# Evergreen Surveillance

**Senior Project**

Primary Advisor: **Mr. Salman Chaudhry**
Secondary Advisor: **Mr. Akheem Yousaf**

Presented by:

| | |
|---|---|
| 22-10293 | Bareera Malik |
| 22-11345 | Bilal Saddique |
| 22-11108 | Mahad Rashid Khurshid |

Department of Computer Science

**Forman Christian College (A Chartered University)**

# Evergreen Surveillance

**By**

**Bareera Malik**
**Bilal Saddique**
**Mahad Rashid Khurshid**

Project submitted to :

**Mr. Salman Chaudhry**

Department of Computer Science,

Forman Christian College (A Chartered University),

Lahore, Pakistan.

In partial fulfillment of the requirements for the degree of

**BACHELOR OF SCIENCE**
**IN**
**COMPUTER SCIENCE (Honors)**

______________________________ Primary Project Advisor

______________________________ Secondary Project Advisor

______________________________ Senior Project Management Committee Representative

# Abstract

Trees are crucial for countless human services and it won't be wrong to label them as the life support systems for the planet. It is an undoubted fact that urban areas experience higher levels of air pollution due to the "human impact" which involves releasing of harmful gases and urban sprawling. Many pollutants, such as SO2, NO2, CO2, NO, CO, NOx, PM2.5, and PM10 can be found in the atmosphere. Trees are much crucial assets than we imagine them to be against the harmful air quality which propose threatening diseases for us humans. As essential as they are for us and the environment we breathe in, it is equally important for us to be aware of the trees we live with or the urban forest that surrounds us. From individually standing street trees, greenspaces, and groves to suburban forest all put together make up the green infrastructure of our community. In order to ensure an excellently breathable **Air Quality Index**, tracking **tree count** and regularly monitoring them is the main step to achieving the goal. After going through a number of researches, facts and figures, we found that presence of trees is directly proportional to the improvement in AQI given the kind of area targeted is a constant factor. To serve the purpose, this paper "**Evergreen Surveillance"** presents a "**tree detection & monitoring system to predict AQI**" using google maps API. The approach of our system is such that it works by comparing satellite images of a targeted area where trees or any green area is detected and the green covered area (sqkm) is constantly recorded in the data log. The data log is regularly being updated as a newer image comes in and is compared with the previous image in account of the green covered area. And thus by comparing the value of green covered area over a certain period of time, we can derive where plantation needs to be done in order to improve the AQI eventually. The next part of our project involves predicting the Air Quality Index for which we have used the prediction model of python essentially by collecting and analyzing past data. We have trained a model that will detect patterns in order to predict the final outcome. The results of our project will highlight the importance of trees/greenery in the improvement of Air Quality Index and is a specific solution for all the organizations and institutes who need to be aware of the vegetation, trees or the green area condition in order to improve and maintain the air quality.

# Acknowledgement

The completion of this project would not have been possible without the expertise of Sir Salman Chaudhry, our respected project advisor. We would like to express our deepest appreciation and give our warmest thanks to him. His guidance, persistent help and overall insights in this field has carried us through all the stages of our final year project and has made this an inspiring experience for us. We especially acknowledge him for his involvement with originality which has nourished our intellectual maturity that will help us for a long time to come.

In addition, we'd also like to thank Sir Akheem Yousaf, whose invaluable feedback and encouragement greatly influenced the course of our project. Both our advisors, Sir Salman Chaudhry and Sir Akheem Yousaf, have been incredible mentors and we are grateful to them. Also, deep sense of gratitude for the Faculty of Computer Science at Forman Christian College (A Chartered University) for providing us with the resources and knowledge to pursue graduate study in this department.

Most importantly, we owe this project to our parents. We are forever indebted to them for their love and emotional as well as financial support. Without their constant presence and support, this day would not have been possible. Last but the most significant, we would like to praise and thank Allah, the almighty, who has granted countless blessing, knowledge, and opportunities to us, so that we have been finally able to witness this accomplishment.

# List of Figures

# List of Tables

# Table of contents

# Revision History

| Name | Date | Reason For Changes | Version |
|---|---|---|---|
| | | | |
| | | | |

# Chapter 1. Introduction

## 1.1 Introduction

According to an official study, Pakistan has five trees per person while the world average is 422. Other than just producing cleaner air for us to breathe, trees are also responsible for reduction in energy use, improvement in water quality, increased carbon capture and storage, and an improvement in individual and community wellbeing. According to the International Society of Arboriculture, the largest type of existing pollutants are called “particulates” which are less than 2.5 microns in size normally generated by combustion of fossil fuels, construction and complex reactions between sunlight and gaseous pollutants. These particulates have been associated with respiratory and cardiopulmonary diseases, and cancer. The good thing is even though they are not fully absorbed by trees because of their tiny size, leaves are able to collect them on the surface and remove them from the atmosphere.

The motivation behind proposing our project is to spread awareness and monitor our environment because only then can a change be brought about on an individual as well as the state level. Our meticulously designed system is using image processing to analyze satellite images and then predict AQI (Air Quality Index) for future days. It uses techniques such as color thresholding and image segmentation to extract the green covered area from satellite images. The extracted green area is then used as a feature to align with the predicted AQI levels. The predictor model is trained using historical AQI data and meteorological data to have it working. We have used python libraries such as OpenCV, scikit-learn, numpy and pandas. The project can be run on a local machine or on a remote server, and the user can request predicted AQI value of 1th day, 15$^{th}$ day, and 30$^{th}$ day as well. The user can request for AQI predicted value using location and latitude longitude as well. In return, our system will show the map view as well as the satellite view of the requested location with the covered area and AQI value against the two views. It is worth noting that such predictions are usually approximate, and that the accuracy of these predictions may vary depending on other environmental factors and the quality of the data used for training and testing and the complexity of the model. Moreover, to make the system precise, we are taking satellite images that are of different zoom levels for detailed and meticulous results. Improving the urban forest around us provides a baseline to start monitoring patterns and trends as things change with climate change. Our proposed project is intended to act as a kick-start for different institutes and organizations in Pakistan to overcome problems that require tree count to be known as the first thing.

## 1.2 Objectives

Our main objectives for our system are:

- Utilize image processing techniques(OpenCV) and machine learning techniques(TensorFlow) to analyze satellite images and identify the amount of green coverage present in a specific area
- Train a model that can accurately predict future AQI levels for different days (1th, 15th and 30$^{th}$ day in our case)
- Monitor the green coverage and AQI in a specific area and forecast the AQI for future days.
- Provide insights to authorities and organizations that can help in making decisions on environmental policies, urban planning and public health measures.
- Visualizations of the data collected and analyzed, such as satellite images of the area showing green coverage
- Results of the AQI predictions over time.

Overall, this project aims to use image processing and machine learning techniques to gain a better understanding of the relationship between green coverage and air quality index through the results. Moreover, what is even more crucial is the fact that there are many other factors involved in determining the AQI of an area, and so this can be deduced if AQI of a particular area remains the same even after green covered area has increased over time.

## 1.3 Problem Statement

Urban areas need trees to be environmentally sustainable. Urban forestry management can be a challenging task with uncountable trees in our environment. We are all aware that Afforestation is critical to mitigate climate change but Pakistan's rapid deforestation still remains the second highest in Asia. Being a climate-vulnerable country where deadly heat waves are becoming more common, Pakistan's forest cover is among the lowest in the world (5% compared with the global average of 31%). According to environmentalists, our cities are becoming completely concretized and town planners don't keep plantations in mind ignoring the fact that urban plantation is equally important as it provides shade, beauty but most importantly a cooling effect. While it should be the other way around, it is seen that town planners sometimes cut down trees in order to construct buildings. Other than this, while it is true that buildings and infrastructure experience a cooling effect from urban green spaces, old and dead trees are a hazard as they pose safety risks and can spread diseases so cutting down trees is also something that needs to be taken care of.

By highlighting the significance of these problems, it can be easily concluded that the solution to all these problems is only possible if rigorous monitoring is done of urban trees and our project aims to do exactly that.

With the department of climate change and meteorology being the primary body that will need this system, there are other organizations and institutes that need this update in account of the problems listed above. Plantation drive initiators and urban forest projects also require this system so they are aware of the current situation of the depleting green cover in an area and can plan out their projects accordingly.

Laws against cutting down trees can only be implemented if authorities are aware of when trees are cut down and where. Through our project, reports can be generated highlighting the spots wherever any sort of violation occurs.

## 1.4 Scope

We are trying to step up from the manual Tree/ Management System to the autonomous Web related interface. Our system will have two components, the first component includes detection of green covered area with justified accuracy using Google Maps. Second component includes a predictor that will predict the air quality index. Our objective is to propose a fully automated tree detection system that can process thousands of images using publicly available satellite images of Google Maps. We will be applying CNN (Convolutional Neural Networks) and LSTM (Long Short Term Memory Model) for image processing and prediction. Initial steps of our system are that satellite images of our area of interest are downloaded and then green covered are is detected per image. In addition to this, our system will also be predicting the air quality index in the targeted area. For instance, if our system shows the AQI of today, it is trained to predict the AQI for the very next day, after 15 days and then 30 days as well. This prediction will align (with other factors as well) since temperature, oxygen and carbon-monoxide levels, along with some other gasses are also affected by the number of trees in an area.

# Chapter 2. Requirements Analysis

## 2.1 Literature Review

Tree detection in cities has gained attention since the early 2000s. Early methods for single tree delineation in cities were inspired by scale-space theory (initially also developed for forests by Brandtberg and Walter (1998)). A common strategy is to first segment data into homogeneous regions, respectively 3D clusters in point clouds, and then classify regions/clusters into tree or background, possibly followed by a refinement of the boundaries with predefined tree shape priors or active contours. For example, Straub (2003) segments aerial images and height models into consistent regions at multiple scales, then performs refinement with active contours. Recent work in urban environments (Lafarge and Mallet, 2012) creates 3D city models from dense aerial LiDAR point clouds, and reconstructs not only trees but also buildings and the ground surface. After an initial semantic segmentation with a breakline-preserving MRF, 3D templates consisting of a cylindrical trunk and an ellipsoidal crown are fitted to the data points. Similarly, tree trunks have been modeled as cylinders also at smaller scales but higher resolution, using LiDAR point clouds acquired either from UAVs (Jaakkola et al., 2010) or from terrestrial mobile mapping vehicles (Monnier et al., 2012).

## 2.2 User Classes and Characteristics

In a Python project that uses image processing to extract information about vegetation from satellite images and then uses that information, along with air quality data, to predict future AQI values.
There could be several user classes each with different characteristics since our system falls in the category of "environmental projects." The characteristics of these user classes are that they may have different levels of technical knowledge, different goals and different ways of accessing and using the system. The user classes are as follows:

General Public: These users will be interested in accessing the predictions to understand the air quality index in their particular area and how it may be affected by vegetation. They have a general interest in environmental issues and in improving the environmental condition around us to protect themselves from threatening diseases. They may not have a specific background in the field.

Government Officials: These users are interested in using the predictions regarding AQI and data about the green covered area to make decisions about air quality management and policy as well as urban forestry programs. Government officials will also halt urban town planning if they need to if a situation in an area gets alarming. These users are interested in using the predictions to support their decisions.

Environmental Scientist: These users are interested in using the predictions to understand the relationship between vegetation and air quality. These users are also expected to be familiar with AQI data and its implications.

Developers: These users are responsible for the development, maintenance, and testing of the system. They have a strong background in software development and are familiar with image processing, machine learning and web development technologies.

We have considered the user's needs and goals when designing the system, by making it user-friendly and accessible to a wide range of users.

## 2.3 Design and Implementation Constraints

There could be several design and implementation constraints that need to be taken into account:

Dataset Availability: The availability of high-quality satellite images and air quality data may be limited, which could impact the accuracy of the results. The project may need to incorporate data from multiple sources and pre-processing techniques to clean and organize the data.

Computational Technicalities/Resources: The image processing and machine learning tasks required for our system may require significant computational resources, especially if the project needs to process large amounts of data. The project may need to be designed to take advantage of distributed computing systems or cloud-based services to handle the computational load.

Data Privacy and Security: The project may need to handle sensitive data such as personal information of the users and location data, so the project must be designed with data privacy and security in mind. This can include using encryption to protect data, and implementing access controls to restrict access to sensitive data.

User Interface: The project should be designed with a user-friendly interface to make it easy for users to access the predictions and understand the results. This may include incorporating visualizations, interactive maps, and other tools to help users understand the predictions.

