

“Automated Feedback System for Non-functional Streetlight”

**A Project Report Submitted to
Rajiv Gandhi Proudtyogiki Vishwavidyalaya**



**Towards Partial Fulfillment for the Award of
Bachelor of Engineering in Computer Science Engineering**

Submitted by:

**Mayank Solanki(0827CS201134)
Kanishk Chouhan(0827CS201112)
Khushboo Sen(0827CS201114)
Mahak Soni(0827CS201127)**

Guided by:

**Mr. Ambrish Shrivastav
Professor, CSE
AITR, Indore**



***Acropolis Institute of Technology & Research, Indore
July - Dec 2022***

EXAMINER APPROVAL

The Project entitled “**Automated Feedback System for Non-functional Streetlight**” submitted by **Mayank Solanki (0827CS201134)** , **Kanishk Chouhan (0827CS201112)** , **Khushboo Sen (0827CS201114)** , **Mahak Soni (0827CS201127)** has been examined and is hereby approved towards partial fulfillment for the award of ***Bachelor of Technology degree in Computer Science Engineering*** discipline, for which it has been submitted. It understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein, but approve the project only for the purpose for which it has been submitted.

(Internal Examiner)

Date:

(External Examiner)

Date:

GUIDE RECOMMENDATION

This is to certify that the work embodied in this project entitled “**Automated Feedback System for Non-functional Streetlight**” submitted by **Mayank Solanki (0827CS201134)** , **Kanishk Chouhan (0827CS201112)** , **Khushboo Sen (0827CS201114)** , **Mahak Soni (0827CS201127)** is a satisfactory account of the bonafide work done under the supervision of Mr. Ambrish Shrivastav is recommended towards partial fulfillment for the award of the Bachelor of Technology (Computer Science Engineering) degree by Rajiv Gandhi Proudyogiki Vishwavidhyalaya, Bhopal.

(Project Guide)

(Project Coordinator)

STUDENTS UNDERTAKING

This is to certify that project entitled “**Automated Feedback System for Non-functional Streetlight**” has developed by us under the supervision of **Mr.Ambrish Shrivastav** . The whole responsibility of work done in this project is ours. The sole intension of this work is only for practical learning and research.

We further declare that to the best of our knowledge this report does not contain any part of any work which has been submitted for the award of any degree either in this University or in any other University / Deemed University without proper citation and if the same work found then we are liable for explanation to this.

Mayank Solanki (0827CS201134)

Kanishk Chouhan (0827CS201112)

Khushboo Sen (0827CS201114)

Mahak Soni (0827CS201127)

Acknowledgement

We thank the almighty Lord for giving me the strength and courage to sail out through the tough and reach on shore safely.

There are number of people without whom this projects work would not have been feasible. Their high academic standards and personal integrity provided me with continuous guidance and support.

We owe a debt of sincere gratitude, deep sense of reverence and respect to our guide and mentor **Dr. Kamal Kumar Sethi**, Professor, AITR, Indore for his motivation, sagacious guidance, constant encouragement, vigilant supervision and valuable critical appreciation throughout this project work, which helped us to successfully complete the project on time.

We express profound gratitude and heartfelt thanks to **Dr Kamal Kumar Sethi**, Professor & Head CSE, AITR Indore for his support, suggestion, and inspiration for carrying out this project. I am very much thankful to other faculty and staff members of IT Dept, AITR Indore for providing me all support, help and advice during the project. We would be failing in our duty if do not acknowledge the support and guidance received from **Dr S C Sharma**, Director, AITR, Indore whenever needed. We take opportunity to convey my regards to the management of Acropolis Institute, Indore for extending academic and administrative support and providing me all necessary facilities for project to achieve our objectives.

We are grateful to **our parent** and **family members** who have always loved and supported us unconditionally. To all of them, we want to say “Thank you”, for being the best family that one could ever have and without whom none of this would have been possible.

Vedika Shrivastava (0827CS141117), Yash Bhawsar (0827CS141119)

Executive Summary

Real-Time Object Detection and Recognition

This project is submitted to Rajiv Gandhi Proudyogiki Vishwavidhyalaya, Bhopal (MP), India for partial fulfillment of Bachelor of Engineering in Information Technology branch under the sagacious guidance and vigilant supervision of ***Dr. Kamal Kumar Sethi***.

The project is based on Deep Learning, which is a sub field of machine learning, concerned with algorithms inspired by the structure and function of the brain called artificial neural networks. In the project, TensorFlow is used, which is an open-source software library created by Google for machine learning applications. It is used for detecting, identifying and tracking objects through the camera in real time. The project uses a pre-trained model on Microsoft Common Objects in Context (COCO) data set , which contains approximately all common objects. The purpose of this project is to implement 'Students and vehicles counter' in the college in real-time.

