows that after the data is uploaded on the server, the user can view it any time they want through the website and the app.

### Non-functional requirements

#### NFR01 – User Friendly

Table 5-NFR01-User Friendly

|  |  |
| --- | --- |
| NFR 01 | The GUI of the application and website will be very user friendly. |

Table 5 shows that our app and other system will be user friendly and very easy to understand.

#### NFR02 – Simple and minimalistic design

Table 6-NFR03-Simple and minimalistic design

|  |  |
| --- | --- |
| NFR 02 | The design of the application will be simple and minimalistic making it extremely easy to check out all the options and use them. |

Table 6 shows that our app will have an ultra-simple and minimalistic design which will help in transparency and understanding.

#### NFR03 – Portability

Table 7-NFR03-Portability

|  |  |
| --- | --- |
| NFR 03 | The application will be usable on any device with an Android version of 6.0.1 or higher. |

Table 7 shows that our app will be usable on any android device which has an android version of 6.0.1 Marshmallow or higher.

#### NFR04 – Documentations and help

Table 8-NFR04-Documentations and help

|  |  |
| --- | --- |
| NFR 04 | A complete documentation and how to use of the application will be provided. |

Table 8 shows that there will be a complete documentation about the application and the system which shall be provided.

#### NFR05 – Responsive and fast

Table 9-NFR05-Responsive and fast

|  |  |
| --- | --- |
| NFR 05 | The calculation of the Air Quality level will be done within seconds, even on low end devices. The application will be quick to generate results. |

Table 9 shows that our app will be highly responsive and quick to generate results as well as showing them to the user. There’ll be no unnecessary waiting.

#### NFR06 – Accuracy

Table 10-NFR06-Accuracy

|  |  |
| --- | --- |
| NFR 06 | The application will generate results that will be reasonably accurate. |

Table 10 shows that our app will generate results that are accurate up to a very good percentage so that the user can trust the results and the app.

## Use case descriptions and designs

### App Side

Table 11-Use Case-App Side

|  |
| --- |
| **Use case ID:** 01 |
| **Use case name:** AQI Predictor app |
| **Actors:** User, Server |
| **Use Case Summary:** The Air Quality level is determined by the user’s input image through the device’s camera. The Air Quality level will be displayed on the app. |
| **Pre-condition:** Must have app installed, must have android phone with android version 6.0.1 or higher |
| **Course of events:** The user captures a picture, the app will input the picture to the model and the model will send calculated Air Quality level from the picture as an output. This AQI level will be displayed by the app. |
| **Post condition:** Air Quality level is determined by the model and displayed on the app. |
| **Use Case Cross References:** |
| **Includes:** Information, settings, take picture, predict Air Quality level. |
| **Extends:** No GPS Permission, no internet connection. |

Table 11 shows us that the use case “AQI Predictor app” with the use case ID “01” explains the workflow of the application side of the project. This application side is the one that the user is going to deal with. There are two actors in this use case, the user and the server. The user performs majority of the tasks here including taking a picture. Then the picture is run through the app to the model which then predicts the Air Quality level and shows it back to the user on the screen.

### Design

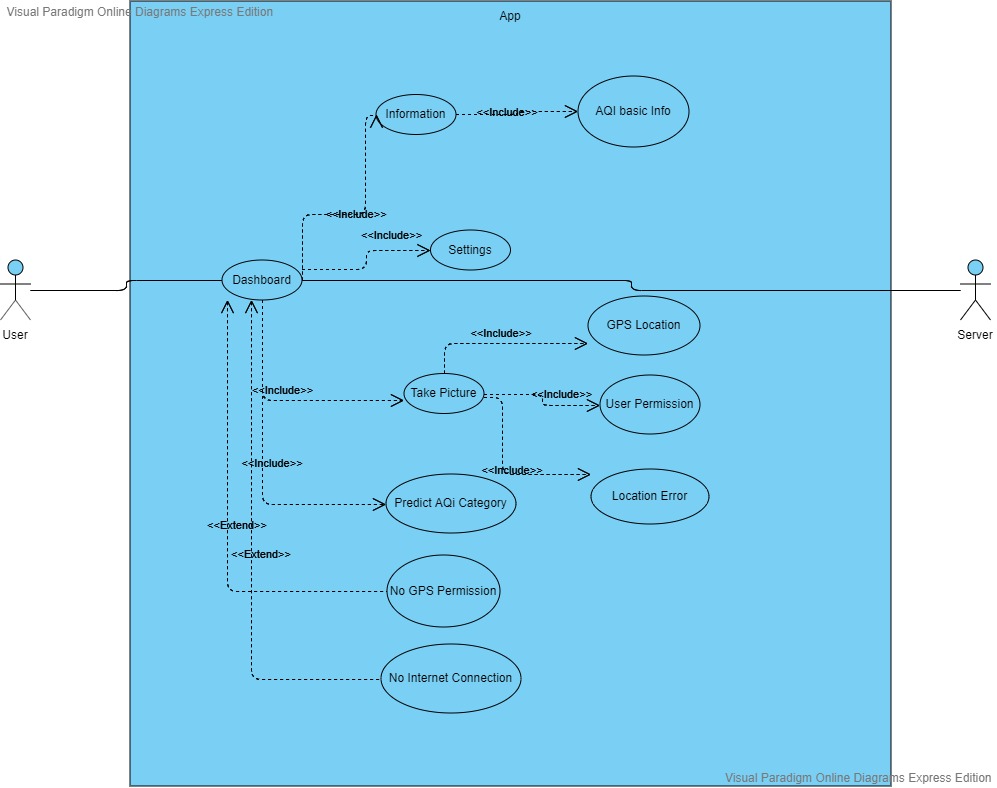


Figure 2‑3 – Use case: App Side

Figure 2-4 shows that the actors in the use case are the user and the server. The user is the main actor here who performs the tasks while the server is connected to the backend. The user takes a picture, the picture goes through the model and then the Air Quality level is predicted. The Air Quality level is then shown to the user on the screen. Then the data is uploaded on the server if there’s an internet connection otherwise it will be done later.