Scalability: The project should be designed to be scalable, so that it can handle increasing amounts of data and user demand as the project grows. This could include designing the project to be modular, so that new features can be easily added, and implementing load balancing and other techniques to handle increased traffic.

Testing and validation: The project should be thoroughly tested and validated to ensure that the predictions are accurate and that the system is reliable. This includes testing the image processing and machine learning algorithms, as well as the user interface, to ensure that the system is working as expected.

Hardware constraints: The project may have constraints on the hardware and operating system it can be deployed on, especially if it is intended to run on a specific platform.

## 2.4 Assumptions and Dependencies

- A person using the system should have a minimum understanding of the system and should have minimal qualifications.
- The images should be clear.
- The picture of vegetation should contain more subjects and minimum distractions.
- A crisp sharp image instead of blurry images should be scanned.
- The hardware and software requirements of the system should be up to the mark on which the Application has to run.

## 2.5 Functional Requirements

### Use Cases:

### 2.5.1 User opens the website

| | | | |
|---|---|---|---|
| **Identifier** | UC-1: User opens the website | | |
| **Purpose** | To collect the satellite images for database. | | |
| **Priority** | High | | |
| **Pre-conditions** | Internet connection is required. | | |
| **Post-conditions** | User allows software to take the picture. | | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | **System Response** | |
| **1** | User opens the website | Login | |
| **2** | Open their dashboard | Dashboard opened | |
| **3** | Click on "Take Snapshot" | Website asks the user to access its geolocation | |
| **…** | The user selects the geolocation | Website asks the user to take the snapshot | |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | **System Response** | |
| **1** | | | |
| **2** | | | |
| **3** | | | |
| **…** | | | |

**Table 1: UC-1**

### 2.5.2 Give Access to API

| | | | |
|---|---|---|---|
| **Identifier** | UC-1: Give Access to API | | |
| **Purpose** | The user gives access to the application | | |
| **Priority** | High | | |
| **Pre-conditions** | Internet connection is required. | | |
| **Post-conditions** | The user will give access to the application | | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | User gives access to the application | | System gets the access of the geolocation |
| **2** | | | |
| **3** | | | |
| **…** | | | |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | User denies giving access to the Map API. | | System does not get access to the Map API. |
| **2** | | | System displays information “Map API access is required”. |
| **3** | | | |
| **…** | | | |

**Table 2: UC-2**

### 2.5.3 Give Access to Location

| | | | |
|---|---|---|---|
| **Identifier** | UC-3: Give Access to Location | | |
| **Purpose** | The user gives access to the application to get the geolocation | | |
| **Priority** | High | | |
| **Pre-conditions** | Internet connection is required. | | |
| **Post-conditions** | The geolocation of the user will be known to the application | | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | **System Response** | |
| **1** | User gives access to the location | System gets the access of the location | |
| **2** | | | |
| **3** | | | |
| **…** | | | |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | **System Response** | |
| **1** | User denies giving access to the location | System does not get access to the Map API. | |
| **2** | | System displays information “Location access is required”. | |
| **3** | | | |
| **…** | | | |

**Table 3: UC-3**

### 2.5.4 Take Snapshot

| | | | |
|---|---|---|---|
| **Identifier** | UC-4: Take Snapshot | | |
| **Purpose** | To take snapshot of green covered area | | |
| **Priority** | High | | |
| **Pre-conditions** | • A connection to the internet<br>• Access to user's Map API<br>• Access to user's location | | |
| **Post-conditions** | The user will have to retake snapshot of the trees | | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | **System Response** | |
| **1** | The user takes a snapshot of trees | The system gets the user the option to upload or retake snapshot | |
| **2** | | | |
| **3** | | | |
| **…** | | | |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | **System Response** | |
| **1** | | | |
| **2** | | | |
| **3** | | | |
| **…** | | | |

**Table 4: UC-4**

### 2.5.5 Retake snapshot

<table>
<tr><td>Identifier</td><td colspan="2">UC-4: Retake Snapshot</td></tr>
<tr><td>Purpose</td><td colspan="2">The user retakes the snapshot</td></tr>
<tr><td>Priority</td><td colspan="2">High</td></tr>
<tr><td>Pre-conditions</td><td colspan="2"><ul><li>A connection to the internet</li><li>Access to user’s Map API</li><li>Access to user’s location</li></ul></td></tr>
<tr><td>Post-conditions</td><td colspan="2">The user will have a retaken snapshot of trees</td></tr>
<tr><td colspan="3">Typical Course of Action</td></tr>
<tr><td>S#</td><td>Actor Action</td><td>System Response</td></tr>
<tr><td>1</td><td>The user retakes a snapshot of trees</td><td>The system gives the user the option to upload or retake the snapshot</td></tr>
<tr><td>2</td><td></td><td></td></tr>
<tr><td>3</td><td></td><td></td></tr>
<tr><td>…</td><td></td><td></td></tr>
<tr><td colspan="3">Alternate Course of Action</td></tr>
<tr><td>S#</td><td>Actor Action</td><td>System Response</td></tr>
<tr><td>1</td><td></td><td></td></tr>
<tr><td>2</td><td></td><td></td></tr>
<tr><td>3</td><td></td><td></td></tr>
<tr><td>…</td><td></td><td></td></tr>
</table>

**Table 5: UC-5**

### 2.5.6 Upload the Snapshot

| | | | |
|---|---|---|---|
| **Identifier** | UC-4: Upload the Snapshot | | |
| **Purpose** | The user uploads the snapshot of the green covered area | | |
| **Priority** | High | | |
| **Pre-conditions** | • A connection to the internet<br>• Access to user's location | | |
| **Post-conditions** | The image is uploaded to the server | | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | **System Response** | |
| **1** | The user uploads the image2. | The application tells the user that the image has been successfully uploaded. | |
| **2** | | | |
| **3** | | | |
| **…** | | | |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | **System Response** | |
| **1** | The user uploads the image | The application tells the user that the image has not been successfully uploaded. | |
| **2** | | | |
| **3** | | | |
| **…** | | | |

**Table 6: UC-6**

### 2.5.7 Analyze Image

| | | |
|---|---|---|
| **Identifier** | UC-4: Analyze Image | |
| **Purpose** | The image is analyzed to see whether or not it has green covered area. | |
| **Priority** | High | |
| **Pre-conditions** | The user has uploaded the snapshot | |
| **Post-conditions** | The geolocation of the image is either stored or discarded | |
| **Typical Course of Action** | | |
| **S#** | **Actor Action** | **System Response** |
| **1** | The image is analyzed by the server and green covered area is found. | The geolocation of the user is stored in the database. |
| **2** | | |
| **3** | | |
| **…** | | |
| **Alternate Course of Action** | | |
| **S#** | **Actor Action** | **System Response** |
| **1** | The image is analyzed by the server and green area is not found. | The geolocation of the user is discarded. |
| **2** | | |
| **3** | | |
| **…** | | |

**Table 7: UC-7**

### 2.5.8 Admin Login

| | | |
|---|---|---|
| **Identifier** | UC-8: Admin Login | |
| **Purpose** | To manage the server | |
| **Priority** | High | |
| **Pre-conditions** | Admin should have valid credentials. | |
| **Post-conditions** | Admin will be logged in to the server. | |
| **Typical Course of Action** | | |
| **S#** | **Actor Action** | **System Response** |
| **1** | Admin enters Username/Email | |
| **2** | Admin enters Password | |
| **3** | | Display Main Page |
| **…** | | |
| **Alternate Course of Action** | | |
| **S#** | **Actor Action** | **System Response** |
| **1** | | If admin enters invalid details, show an error message. |
| **2** | | |
| **3** | | |
| **…** | | |

**Table 8: UC-8**

### 2.5.9 Store Geolocation and Image

| | | | |
|---|---|---|---|
| **Identifier** | UC-9: Store geolocation and image | | |
| **Purpose** | To store the geolocation and image of the green covered area | | |
| **Priority** | High | | |
| **Pre-conditions** | The analyzed image | | |
| **Post-conditions** | The geolocation and image of the green covered area is stored on the server. | | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | **System Response** | |
| **1** | The analyzed image contains green covered area | The geolocation is stored on the server database | |
| **2** | | The image is stored on the server database | |
| **3** | | | |
| **…** | | | |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | **System Response** | |
| **1** | The analyzed image does not contain any green covered area. | The geolocation is not stored on the server. | |
| **2** | | The image is not stored on the server database. | |
| **3** | | | |
| **…** | | | |

**Table 9: UC-9**

### 2.5.10 Accessing the AQI database

| | | |
|---|---|---|
| **Identifier** | UC-10: Accessing the AQI database | |
| **Purpose** | The application accesses the AQI database | |
| **Priority** | High | |
| **Pre-conditions** | | |
| **Post-conditions** | Access to the AQI database | |
| **Typical Course of Action** | | |
| **S#** | **Actor Action** | **System Response** |
| **1** | Admin will ask the server to access the AQI database | Access to AQI database will be granted |
| **2** | | |
| **3** | | |
| **…** | | |
| **Alternate Course of Action** | | |
| **S#** | **Actor Action** | **System Response** |
| **1** | | |
| **2** | | |
| **3** | | |
| **…** | | |

**Table 10: UC-10**

### 2.5.11 Access the predicted AQI value

| | | |
|---|---|---|
| **Identifier** | UC-11: Access predicted AQI value | |
| **Purpose** | The application generates the predicted AQI value | |
| **Priority** | High | |
| **Pre-conditions** | AQI database is granted | |
| **Post-conditions** | | |
| **Typical Course of Action** | | |
| **S#** | **Actor Action** | **System Response** |
| **1** | Admin will ask to share the predicted AQI value. | The server will send the AQI value. |
| **2** | | |
| **3** | | |
| **…** | | |
| **Alternate Course of Action** | | |
| **S#** | **Actor Action** | **System Response** |
| **1** | | |
| **2** | | |
| **3** | | |
| **…** | | |

**Table 11: UC-11**

### 2.5.12 Generating graphs

| | | | |
|---|---|---|---|
| **Identifier** | UC-12: Generating graphs | | |
| **Purpose** | To visualize the AQI value over time | | |
| **Priority** | High | | |
| **Pre-conditions** | We have the results for comparison | | |
| **Post-conditions** | The graphs are shown on the website | | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | **System Response** | |
| **1** | The user will be able to see the results and derive conclusions and suggestions. | The website will show the result graphs on the webpage. | |
| **2** | | | |
| **3** | | | |
| **…** | | | |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | **System Response** | |
| **1** | | | |
| **2** | | | |
| **3** | | | |
| **…** | | | |