### Server side description

Table 12-Use Case-Server Side

|  |
| --- |
| **Use case ID:** 02 |
| **Use case name:** AQI Predictor Server |
| **Actors:** Admin, App, User, Database |
| **Use Case Summary:** Admin will have access to modify and add into the dataset, the model and the application at any time to make any changes or increase efficiency. |
| **Pre-condition:** The admin must have access to the internet/server in order to perform the tasks. |
| **Course of events:** Admin opens the console, goes into the server and increases the images in the dataset. Then the admin trains the model through the increased images which results in increased efficiency. The admin can also update the application through this. |
| **Post condition:** None |
| **Use Case Cross References:** |
| **Includes:** Connection, training, server, updation, result verdict. |
| **Extends:** Location error, internet error |

Table 12 shows us that the use case with ID 02 titled “AQI Predictor server” explains how everything works at the server end of the application. The dataset can be increased, the model can be retrained and redeployed and the app can be updated through the server side.

### Server side design

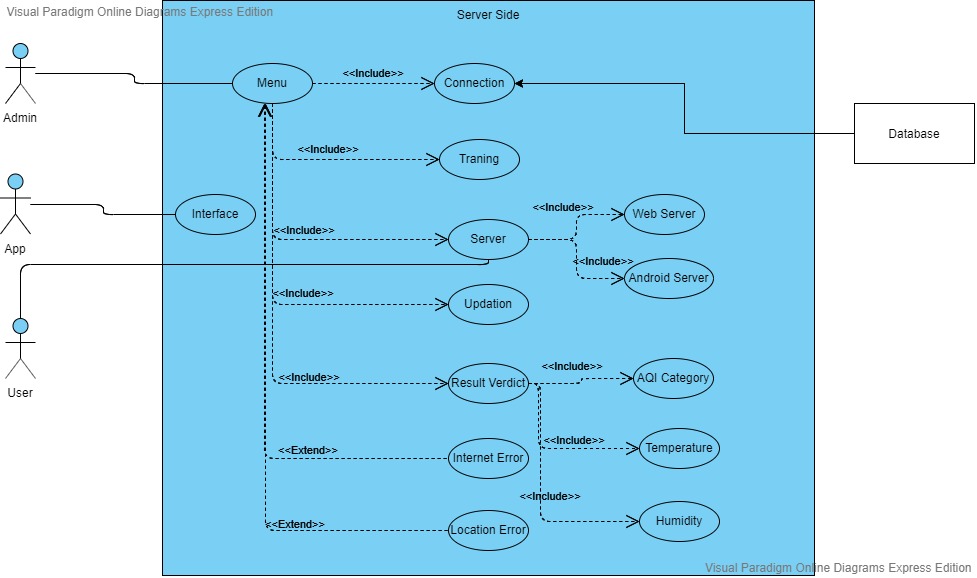


Figure 2‑4 Use case: Server side

Figure 2-5 shows us the use case of Server Side. There are three actors in this use case:

1. Admin
2. App
3. User

This figure explains us that the main actor is the admin. It explains to us the working and how the admin can work through the server side. There’s a menu through which the admin can manage things such as managing the dataset, increasing the dataset, retraining the model and updating the app. All of this is done in connection with the database.