**Table 12: UC-12**

## 2.6 Use Case Diagram

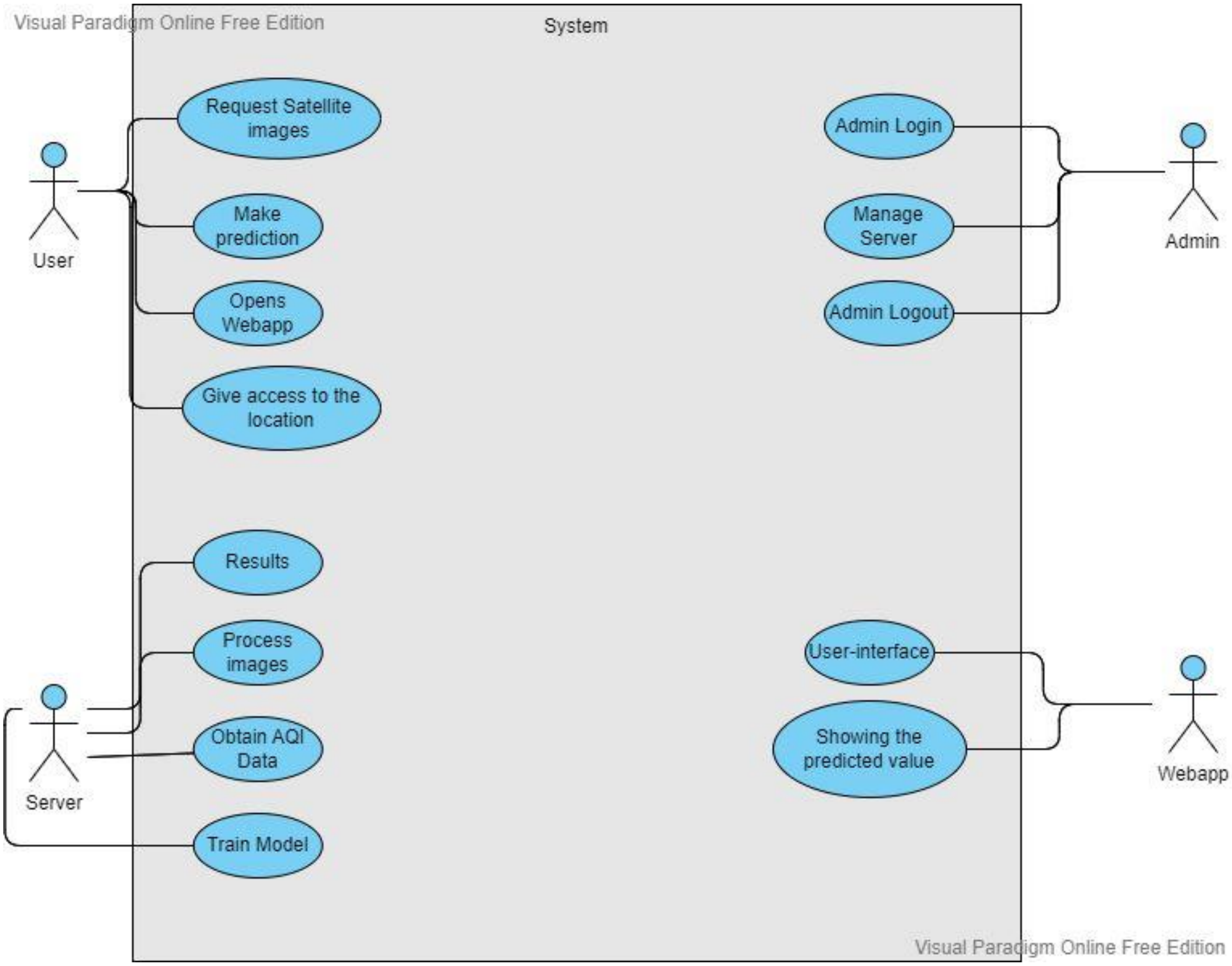


**Figure 1: Use Case Diagram**

## 2.7 Nonfunctional Requirements

### 2.7.1 Performance Requirements

The image processing tasks required for the project must be performed quickly, as the user may need to process a large number of images in a short period of time.

The predictions made by the system must be as accurate as possible as different user classes will be relying on them to make major decisions about air quality management and policy.

The system should be able to handle huge amounts of data under user demand as the project grows. This means the system should be able to handle increased traffic.

The system should be able to intelligently use the memory and storage resources, as the user will need to process and run a large number data in the form of images and store the predictions and the data.

The system should be responsive and available to the user requests most of the time, as the user may be relying on the predictions in real-time.

### 2.7.2 Safety Requirements

The system should ensure that the data used for image processing and machine learning is accurate and reliable. This might require to implement data validation and error checking to ensure error free data.

The system should be able to protect the privacy and confidentiality of the user's data, such as personal information and location data by implementing encryption.

The system should ensure that the data and predictions are backed up regularly, and that the system can recover from any failures. This may include installing a disaster recovery procedures.

The project should be thoroughly tested and validated to ensure that the predictions are accurate and that the system is reliable. This should include testing the image processing and machine learning algorithms, as well as the user interface to make sure that the system is working as expected.

### 2.7.3 Security Requirements

The project should make sure that the system is secure and protects the user's sensitive data from unauthorized access and breaches. This may include implementing firewalls and monitoring the system for potential security threats.

The project should ensure that the system complies with any relevant regulations and standards. This may include implementing auditing and other mechanisms to track user access and system activity and make sure that the system meets the industry standards for data security and privacy.

Machine learning models used in the system should also be safe and should not produce any unintended and harmful results. For this, testing of the model with diverse data may be required to monitor the model for any unexpected behavior.

### 2.7.4 Additional Software Quality Attributes

The application is reusable as it can be reused to fit a different role.

The application is reliable.

The application can be used at any time of the day.

No high end / expensive hardware or third party app is being used.

## 2.8 Other Requirements

The system should have low latency as the user may need the predictions in real-time. This might require to optimize the system architecture and implementing caching mechanisms to reduce latency.

# Chapter 3. System Design

## 3.1 Application and Data Architecture

### 3.1.1 Component Diagram

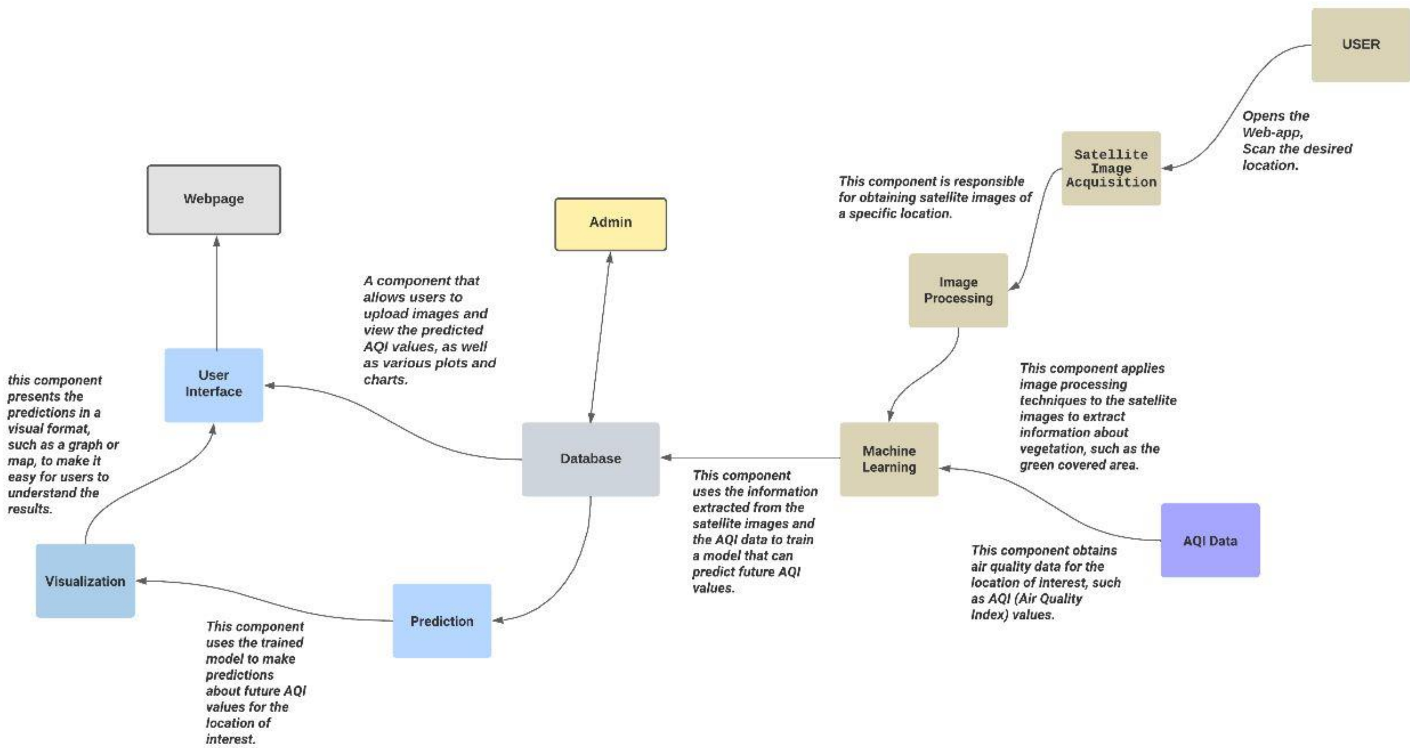


**Figure 2: Component Diagram**

### 3.1.2 Entity-Relationship Model

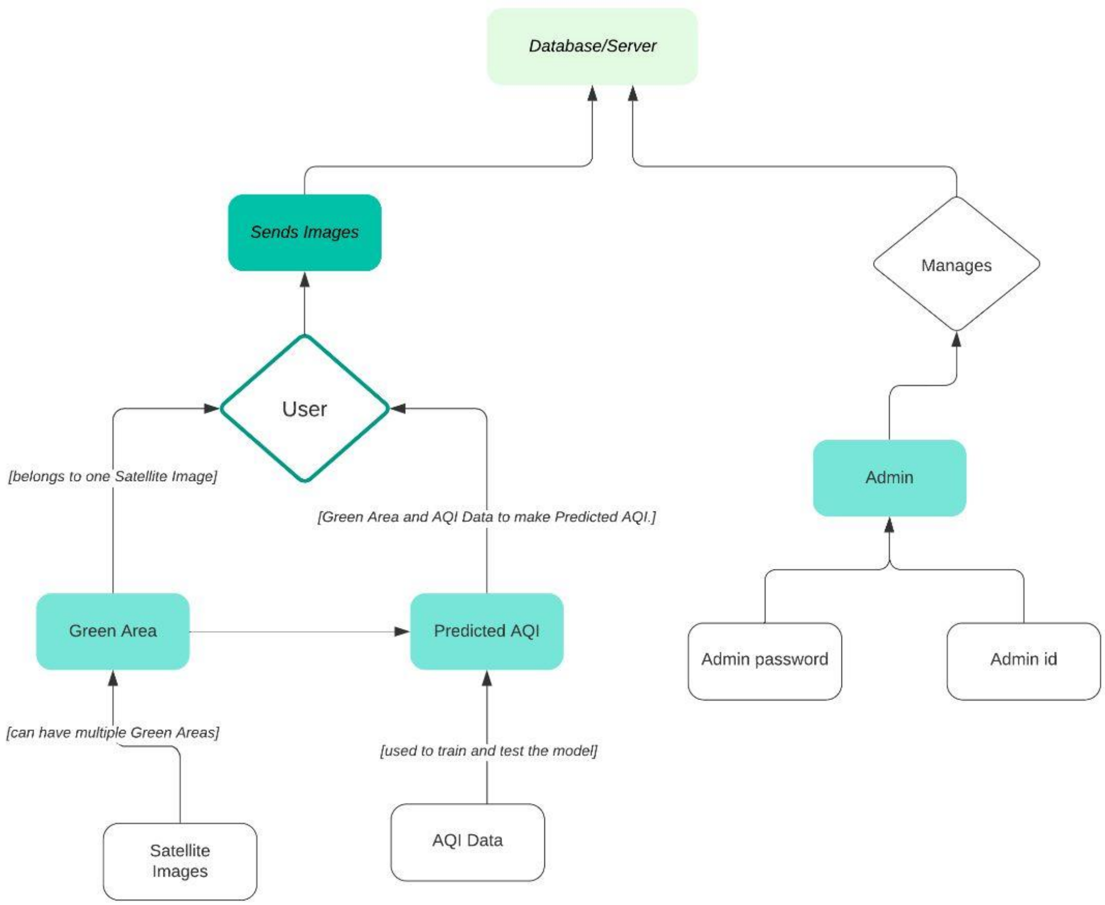


**Figure 3: Entity-Relationship model**

### 3.1.3 Class Diagram

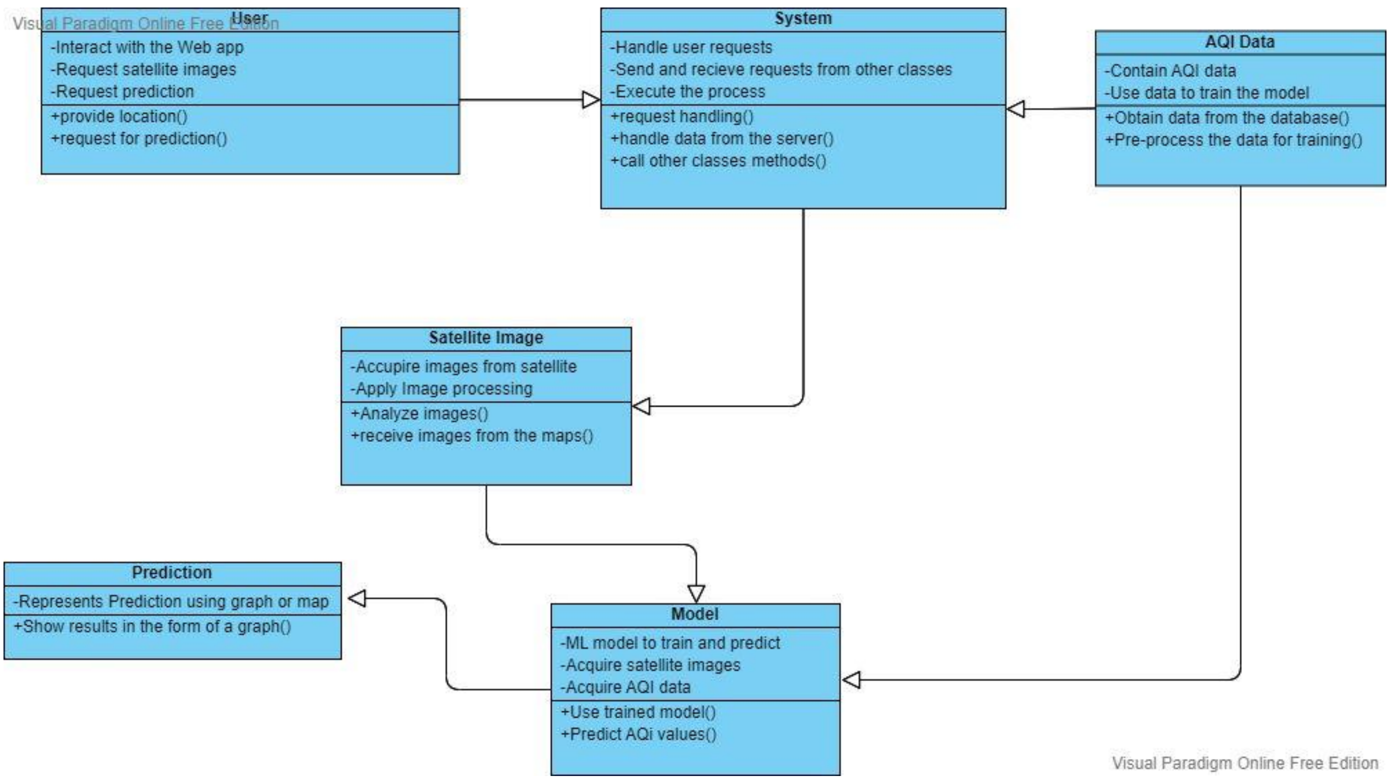


**Figure 4: Class Diagram**

## 3.2 Component Interactions and Collaborations

### 3.2.1 Design Level Sequence Diagram

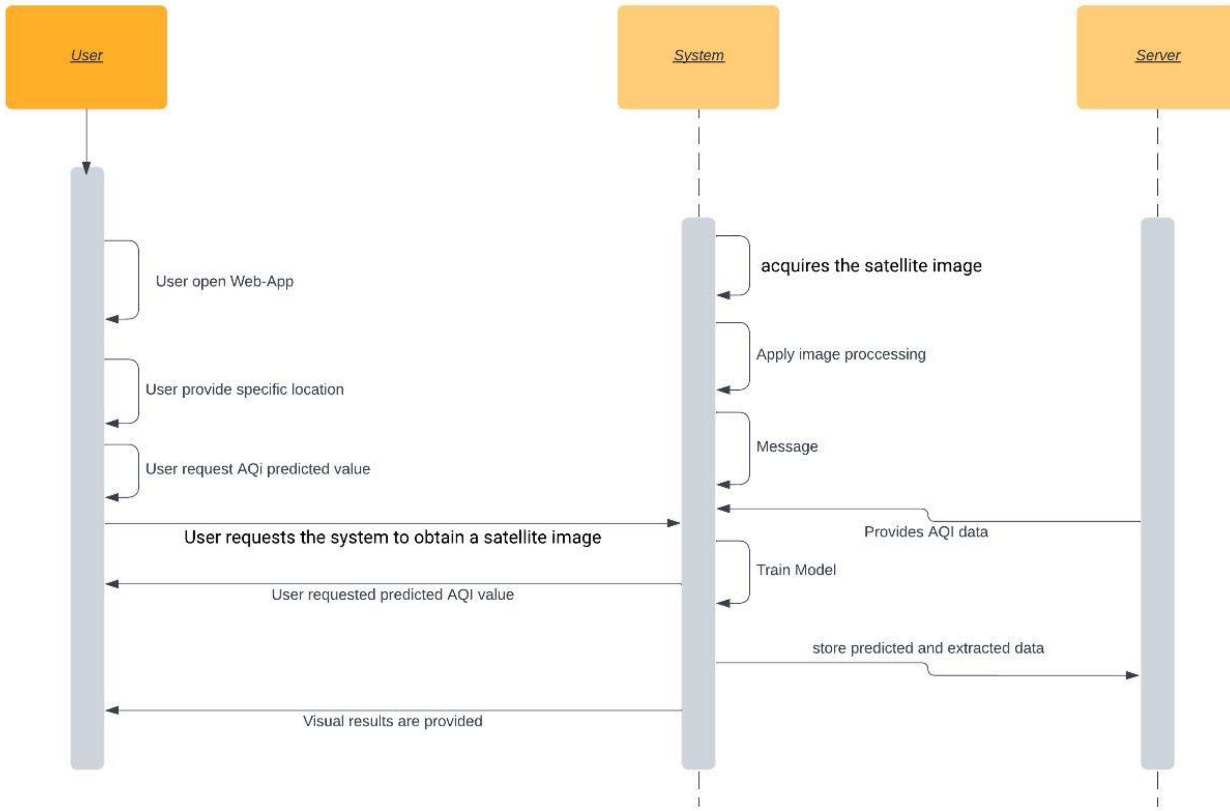


**Figure 5: Sequence Diagram**

### 3.2.2 Data Flow Diagram

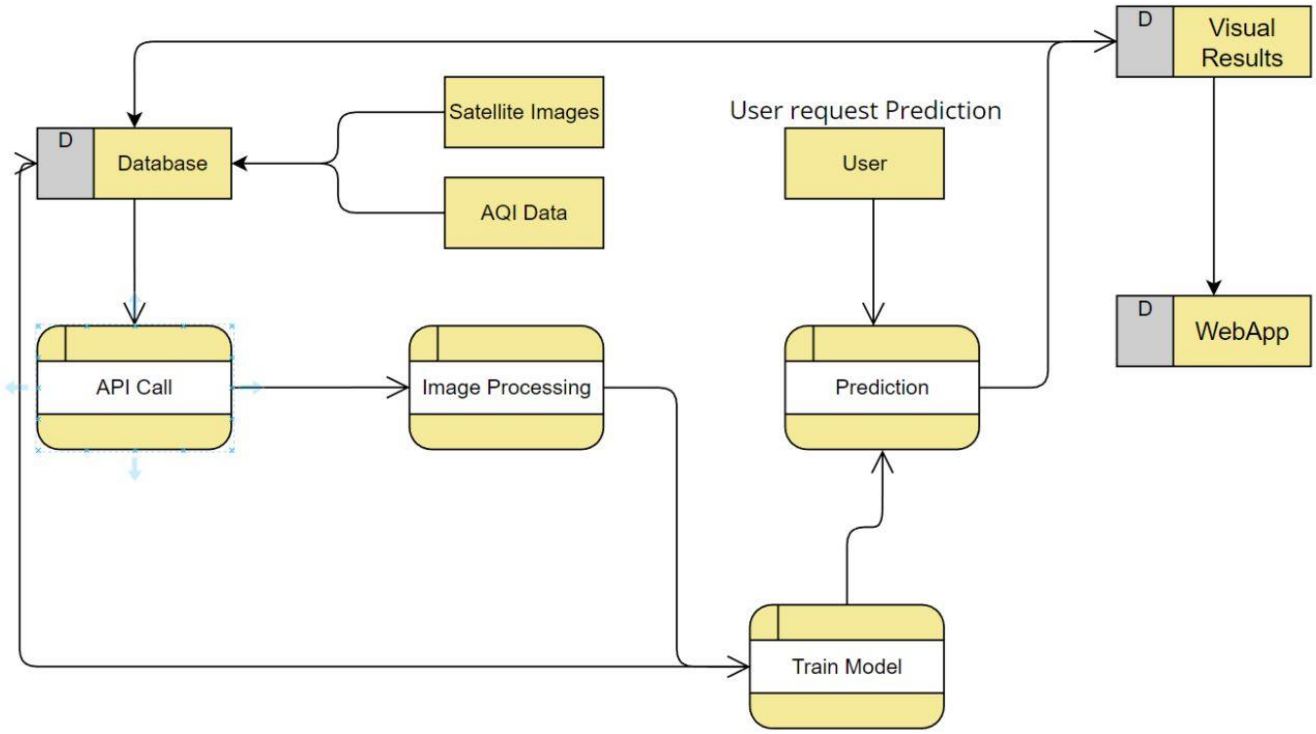


**Figure 6: Data Flow Diagram**

## 3.3 System Architecture

There are different components of our proposed system. To begin with, our front-end complies of a web-based user interface built using Django's built-in template engine and front-end libraries such as Bootstrap that allows user to visualize the predictions, request new location, and access historical data. The back-end of our system which is the server side application is built using Django web framework using tailwind CSS, HTMLwhich handles the image processing, data preparation, LSTM model training and prediction tasks. The back-end is also using libraries such as OpenCV, Pandas, numpy, matplotlib, and TensorFlow to have the system functioning. SQLite is used as a database which is supported by Django framework. Moreover, for machine learning model which incorporates data processing, model selection, training, evaluation, and deployment of ML models, we have used sickit-learn etc. There are many ways to collect data to run machine learning algorithms on it.

For our project, we have collected dataset from more than just one authentic website. As an image is fetched through Google API, it is converted to a NumPy array, and then read into OpenCV format. The green covered area is detected and scanned through hues. The data has to be preprocessed in order to remove the unwanted data and to make the data useful. The preprocessing stage is important because it helps us to transform the data into a useful format. Our main step involved in data preprocessing has been feature scaling. By using sickit-learn library, we have used this method to normalize the range of our independent variables or the features of our data. This method is specifically used to bring all the features in the same range. There are a range of methods to scale your data. We have used Normalization. The general formula for normalization is:

$$X' = \frac{x - \min(x)}{\mathrm{Max}(x) - \min(x)}$$

The data is transformed to improve the structure and format for better data driven decision making.

After collecting the dataset, and performing data pre-processing techniques like feature scaling and data transformation. After the data is ready to be used, we put our data in arrays with two parameters with one having the air particles composition and the other one having the AQI data. We have considered PM2.5 variable for the prediction of our AQI variable. This is how the data is reshaped. The machine learning algorithm of Time-Series Analysis has been used to build a model for prediction. This particular method obtains inferences through observing occurrences based on past data with the data point series and predicts what is going to happen in the future. It is specifically used to for prediction. The same preprocessed data is used to train the model with the identified algorithms. The built models will then be tested by calculating performance metrics. The performance metrics will test how well the model is predicting the AQI.

Since our project deals with prediction, we have considered Mean Absolute Error and Root Mean Square Error. We are validating our machine learning models by comparing the performance metrics. The lower the MAE, RSME, the machine learning model performs better.

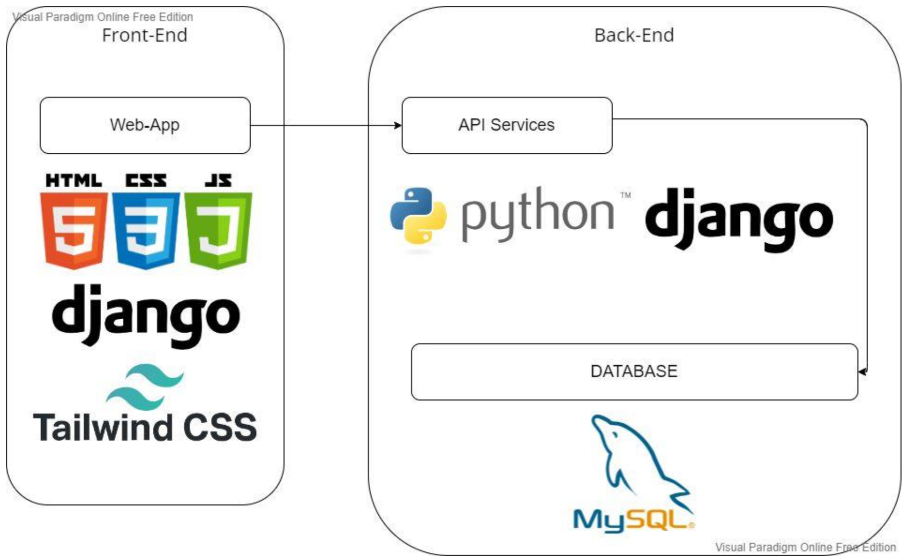


**Figure 7: System**

## 3.4 Architecture Evaluation

We have used CNN (Convolutional Neural Network) and LSTM (Long Short Term Memory) model as they will both work well together to perform image processing and prediction tasks. LSTM is designed to maintain memory over a long sequence of inputs. This allows LSTMs to capture long-term dependencies and patterns in the data, making them effective for prediction tasks. On the other hand, CNNs are better in image recognition and processing tasks because they are designed to capture local patterns and features in images. In prediction tasks, the data has a string temporal aspect, LSTM is a better choice than CNN because of their ability to maintain memory. In tasks where the data is visual or spatial in nature, we have taken CNNs as a better choice. Alternative model such as RNN (Recurrent Neural Network) is also used for prediction tasks. However, they are less sophisticated than LSTMs and struggle to capture long-term dependencies in the data. On the other hand, GRUs (Gated Recurrent Unit) are faster to train and more computationally efficient models but we have used LSTM because it is best suited for the size and structure of the data.