## Software development life cycle model

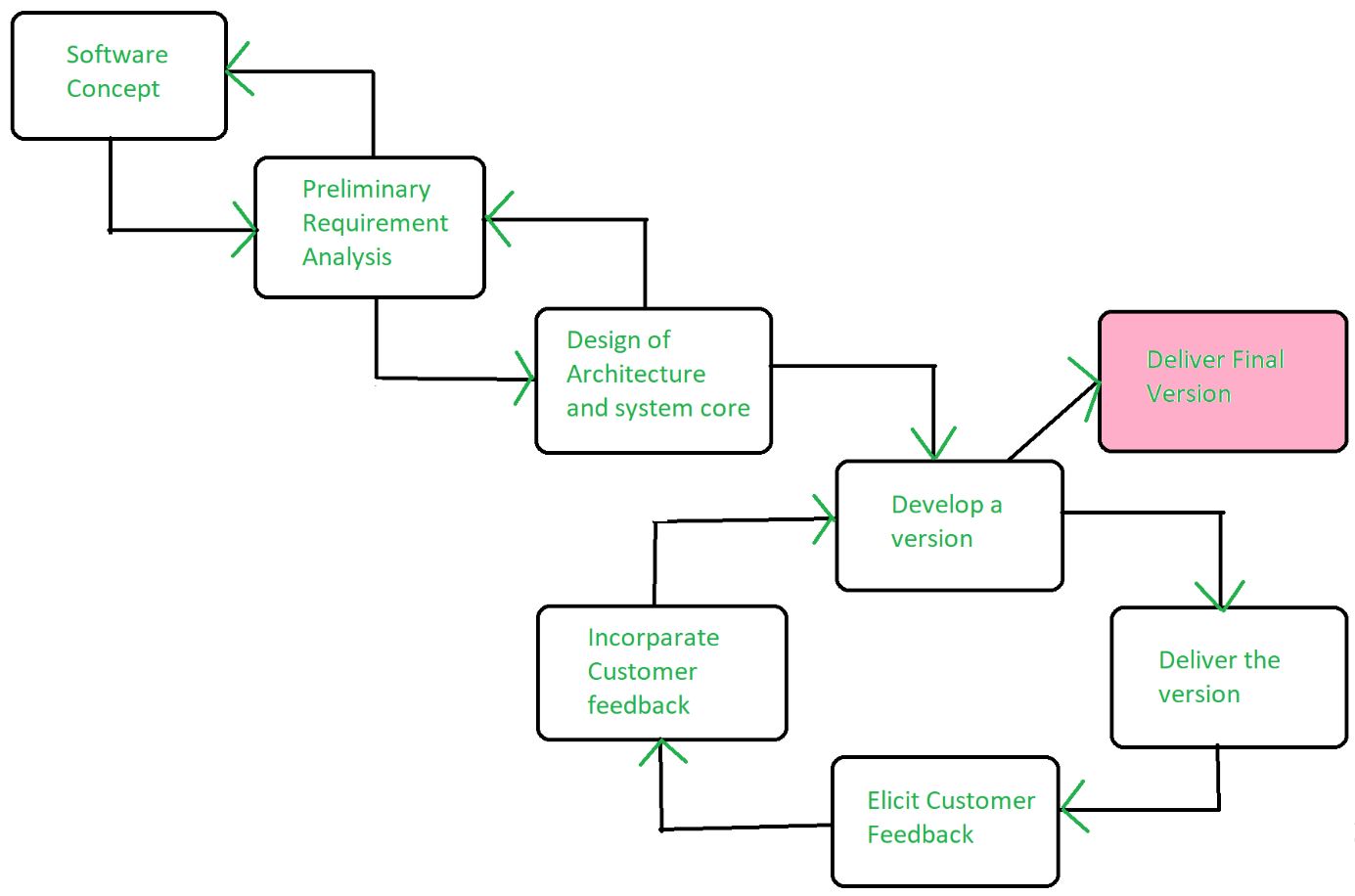


Figure 2‑5 – SDLC Model

Figure 2-6 shows the Software Development Life Cycle Model we have chosen for our project which is the Evolutionary model. The reason behind choosing this model is that we need to redefine our features and functions as we keep developing. Our project is based on modules which are completed in sequence and then integrated into the whole project. The evolutionary model combines iterative and incremental model. We will divide our project into smaller, incremental developments. Then those small developments will be tested and checked if there are any issues. Then we will develop on top of these small developments which will in the end give us our complete projects with updated and latest features and functions as well as efficiency while addressing our issues along the way. We will keep updating (changing and adding new features) our project as we keep gaining more knowledge on how this project could be made better.

**CHAPTER #3**

**SYSTEM DESIGN**

# System Design

Chapter 3 shows all the designs that were created of this project:

## Work breakdown structure

The work breakdown structure is as follows:

Figure 3‑1 – Work Break Down Structure Diagram

Figure 3-1 shows us the work breakdown structure of the whole project starting from research till deployment. It explains all the working of the whole project.

## Sequence Diagram

Sequence diagrams explains the whole working of the system.

### Taking picture, sending to application and getting Air Quality level results:

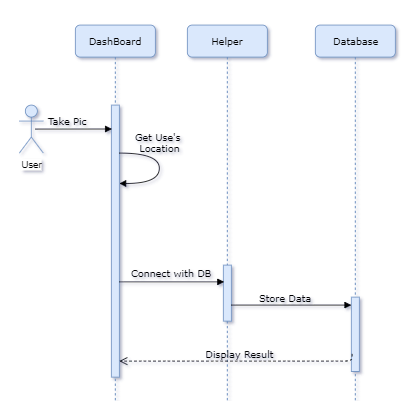


Figure 3‑2 – Sequence diagram – Taking Picture

Figure 3-2 explains the workflow of the entire sequence of how a user takes a picture through the camera of his/her phone, then the picture is sent with the user’s location to the model in the app, then the results are calculated and shown back to the user. After all that, if the user’s device is connected with the internet, it connects with the database at the server end and stores data there.

### Training Sequence Diagram

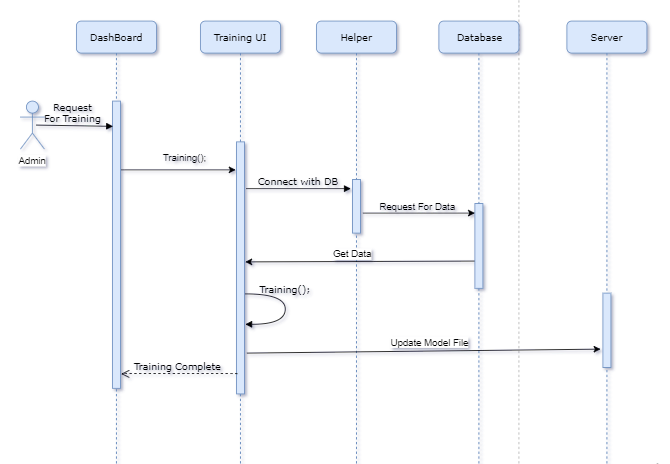


Figure 3‑3 – Sequence Diagram – Training

Figure3- 3 shows us the training sequence of the model. Only the admin can initiate training or retraining of the model embedded in the app. When the admin initiates the training sequence, the model is connected with the database, the data is picked up and fed to the model and then the training is performed. After the training is done, the model file is updated and embedded in the application and the training is complete.

### Web Server Sequence Diagram

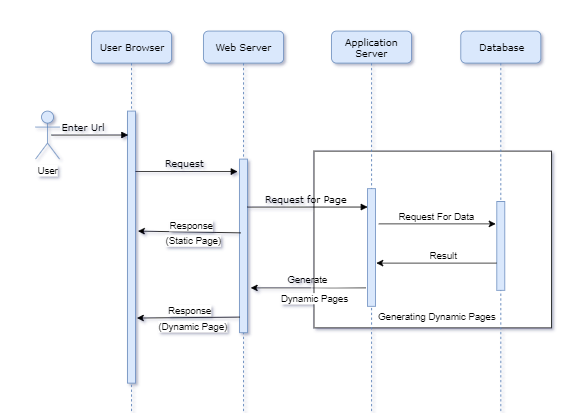


Figure 3‑4 – Sequence Diagram - Web Server

Figure 3-4 shows us the Web Server Sequence Diagram which explains the workflow of the web side of the project. When the user opens the website through the designated URL, the browser requests for access for the page and the data that is available on the database. Upon this, the results are generated from the database and it generates dynamic web pages that are displayed to the user in their browsers.

## High Fidelity Diagrams of the System

The high fidelity diagrams of the whole project are as follows:

### Application:

#### The main page of the application:

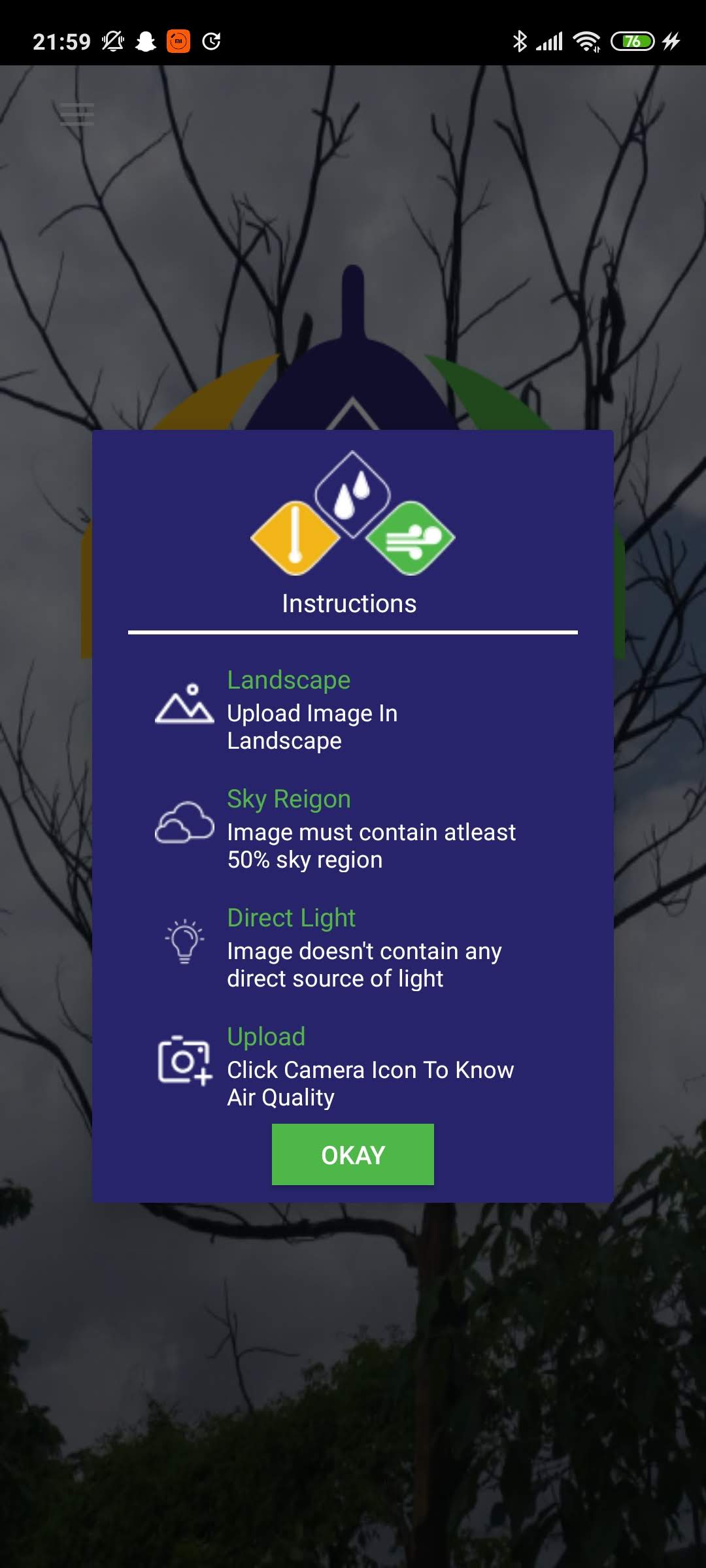


Figure 3‑5 - Main/Home Page of the application

Figure 3-5 shows us home page is pretty straightforward and simple. It does everything it is supposed to do. There are four main options:

1. Camera: to take photos to predict Air Quality level
2. Info: Information about how the whole system works
3. Help: Instructions and FAQs
4. Settings: User can change the temperature format, etc

#### Results Page



Figure 3‑6 - Results page of the application

Figure 3-6 shows the results page of the application. When the results are generated and shown the user in terms of Air Quality level along with some additional information such as humidity level, maximum and minimum temperature which can be seen in the interface. The sun icon will change according to the sky conditions.

### Website

#### Home Page of website

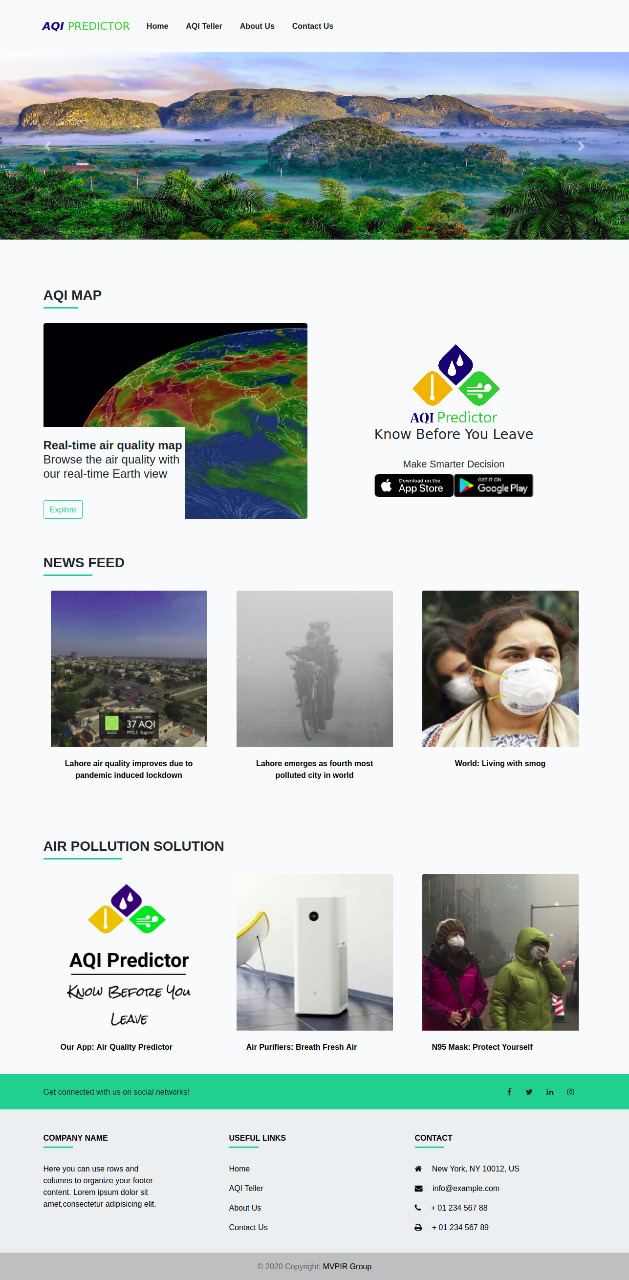


Figure 3‑7- Website Home Page

Figures 3-7 show the home page of the initial website designed for the project. The home page houses the header, the footer and the following features:

1. AQI Map which shows the AQI marked of the cities on the map using Google Maps API
2. A news feed that houses relevant news stories
3. The solutions to Air Pollution

#### AQI Teller Page

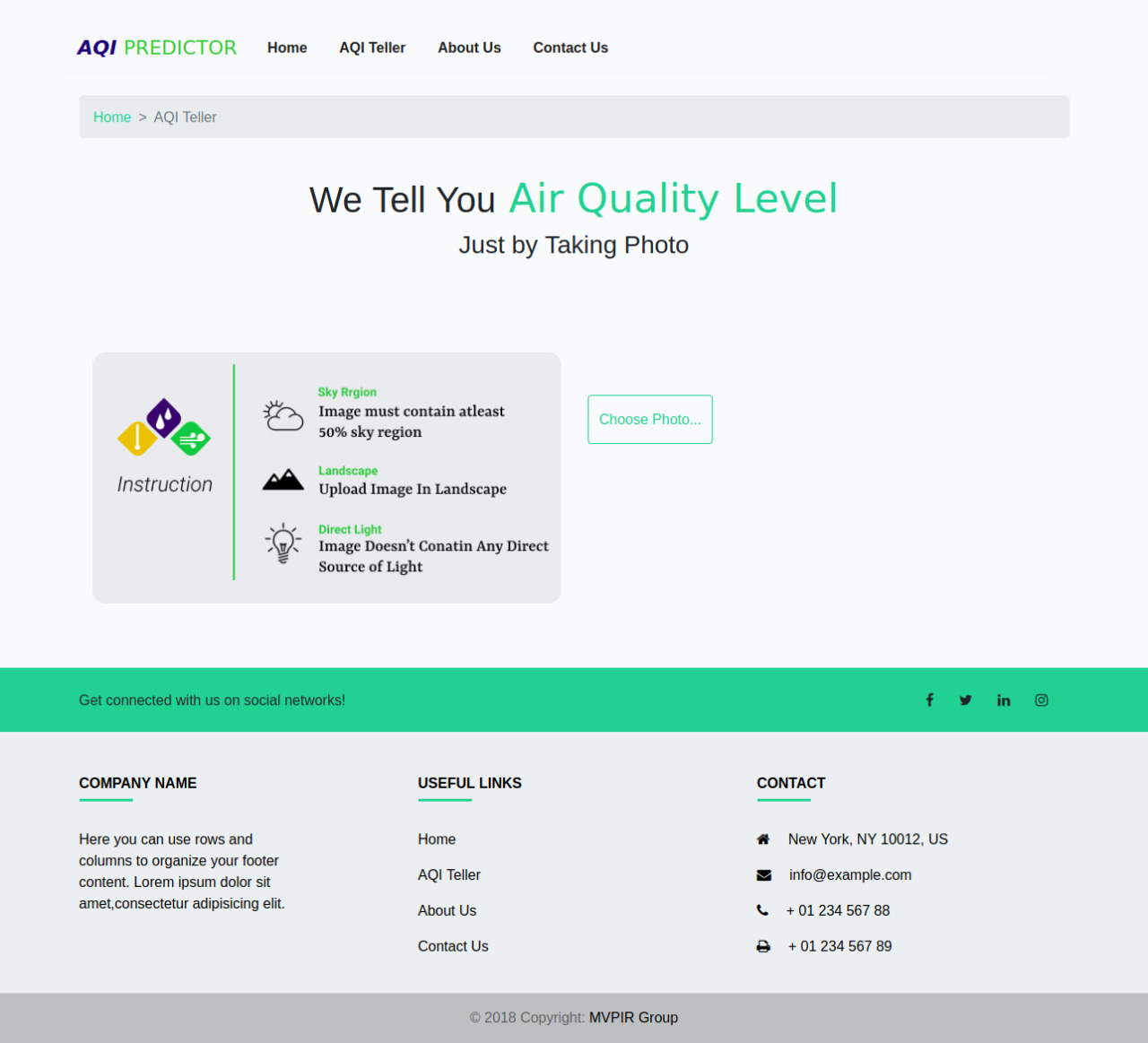


Figure 3‑8- AQI Teller Page of Website

Figure 3-8 shows the AQI Teller Page of the website. The user can select the choose file option and input a picture to the website which will use the trained model to calculate the Air Quality level of the image. This will be one of the most important features of the website.