Moreover, we have not used the NASA's API even though the images in that API were of an extremely high definition. The reason behind this is that such HDR images require a lot of time to be processed which we could not afford.

We have used Django framework for our web interface for a couple of reasons. Django has a large number of built-in features, such as the templating engine and more. For our scope of project, Django was best suited for our scale web interface and handle the traffic.

## 3.5 Component-External Entities Interface

The satellite image provider is one external entity which is providing the system with satellite images, the interface is Goggle Maps API which accepts parameters such as location, date or resolution and returns an image. AQI data provider is another external entity in our system.

## 3.6 Screenshots/Prototype

### 3.6.1 Workflow

Our project "Evergreen Surveillance" uses image processing to detect the green covered area in sqkm from satellite images and then uses that information, along with air quality data to predict future AQI values using an LSTM model and the Django web framework. The system starts by acquiring satellite images of a specific location and applies image processing techniques to extract information about vegetation, such as the green covered area using libraries like OpenCV and sickit-image. The system then organizes and pre-processes that data like cleaning and normalizing it and creating a time-series dataset. Then, the AQI data is used to train an LSTM model using TensorFlow. The model is then validated to ensure that it is accurate. The system then deploys the trained model in the back-end. When a user makes a request to the system through the front-end which is handled by Django's views and controllers, the system uses the deployed LSTM model to make a prediction about future AQI values for the location of user's interest. The results of predicted AQI are shown for three different days i.e. after 1$^{st}$, 15$^{th}$ and 30$^{th}$ day of current AQI.