#### Results Page

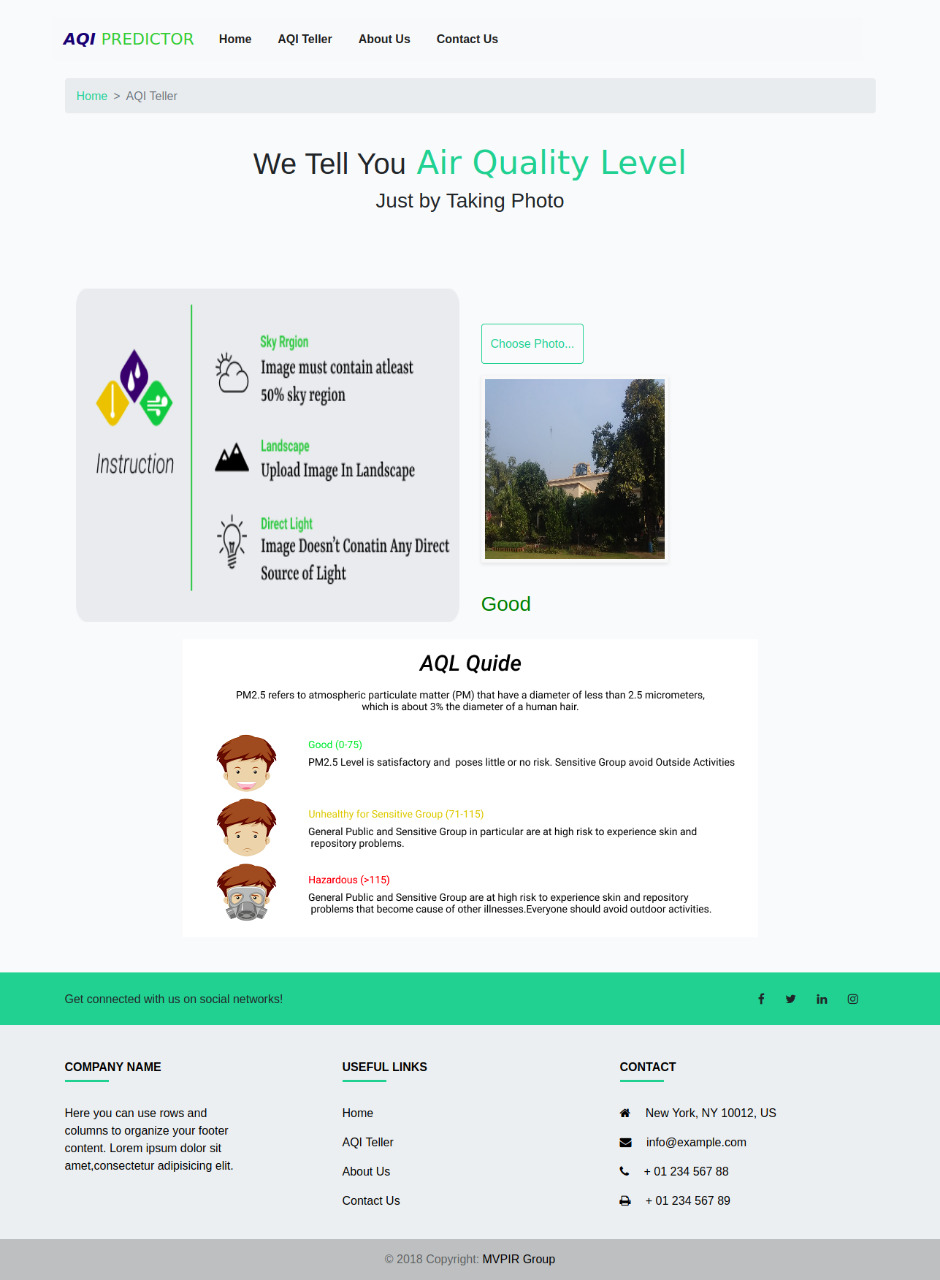


Figure 3-9 – Results Page

Figure 3-9 shows us the results page that is displayed after the air quality level of a photo is predicted.

#### About us

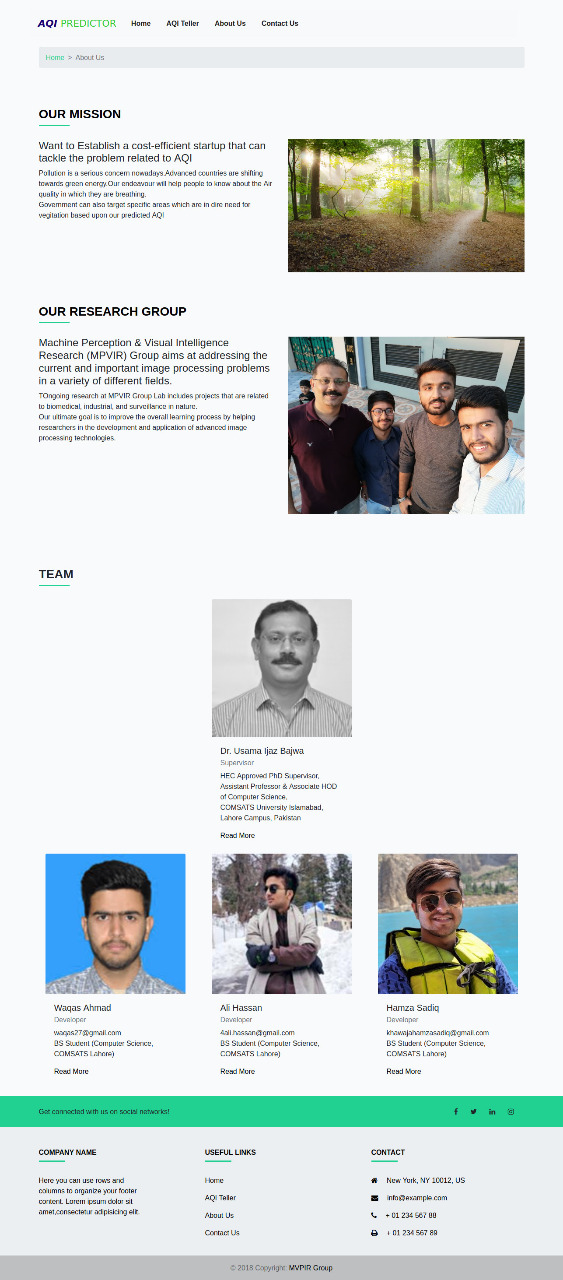


Figure 3‑10 - About Us page of Website

Figures 3-10 show the about us page of the website. It houses all the information about what this project is, who are the creators and who is the supervisor in this project.

#### Contact us page

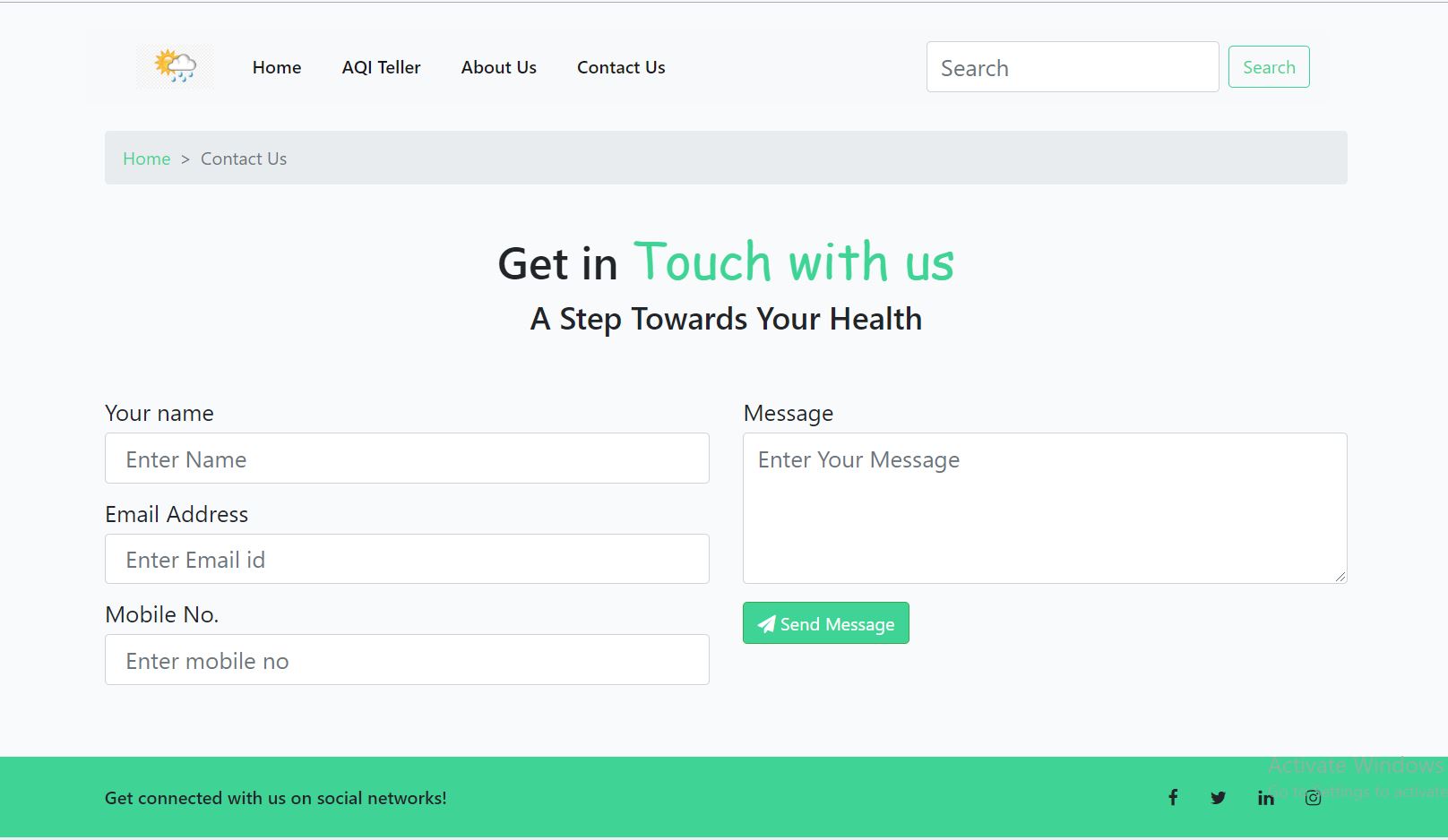


Figure 3‑11 - Contact Us Page of the website

Figure 3-11 shows the contact us page of the website. It’s pretty straightforward. If anybody needs to get in touch or needs some help regarding relevant issues, they can contact us through this simple form.

## Database diagram

|  |
| --- |
| **AQI Predictor** |
| **ID**  ***PK*** |
| **Latitude** |
| **Longitude** |
| **City** |
| **Image URL** |

Figure 3‑12 – Database Diagram

Figure 3-12 shows the database diagram of the project. The database that is going to be used in this project will have the following attributes and constraints. There will be location data of the image and the image’s link in the form of URL.

## Network Diagram (Gantt chart)

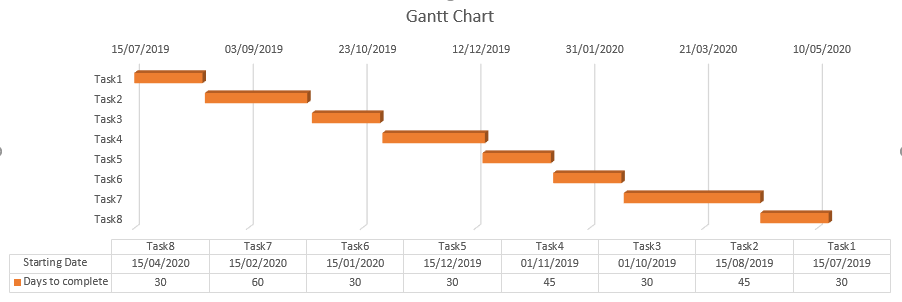


Figure 3‑13 – Gantt Chart

Table 13-Gantt Chart

|  |  |  |  |
| --- | --- | --- | --- |
| Sr No. | Task | Starting Date | Dates to Complete |
| 1 | Research existing solutions,  Requirement gathering, Prioritizing the requirements | 15-July-19 | 30 |
| 2 | Documentation (Use cases, Sequence diagrams) | 15-August-19 | 45 |
| 3 | Gathering the data to train the model, Designing the Front-end | 1-October-19 | 30 |
| 4 | Development of interfaces | 1-November-19 | 45 |
| 5 | Training of the model | 15-December-19 | 30 |
| 6 | Integrating the front-end and backend | 15-January-20 | 30 |
| 7 | Formal testing of application and the website | 15-Feburary-20 | 60 |
| 8 | Deployment | 15-April-20 | 30 |

Figure 3-13 and table 13 shows the Gantt chart of the project which tells the progress of the whole project with respect to time and explains what part of the project will be completed by when.

**CHAPTER #4**

**TESTING**

# Testing

### Air Quality Predictor (Android application)

Table 14 – How the user can view the application

|  |  |
| --- | --- |
| **Test Case ID:** TC-01 | |
| **Application Name:** | AQI Predictor |
| **Use Case(s):** | AQI Predictor Android application |
| **Input summary:** | The user will open the application. The user will tap on the image icon to select an image from either gallery or live through camera. |
| **Output summary:**  Success:  The application will take user to the desired page. | |
| **Pre-conditions:**  User must have turned their location on.  User must have an android smartphone with a working camera. | |
| **Post-conditions:**  Air quality level prediction process will start. | |

Table 14 describes how the user views the application upon launching it. The users can choose any option from two available options i.e. open image from gallery or capture from camera.

Table 15 – How the AQ Level is predicted

|  |  |
| --- | --- |
| **Test Case ID:** TC-02 | |
| **Application Name:** | AQI Predictor |
| **Use Case(s):** | AQI Predictor Android application |
| **Input summary:** | After user has provided the picture through either gallery or camera, the application will process the picture and provide Air Quality level and resultant page. |
| **Output summary:**  Success:  Air Quality level has been predicted. | |
| **Pre-conditions:**  User must have turned their location on.  User must have an android smartphone with a working camera. | |
| **Post-conditions:**  Air quality level will be predicted and shown on the screen alongside other information. | |

Table 15 describes how our android application will process the picture and predict Air Quality level of the place in the picture using either the image from gallery or through camera.

### Air Quality Predictor (Website)

Table 16 – How the user views the website

|  |  |
| --- | --- |
| **Test Case ID:** TC-03 | |
| **Application Name:** | AQI Predictor |
| **Use Case(s):** | AQI Level Prediction using the website |
| **Input summary:** | On our website, the user can tap on “AQI Teller” page and open it. Then the user clicks on “Choose photo” to upload a photo. The website starts processing the image. |
| **Output summary:**  Success:  The website starts processing. | |
| **Pre-conditions:**  User must have a working internet connection.  User must have a device on which the website can be opened in a supported browser. | |
| **Post-conditions:**  Air quality level prediction process has started. | |

Table 16 describes how the user can view our website to choose a photo of which AQ level they want to check. It can be any photo with a good amount of light and sky in it.