### 3.6.2 Screens

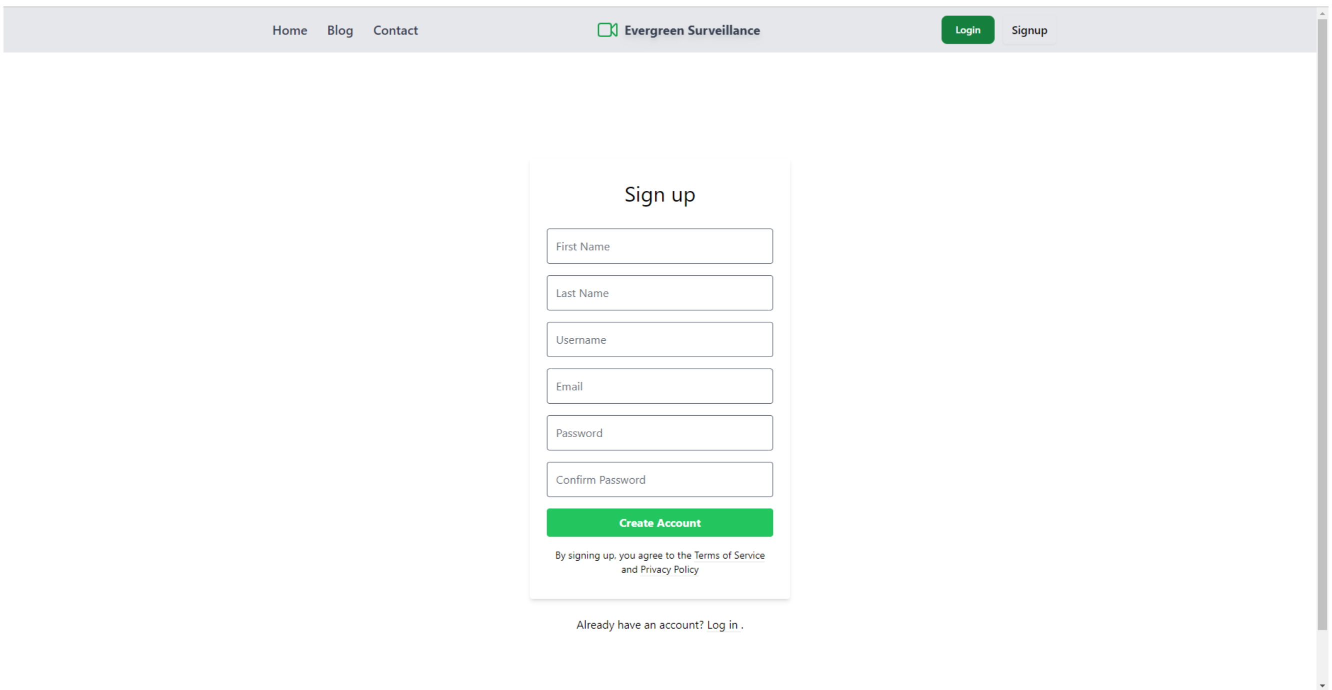


**Figure 8: Sign Up Page**

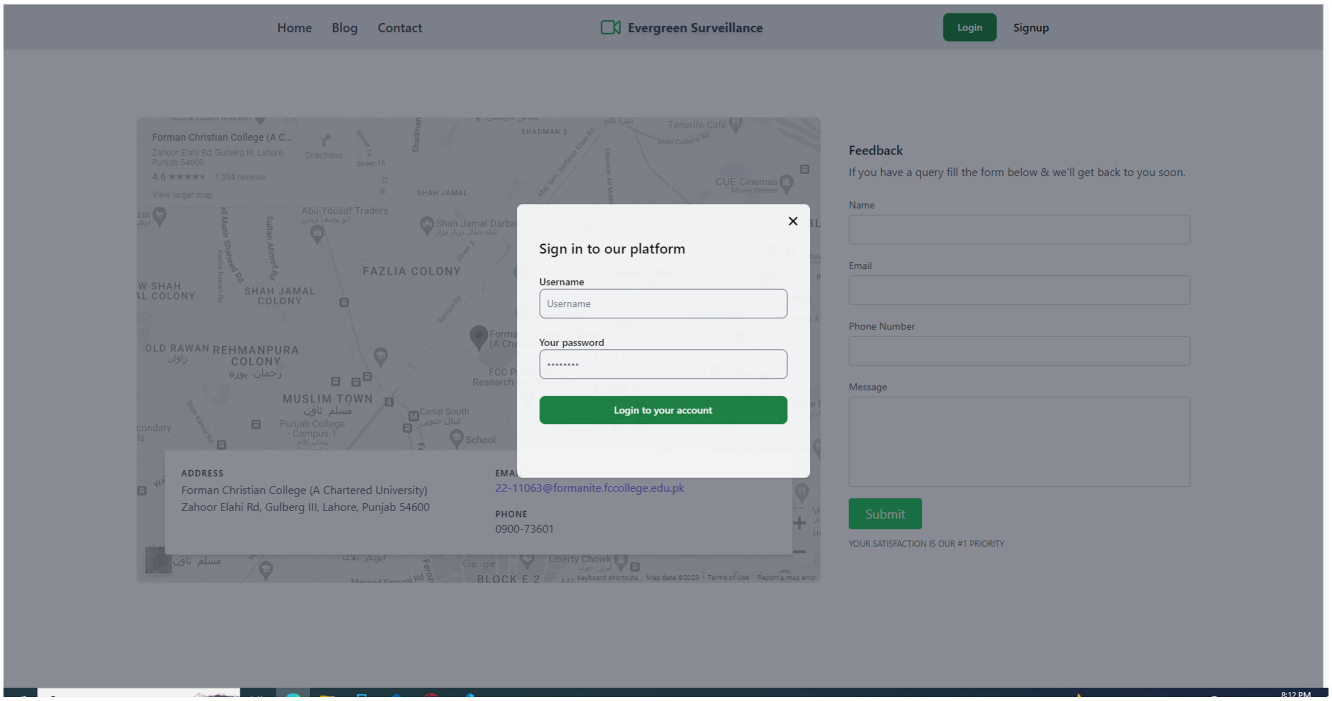


**Figure 9: Login Page**

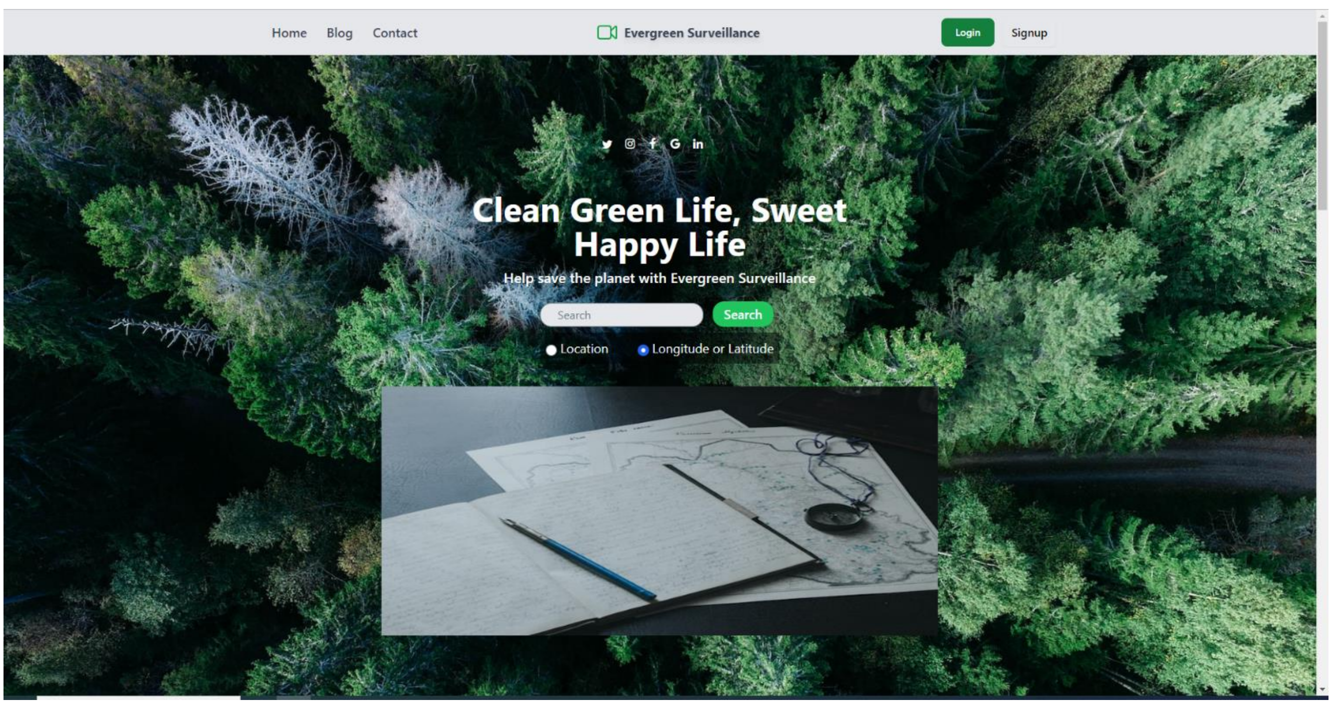


**Figure 10: Homepage**

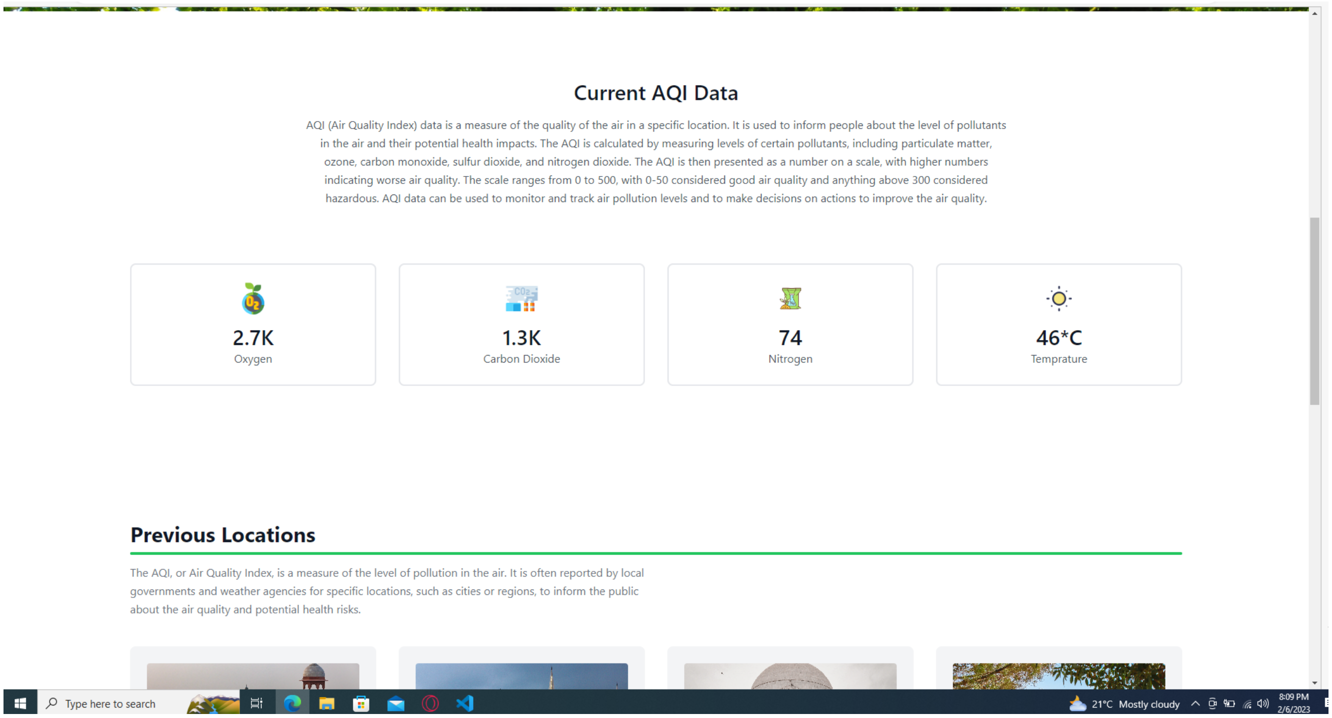


Figure 11: Homepage

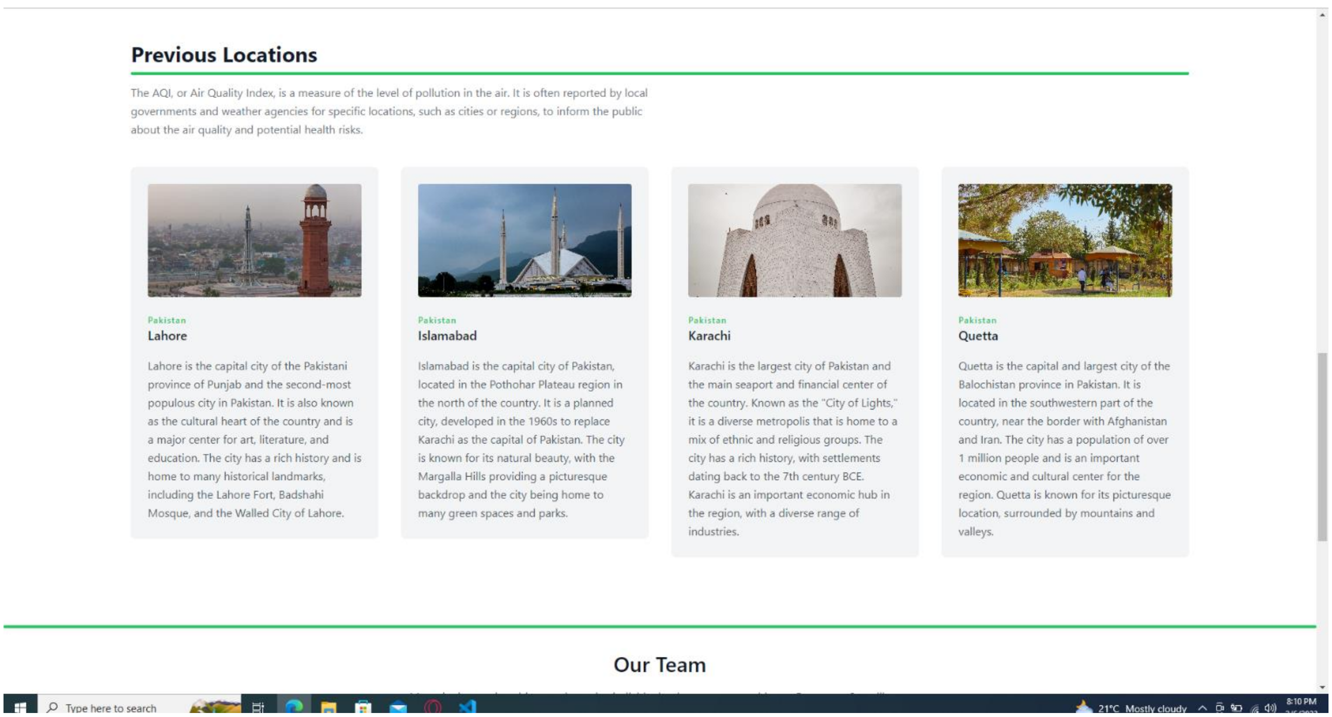


Figure 12: Homepage

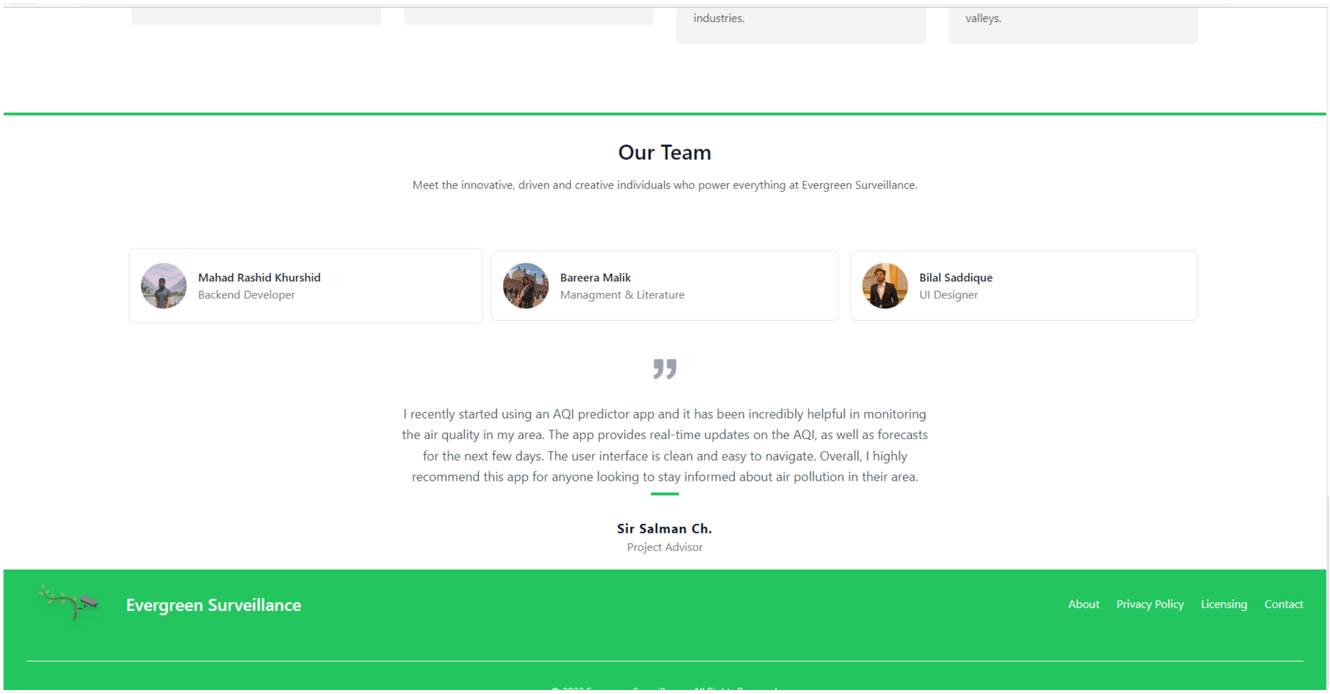


**Figure 13: Homepage**

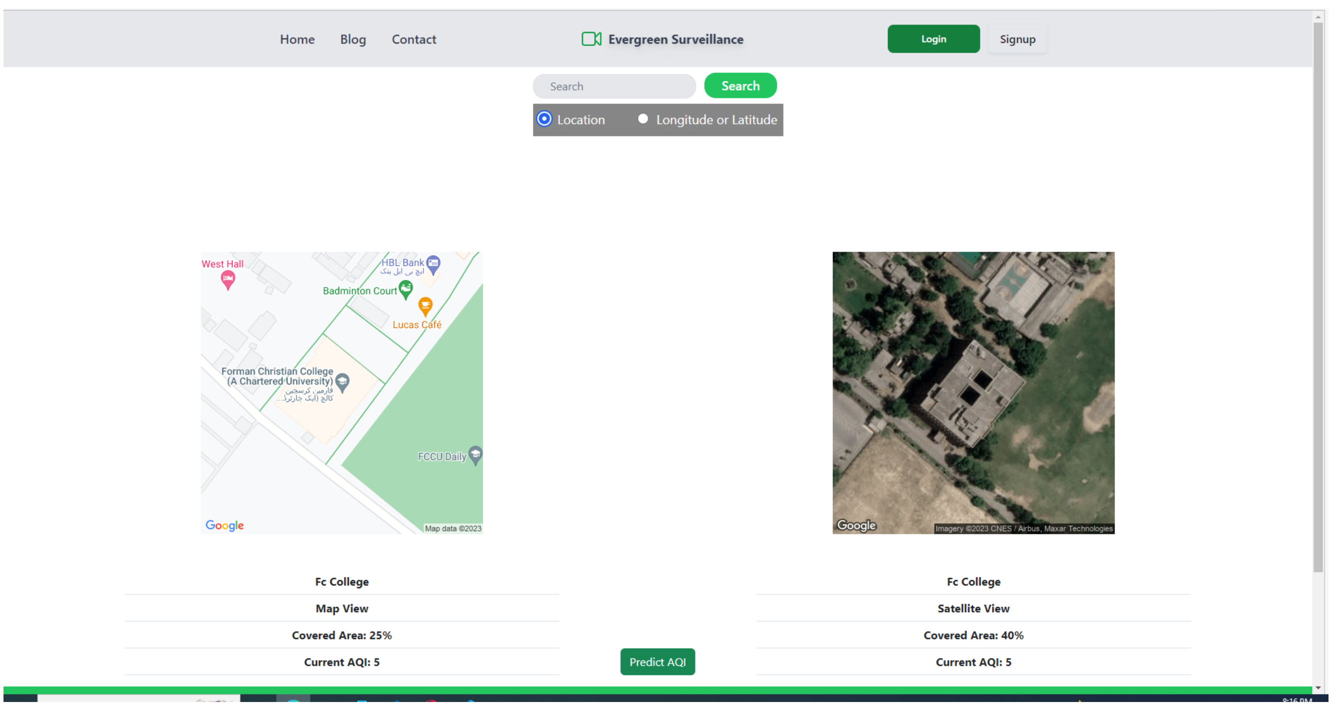


**Figure 14: Result Page**

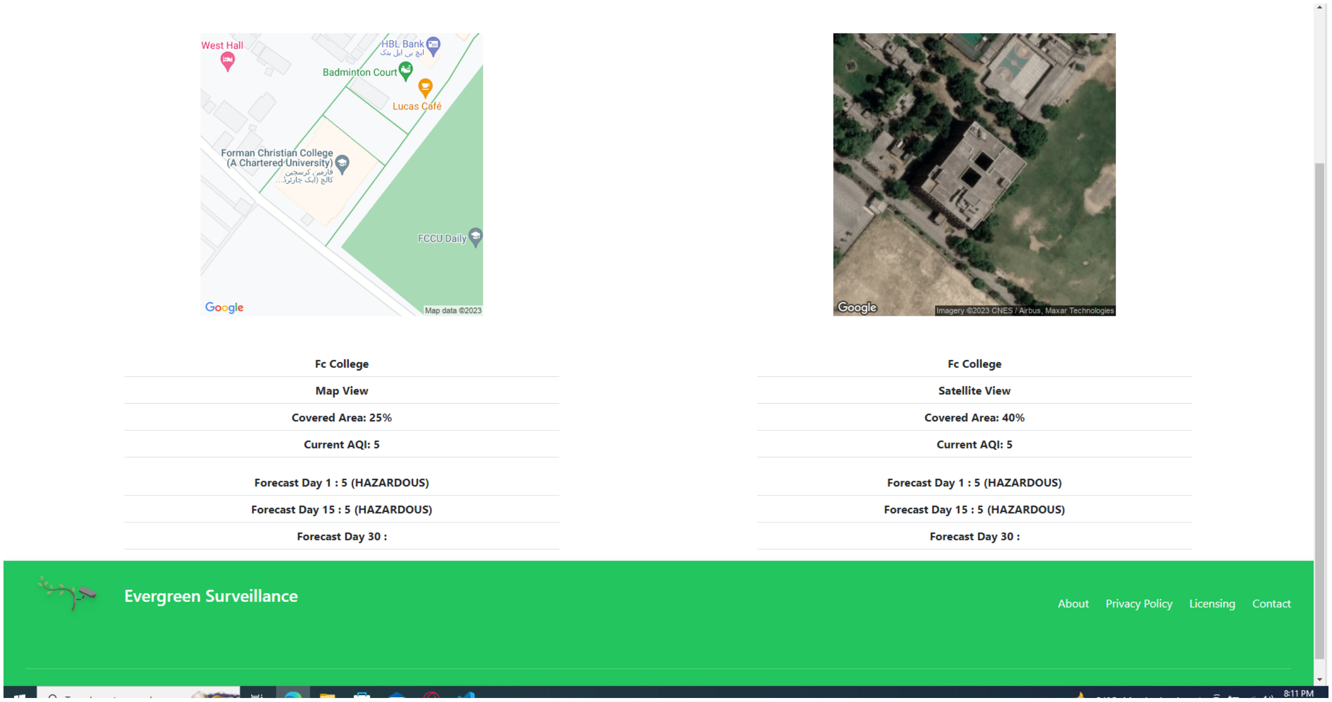


**Figure 15: Result Page**

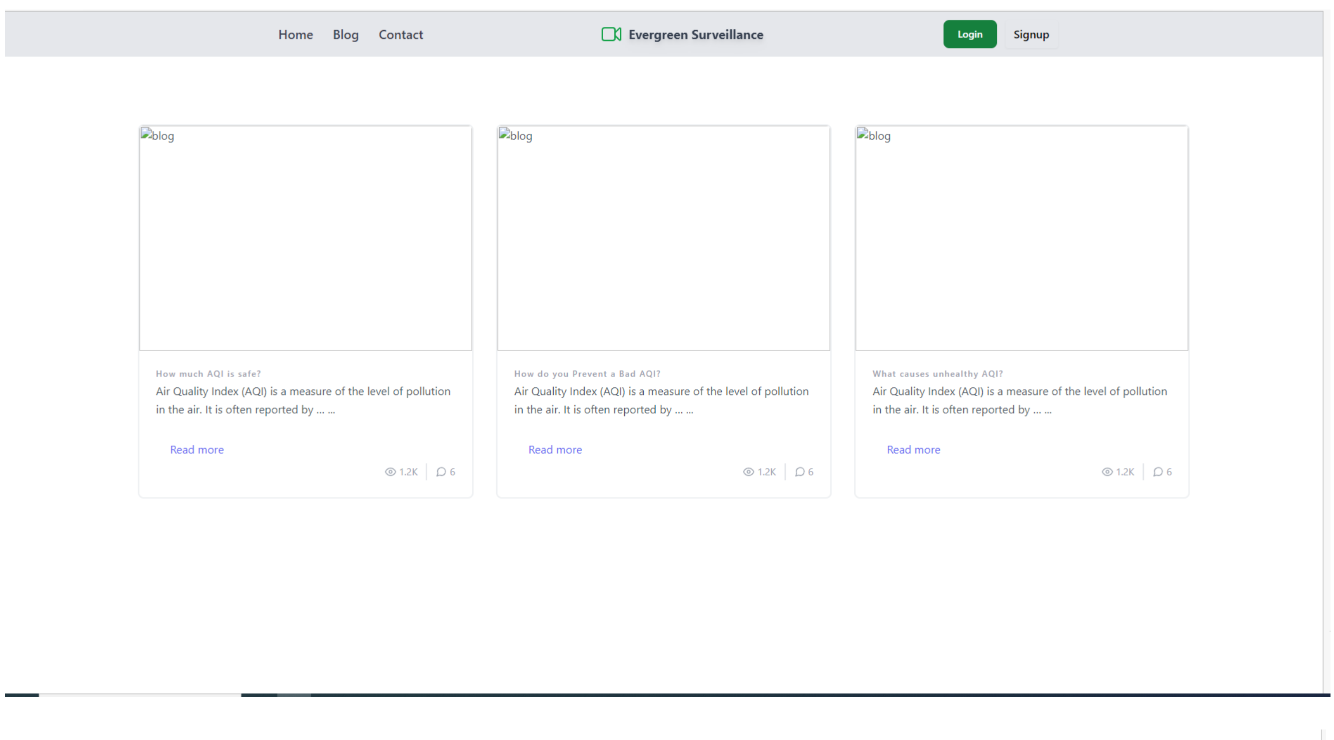


**Figure 16: Blog Page**

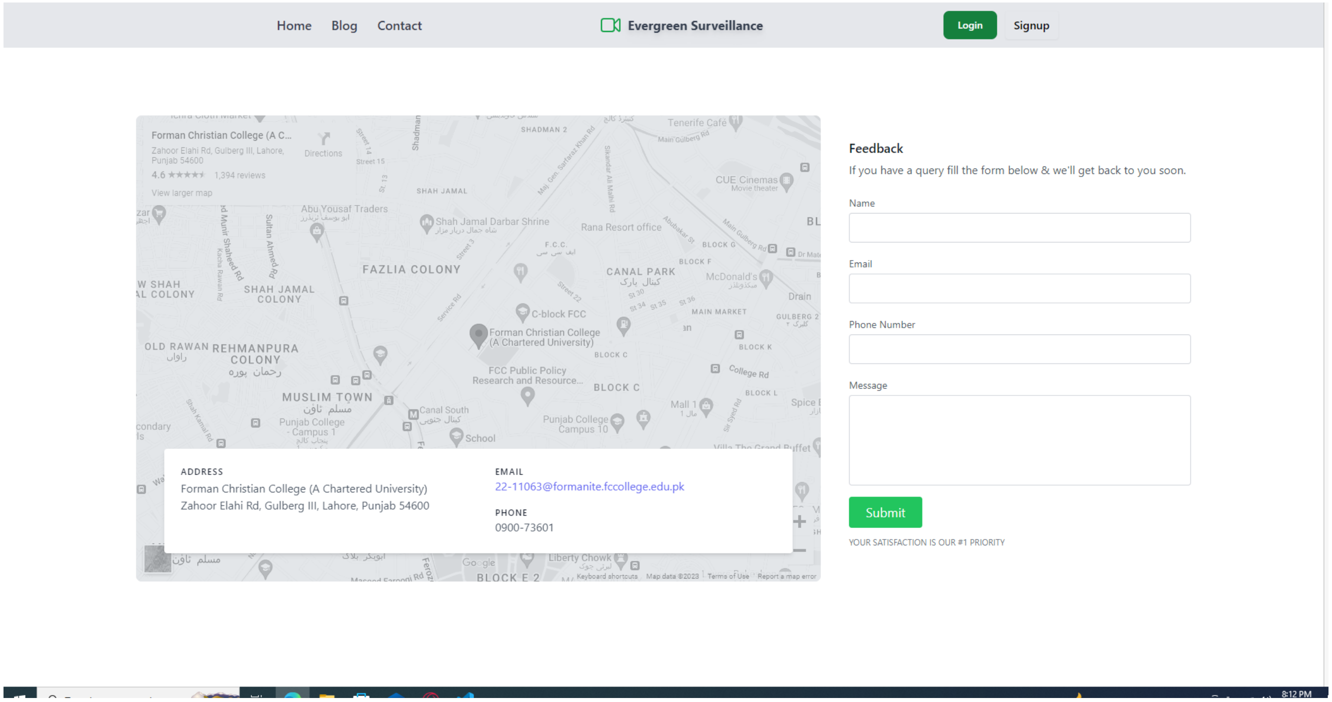


**Figure 17: Contact Form**

## 3.7 Other Design Details

All the design details are covered in the previous sections respectively.

# Chapter 4. Test Specification and Results

## 4.1 Test Case Specification

### 4.1.1 TC-1

| | |
|---|---|
| **Identifier** | TC-1 |
| **Related requirements(s)** | UC-1: User opens the website |
| **Short description** | The User opens the website |
| **Pre-condition(s)** | User has an internet connection and knows the url of the website. |
| **Input data** | User enters the correct URL. |
| **Detailed steps** | User will enter the correct URL to access the website |
| **Expected result(s)** | The website will open to its homepage and will be welcomed with a welcome message. |
| **Post-condition(s)** | The user will be able to use the website. |
| **Actual result(s)** | The user will be able to use the website. |
| **Test Case Result** | The user can use the website and interact with it and make use of the features. |

**Table 6.1: TC-1**

### 4.1.2 TC-2

| | |
|---|---|
| **Identifier** | TC-2 |
| **Related requirements(s)** | UC-2: Give Access to API |
| **Short description** | The website requests the user to give access to the API. |
| **Pre-condition(s)** | User has an internet connection. |
| **Input data** | User clicks on the Get Started Button. |
| **Detailed steps** | A pop-up screen asks the user's permission to access the API. If the user has agreed, then the application proceeds as normal. If the user declines, then the user cannot proceed, and if the user exits the application, then the application is closed. |
| **Expected result(s)** | If the user has agreed to give access, he will be allowed to use the system further. If the user denied access, the website will inform the user that it needs to provide it with access first. |
| **Post-condition(s)** | The user will give access to the website. |
| **Actual result(s)** | The user can proceed further after giving access |
| **Test Case Result** | The user can use the website and interact with it and make use of the features. |

**Table 6.2: TC-2**

### 4.1.3 TC-3

| | |
|---|---|
| **Identifier** | TC-3 |
| **Related requirements(s)** | UC-3: Give Access To Location |
| **Short description** | The website requests the user to give access to the geolocation. |