Table 17 – How the AQ Level is predicted on website

|  |  |
| --- | --- |
| **Test Case ID:** TC-04 | |
| **Application Name:** | AQI Predictor |
| **Use Case(s):** | AQ Level Prediction using the website |
| **Input summary:** | After user has provided the picture through the option on website, the application will process the picture and provide Air Quality level and resultant page. |
| **Output summary:**  Success:  Air Quality level has been predicted. | |
| **Pre-conditions:**  User must have a working internet connection.  User must have a device with a supported internet browser. | |
| **Post-conditions:**  Air quality level will be predicted and shown on the screen alongside other information. | |

Table 17 describes how air quality level is predicted on our website. The user chooses the “Choose picture” option on the page of “AQI Teller” and inputs a photo which the website processes and predicts Air Quality level.

### Showing notifications to the user

#### Showing results to user on Android application

Table 18 – Showing user the results

|  |  |
| --- | --- |
| **Test Case ID:** TC-05 | |
| **Application Name:** | AQI Predictor |
| **Use Case(s):** | Showing results to user on Android application |
| **Input summary:** | After user has done his part, the application processes the photo and shows the user the results. |
| **Output summary:**  Success:  Air Quality level has been predicted and shown to the user on the android screen in a neat, understandable manner. | |
| **Pre-conditions:**  AQ Level has been predicted. | |
| **Post-conditions:**  The user can view the results alongside additional information and precautions. | |

Table 18 shows the results of the Air Quality level prediction performed by the application. It shows results alongside additional information such as precautions.

#### Showing results to the user on website

Table 19 – Showing results to User on website

|  |  |
| --- | --- |
| **Test Case ID:** TC-06 | |
| **Application Name:** | AQI Predictor |
| **Use Case(s):** | Showing results to user on AQI website |
| **Input summary:** | After user has done his part, the application processes the photo and shows the user the results. |
| **Output summary:**  Success:  Air Quality level has been predicted and shown to the user on the web page in a neat, understandable manner. | |
| **Pre-conditions:**  AQ Level has been predicted. | |
| **Post-conditions:**  The user can view the results alongside additional information and precautions. | |

Table 19 describes how the user sees the results from the website after it processes the picture.

### Admin/Server End

#### Saved images on drive

Table 20 - Admin can view saved images on Drive

|  |  |
| --- | --- |
| **Test Case ID:** TC-07 | |
| **Application Name:** | AQI Predictor |
| **Use Case(s):** | Admin/Server End |
| **Input summary:** | The admin can view and control saved (by application) images on Google Drive. |
| **Output summary:**  Success:  Admin can view images saved on Google Drive. | |
| **Pre-conditions:**  Saved images on Drive. | |
| **Post-conditions:**  AQ Level values are saved in Database. | |

Table 20 describes how the admin can see the images saved on drive.

#### Saving values in Database

Table 21 - Admin can see saved values in Database

|  |  |
| --- | --- |
| **Test Case ID:** TC-08 | |
| **Application Name:** | AQI Predictor |
| **Use Case(s):** | Admin/Server End |
| **Input summary:** | The admin can view and control saved values in database. |
| **Output summary:**  Success:  Admin can view values saved in database. | |
| **Pre-conditions:**  Saved values in database. | |
| **Post-conditions:**  None | |

Table 21 describes how the admin can see AQ level values saved in Database.

**CHAPTER #5**

**CONCLUSION**

# Conclusion

## Project summary

In conclusion, our project, AQI Predictor, consists of an app and a website. The functionality that predicts Air Quality level and the model embedment will be shared across the app and the website. The app and website will use a machine learning model that will help in accurately predicting the Air Quality level of the place in the picture. Both platforms will be synchronized in terms of data through a mutual database that is connected with both website and android application. All the results and data will be stored in the database.

The data collected through the app will populate our dataset which can later be used for further purposes. One of these purposes will be to show a map on the website that will have the AQI of the places that were calculated by the users in the app. This will help in checking Air Quality level through the website as well as the app, whichever the users prefer.

It can further tell people about more details including if the air is harmful/good and what the necessary precautions should be depending upon the AQI. The AQI can further help authorities and other relevant bodies in deciding and gathering data which is usually collected through expensive sensors.

## Future work

As of now, our project is going to be of intermediate professional level which can predict AQI with good accuracy that will be sufficient for our usage and will only use photos with natural light (day time) to predict results correctly. Currently, our system doesn’t calculate exact AQI value. It calculates the level of the Air Quality based.

In the future, it can be further enhanced by providing the model with a much larger dataset that includes various countries to train it further so it can work for those places too. It can also be updated if there’s a way found such that the model can predict the AQI value. The model can also be enhanced in a way that the accuracy can be increased and can work at night time as well. This will increase the efficiency and effectiveness of this project in the future. Furthermore, as our website grows in functionality and data that will be constantly collected, it will reach a much wider audience that can help us grow into a much larger and effective society helping us open the doors for many more possibilities.

**CHAPTER #5**

**REFERENCES**

# References

A. Chakma, B. V. (2017). Image-based air quality analysis using deep convolutional neural network. *2017 IEEE International Conference on Image Processing (ICIP)*, 3949-3952.

AirCognizer. (2019). *Air Cognizer*. Retrieved from PlayStore: https://play.google.com/store/apps/details?id=com.bvpiee.com.airmirror&hl=en\_US

AirVisual. (2019). *Air Quality || Air Visual*. Retrieved from PlayStore: https://play.google.com/store/apps/details?id=com.airvisual&hl=en

Li, R., Liu, X., & Li, X. (2015, May 12). *Estimation of the PM2.5 Pollution Levels in Beijing Based on Nighttime Light Data from the Defense Meteorological Satellite Program-Operational Linescan System*. Retrieved from MDPI: https://www.mdpi.com/2073-4433/6/5/607/htm

Liu, C., Tsow, F., Zou, Y., & Tao, N. (2016, February 1). *Particle Pollution Estimation Based on Image Analysis*. Retrieved from Plos One: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0145955

Vahdatpour, M. S., Sajedi, H., & Ramezani, F. (2018). Air pollution forecasting from sky images with shallow. *Earth Science Informatics, 11*(3), 413-422.

WANG, Z., ZHENG, W., SONG, C., ZHANG, Z., LIAN, J., YUE, S., et al. (2019). Air Quality Measurement Based on Double-Channel Convolutional Neural Network Ensemble Learning. *IEEE Access, 7*, 145067-145081.