| **Pre-condition(s)** | User has an internet connection. |
| **Input data** | User clicked on “Get Started” button. |
| **Detailed steps** | A pop-up screen asks the user’s permission to access the geolocation. If the user has agreed, then the application proceeds as normal. If the user declines, then the user cannot proceed, and if the user exits the application, then the application is closed. |
| **Expected result(s)** | If the user has agreed to give access, he will be allowed to use the system further. If the user denied access, the website will inform the user that it needs to provide it with location’s access first. |
| **Post-condition(s)** | The system has access to the user’s geolocation |
| **Actual result(s)** | The system has access to the user’s geolocation |
| **Test Case Result** | The system has access to the user’s geolocation |

**Table 6.3: TC-3**

### 4.1.4 TC-4

| | |
|---|---|
| **Identifier** | TC-4 |
| **Related requirements(s)** | UC-4: Take Snapshot |
| **Short description** | The user takes a snapshot of green covered area. |
| **Pre-condition(s)** | • User has an internet connection.<br>• Access to API<br>• Access to user's geolocation |
| **Input data** | User presses on the take snapshot button |
| **Detailed steps** | • The user presses on the take snapshot button<br>• The user then either decides to take the snapshot or not<br>• The user takes the snapshot<br>• Otherwise if the user does not take the snapshot and presses the back button, the user quits the application. |
| **Expected result(s)** | The system gets the user the option to upload or retake snapshot. |
| **Post-condition(s)** | The user will have to retake snapshot of trees |
| **Actual result(s)** | The system gets the user the option to upload or retake snapshot. |
| **Test Case Result** | The system gets the user the option to upload or retake snapshot. |

**Table 6.4: TC-4**

### 4.1.5 TC-5

| | |
|---|---|
| **Identifier** | TC-5 |
| **Related requirements(s)** | UC-5: Retake Snapshot |
| **Short description** | The user retakes a snapshot of green covered area. |
| **Pre-condition(s)** | • User has an internet connection.<br>• Access to API<br>• Access to user's geolocation |
| **Input data** | User presses on the retake snapshot button |
| **Detailed steps** | • The user presses on the retake snapshot button<br>• The user then either decides to take the snapshot or not<br>• The user takes the snapshot<br>• Otherwise if the user does not take the snapshot and presses the back button, the user quits the application. |
| **Expected result(s)** | The system gets the user the option to upload or retake snapshot. |
| **Post-condition(s)** | The user will have to retake snapshot of trees |
| **Actual result(s)** | The system gets the user the option to upload or retake snapshot. |
| **Test Case Result** | The system gets the user the option to upload or retake snapshot. |

**Table 6.5: TC-5**

### 4.1.6 TC-6

| | |
|---|---|
| **Identifier** | TC-6 |
| **Related requirements(s)** | UC-6: Upload the Snapshot |
| **Short description** | The user uploads the snapshot |
| **Pre-condition(s)** | • User has an internet connection.<br>• Access to the location<br>• User must have taken a snapshot |
| **Input data** | A snapshot and the geolocation of the user when the snapshot was taken |
| **Detailed steps** | • The user will press on the upload button<br>• The user will be informed if the snapshot has been uploaded or not. |
| **Expected result(s)** | The system tells the user that the image has been successfully uploaded. |
| **Post-condition(s)** | The image is uploaded to the server |
| **Actual result(s)** | The system tells the user that the image has been successfully uploaded. |
| **Test Case Result** | The system tells the user that the image has been successfully uploaded. |

**Table 6.6: TC-6**

### 4.1.7 TC-7

| | |
|---|---|
| **Identifier** | TC-7 |
| **Related requirements(s)** | UC-7: Analyze Image |
| **Short description** | The server analyzes the image. |
| **Pre-condition(s)** | The user has uploaded the snapshot. |
| **Input data** | The image as well as the location set by the user. |
| **Detailed steps** | The image is analyzed by the server and green covered area is found. |
| **Expected result(s)** | The server will report the green covered area in sqkm |
| **Post-condition(s)** | The green covered area in sqkm will be on the screen. |
| **Actual result(s)** | The server will report the green covered area in sqkm |
| **Test Case Result** | The server will report the green covered area in sqkm |

**Table 6.7: TC-7**

### 4.1.8 TC-8

| | |
|---|---|
| **Identifier** | TC-8 |
| **Related requirements(s)** | UC-8: Admin Login |
| **Short description** | The admin tries to log into the system |
| **Pre-condition(s)** | The user has an internet connection. |
| **Input data** | Admin's credentials |
| **Detailed steps** | The admin enters his credentials into the respective fields. If the credentials are valid, the admin is redirected to the server. Otherwise, an error message is shown |
| **Expected result(s)** | Login in case of success and an error message in case of failure |
| **Post-condition(s)** | The admin is logged into the server. |
| **Actual result(s)** | The user gets logged in and directed to the homepage. |
| **Test Case Result** | The user gets logged in and directed to the homepage. |

**Table 6.8: TC-8**

### 4.1.9 TC-9

| | |
|---|---|
| **Identifier** | TC-9 |
| **Related requirements(s)** | UC-9: Store Geolocation and image |
| **Short description** | The geo location and image are saved |
| **Pre-condition(s)** | The image has been analyzed |
| **Input data** | |
| **Detailed steps** | The server returns the value of green covered area. |
| **Expected result(s)** | The green covered area is reported |
| **Post-condition(s)** | The area in sqkm is returned |
| **Actual result(s)** | The green covered area is reported |
| **Test Case Result** | The green covered area is reported |

**Table 6.9: TC-9**

### 4.1.10 TC-10

| | |
|---|---|
| **Identifier** | TC-10 |
| **Related requirements(s)** | UC-10: Accessing the AQI database |
| **Short description** | The system access the AQI database |
| **Pre-condition(s)** | |
| **Input data** | |
| **Detailed steps** | The system gains access to the AQI data and trains a model |
| **Expected result(s)** | Access to the AQI data will be granted |
| **Post-condition(s)** | Access to the AQI database |
| **Actual result(s)** | Access to the AQI data will be granted |
| **Test Case Result** | Access to the AQI data will be granted |

**Table 6.10: TC-10**

### 4.1.11 TC-11

| | |
|---|---|
| **Identifier** | TC-11 |
| **Related requirements(s)** | UC-11: Access the predicted AQI value |
| **Short description** | The system generates the predicted AQI value |
| **Pre-condition(s)** | AQI database is granted |
| **Input data** | AQI data |
| **Detailed steps** | The system predicts an AQI value for future days |
| **Expected result(s)** | The server will send the AQI value |
| **Post-condition(s)** | Access to the AQI database |
| **Actual result(s)** | The server will send the AQI value |
| **Test Case Result** | The server will send the AQI value |

**Table 6.11: TC-11**

### 4.1.12 TC-12

| | |
|---|---|
| **Identifier** | TC-12 |
| **Related requirements(s)** | UC-12: Generating graphs |
| **Short description** | The AQI value is visualized |
| **Pre-condition(s)** | Results are present |
| **Input data** | AQI values |
| **Detailed steps** | The user sees the results in the form of graphs |
| **Expected result(s)** | The website will show the result graphs on the webpage |
| **Post-condition(s)** | The graphs are shown on the website |
| **Actual result(s)** | The website will show the result graphs on the webpage |
| **Test Case Result** | The website will show the result graphs on the webpage |

**Table 6.12: TC-12**

## 4.2 Summary of Test Results

| **Module Name** | Test cases run | Number of defects found | Number of defects corrected so far | Number of defects still need to be corrected |
|---|---|---|---|---|
| **Website Module** | TC1, TC2, TC3, TC4, TC5, TC6 | None | None | None |
| **Server Module** | TC8, TC10, TC11, TC12 | 2 | 2 | None |
| **Database Module** | TC7, TC9 | 3 | 1 | 2 |
| **Complete System** | TC1 to TC12 | 5 | 3 | 2 |

**Table 6.2: Summary of All Test Results**

# Chapter 5. Conclusion and Future Work

## 5.1 Project summary

With the department of climate change and meteorology being the primary body that will need this system, there are other organizations and institutes that need this update in account of the problems Listed above. Plantation drive initiators and urban forest projects also require this system so they are aware of the current situation of the depleting green cover in an area and can plan out their projects accordingly. Laws against cutting down trees can only be implemented if authorities are aware of when trees are cut down and where. Through our project, reports can be generated highlighting the spots wherever any sort of violation occurs. Other than this, monitoring AQI will awaken the concerned authorities so necessary actions can be taken to improve the quality of the air.

## 5.2 Problems faced and lessons learned

The major problems that we faced while designing and implementing our system were:
Sometimes the quality of the satellite images varied and so it became necessary to clean and pre-process the data before using it to train the model, this can be time consuming and may require effort. It is difficult to achieve accurate predictions due to the complexity of the data. As the amount of data and user traffic increases, it may become necessary to scale the system to handle the load. This can be very challenging. One thing that we realized was that it is important to thoroughly test the system before deployment to ensure its reliability.
By learning these lessons, we have tried to make the project improve its performance, accuracy, and user satisfaction.

## 5.3 Future work

The future work of our AQI prediction and green covered area detection project can include a number of different directions. The accuracy of the AQI prediction model could be improved by using more sophisticated machine learning algorithms or by incorporating additional data sources. The green covered area detection system could be expanded to other regions to provide a more comprehensive picture of the relationship between green spaces and air quality. Predictive maintenance could be integrated into the green covered area detection system to identify areas that need maintenance or intervention to maintain air quality. The AQI prediction and green covered area detection systems could be combined with other environmental data sources, such as weather/temperature data and pollution data, to provide a more complete picture of the environmental conditions.

# References


1. https://www.researchgate.net/publication/285593710_Tree_Detection_and_Species_Identification_using_LiDAR_Data

2. https://www.globalforestwatch.org/dashboards/country/PAK/?category=summary&location=WyJjb3VudHJ5IiwiUEFLIl0%3D&map=eyJjYW5Cb3VuZCI6ZmFsc2UsImRhdGFzZXRzIjpbeyJkYXRhc2V0IjoicG9saXRpY2FsLWJvdW5kYXJpZXMiLCJsYXllcnMiOlsiZGlzcHV0ZWQtcG9saXRpY2FsLWJvdW5kYXJpZXMiLCJwb2xpdGljYWwtYm91bmRhcmllcyJdLCJib3VuZGFyeSI6dHJ1ZSwib3BhY2l0eSI6MSwidmlzaWJpbGl0eSI6dHJ1ZX0seyJkYXRhc2V0IjoidHJlZS1jb3Zlci1sb3NzIiwibGF5ZXJzIjpbInRyZWUtY292ZXItbG9zcyJdLCJvcGFjaXR5IjoxLCJ2aXNpYmlsaXR5Ijp0cnVlLCJ0aW1lbGluZVBhcmFtcyI6eyJzdGFydERhdGUiOiIyMDAxLTAxLTAxIiwiZW5kRGF0ZSI6IjIwMjEtMTItMzEiLCJ0cmltRW5kRGF0ZSI6IjIwMjEtMTItMzEifSwicGFyYW1zIjp7InRocmVzaG9sZCI6MzAsInZpc2liaWxpdHkiOnRydWV9fV19&showMap=true

3. https://www.mdpi.com/2072-4292/12/18/3017/htm

4. https://www.researchgate.net/publication/365875295_Detection_and_mapping_of_green-cover_and_landuse_changes_by_advanced_satellite_image_processing_techniques_a_case_study

5. https://www.sciencedirect.com/science/article/pii/S2214317316300154

6. https://www.nature.com/articles/s41598-022-12355-6

7. https://www.scirp.org/journal/paperinformation.aspx?paperid=120893

# Appendix A Glossary

### A

**API –** Application Program Interface (API) is a set of routines, protocols, and tools for building software applications.

**AQI** – Air Quality Index is the unit by which the quality of air of a specific area is measured.

**Afforestation –** It is an act of planting trees.

### C

**CNN –** Convolutional Neural Network is a kind of artificial neural network used for image processing.

### D

**Data Preprocessing –** It refers to cleaning, transforming and integration of data through various techniques to make the data more useful.

**Deforestation –** it is the act of cutting down trees.

### I

**Image Processing –** It is the process of transforming an image into a digital form to perform certain operations to get required information out of it.

### L

**LSTM –** Long Short Term Memory is an artificial neural network used in deep learning used for entire sequences of data.

### M

**ML –** Machine Learning is a part of artificial intelligence which has the capability to copy intelligent human behavior.

### N

**Normalization –** It is a scaling technique and a stage in pre-processing to make all the values lie in the same scale.

# Appendix B Deployment/Installation Guide

Our system is based on web-interface so the users do not need to download or install any file or resource in order to use our system.

# Appendix C User Manual

The users of Evergreen Surveillance must follow the user manual below to fully make use of the system:

- Search for Evergreen Surveillance on the browser or open the website through URL.
- Create an account by entering First name, Last name, Username, email address and password.
- After signing up, you can login using your credentials.
- Once you have access to the homepage of the website, you will enter your desired location in the search bar.
- Location can be entered using longitude and latitude as well.
- After entering a location, a map and satellite view will appear on the screen.
- Under the maps, respective green covered areas, current AQI values and predicted AQI values will be shown.

# Appendix D Student Information Sheet

| Roll No | Name | Email Address (FC College) | Frequently Checked Email Address | Personal Cell Phone Number |
|---|---|---|---|---|
| 22-10293 | Bareera Malik | 22-10293@formanite.fcollege.edu.pk | bareeramalik2000@gmail.com | 0334 4507296 |
| 22-11345 | Bilal Saddique | 22-11345@formanite.fccollege.edu.pk | 22-11345@formanite.fccollege.edu.pk | 0323 7813133 |
| 22-11108 | Mahad Rashid | 22-11108@formanite.fccollege.edu.pk | mahadrashid0@gmail.com | 0324 4479603 |

# Appendix E Plagiarism Free Certificate

This is to certify that, I am _Mahad Rashid Khurshid__ S/D/o _Rashid Khurshid__, group leader of FYP under registration no ___22-11108______________ at Computer Science Department, Forman Christian College (A Chartered University), Lahore. I declare that my Final year project report is checked by my supervisor and the similarity index is ________% that is less than 20%, an acceptable limit by HEC. Report is attached herewith as Appendix F. To the best of my knowledge and belief, the report contains no material previously published or written by another person except where due reference is made in the report itself.

Date: _____________ Name of Group Leader: __________________________ Signature: ______________

| | | | |
|---|---|---|---|
| Name of Supervisor: | ______________________ | Co-Supervisor (if any): | ______________________ |
| Designation: | ______________________ | Designation: | ______________________ |
| Signature: | ______________________ | Signature: | ______________________ |

Senior Project Management Committee Representative: ______________________

Signature: ______________________

## Appendix F Plagiarism Report