Key words : Image Processing, Neural Networks, Tensorflow

List of Figures

| | |
|---|----|
| Figure 1-1 : Counting people at wedding function..... | 3 |
| Figure 1-2: Counting of eggs and soda bottle..... | 4 |
| Figure 1-3 : Counting people at railway stations and airport..... | 4 |
| Figure 3-1 : Block Diagram..... | 16 |
| Figure 3-2 : R-CNN: Regions with CNN Features..... | 18 |
| Figure 3-3 : Fast R-CNN Architecture..... | 18 |
| Figure 3-4 : YOLO Architecture..... | 18 |
| Figure 3-5 : Faster CNN Architecture..... | 19 |
| Figure 3-6 : Bounding-Box..... | 20 |
| Figure 3-7 Data Flow Diagram Level 0..... | 20 |
| Figure 3-8 Data Flow Diagram Level 1..... | 20 |
| Figure 4-1 : Deep Learning..... | 24 |
| Figure 4-2 : Neural Networks..... | 25 |
| Figure 4-3 : TensorFlow Architecture..... | 26 |
| Figure 4-4 : TensorFlow Working..... | 27 |
| Figure 4-5 : Data Structure in JSON Format..... | 29 |
| Figure 4-6 : Objects in training set..... | 29 |
| Figure 4-7 : Instances per Category..... | 30 |
| Figure 4-8 : Comparison Graphs..... | 30 |
| Figure 4-9 : Screenshot 1..... | 32 |
| Figure 4-10 : Screenshot 2..... | 32 |
| Figure 4-11 : Screenshot 3..... | 32 |

| | |
|---|----|
| Figure 4-12 : Test Case 1 output..... | 34 |
| Figure 4-13 : Test Case 2 Output 1..... | 35 |
| Figure 4-14 : Test Case 2 Output 2..... | 35 |

List of Tables

| | |
|-----------------------------------|----|
| Table 1 : Types of Datasets..... | 19 |
| Table 2 : Database Structure..... | 21 |
| Table 3 : Types of Models..... | 27 |
| Table 4 : Test Case 1..... | 33 |
| Table 5: Test Case 2..... | 34 |

Table of Contents

| | |
|---|-----------|
| CHAPTER 1 . INTRODUCTION | 1 |
| 1.1 Overview | 1 |
| 1.2 Background and Motivation | 2 |
| 1.3 Problem Statement and Objectives | 2 |
| 1.4 Scope of the Project | 3 |
| 1.5 Team Organization | 5 |
| 1.6 Report Structure | 5 |
| CHAPTER 2 . REVIEW OF LITERATURE | 7 |
| 2.1 Preliminary Investigation | 7 |
| 2.1.1 Current System | 7 |
| 2.2 Limitations of Current System | 8 |
| 2.3 Requirement Identification and Analysis for Project | 8 |
| 2.3.1 Conclusion | 14 |
| CHAPTER 3 . PROPOSED SYSTEM | 15 |
| 3.1 The Proposal | 15 |
| 3.2 Benefits of the Proposed System..... | 15 |
| 3.3 Block Diagram | 16 |
| 3.4 Feasibility Study | 16 |
| 3.4.1 Technical | 16 |
| 3.4.2 Economical..... | 17 |
| 3.4.3 Operational | 17 |
| 3.5 Design Representation..... | 18 |
| 3.5.1 Data Flow Diagrams | 20 |
| 3.5.2 Database Structure | 21 |
| 3.6 Deployment Requirements | 21 |
| 3.6.1 Hardware | 21 |
| 3.6.2 Software..... | 22 |

| | |
|--|-----------|
| CHAPTER 4 . IMPLEMENTATION | 23 |
| 4.1 Technique Used | 23 |
| 4.1.1 Deep- Learning | 23 |
| 4.1.2 Neural Networks : | 24 |
| 4.2 Tools Used | 25 |
| 4.2.1 OpenCV | 25 |
| 4.2.2 Tensor Flow | 26 |
| 4.2.3 Models | 27 |
| 4.3 Language Used | 31 |
| 4.4 Screenshots | 32 |
| 4.5 Testing | 33 |
| 4.5.1 Strategy Used..... | 33 |
| 4.5.2 Test Case and Analysis | 33 |
| CHAPTER 5 . CONCLUSION..... | 36 |
| 5.1 Conclusion | 36 |
| 5.2 Limitations of the Work | 36 |
| 5.3 Suggestion and Recommendations for Future Work | 37 |
| BIBLIOGRAPHY | 38 |
| PROJECT PLAN..... | 41 |
| GUIDE INTERACTION SHEET | 42 |
| SOURCE CODE..... | 43 |

Chapter 1. Introduction

Introduction

As good street lighting conditions are required to improve the safety and security of people specially in night in rural areas because the road conditions are not always good at all the places which can cause severe accidents and many more problems. As a result, in the absence of adequate light, some of these areas are unsafe for people at night, especially adults, women and children. It is not like that the authorities don't want to resolve the issue but they don't have the proper information, location of the non functional street light and it will be very difficult task to examine all the street light manually So an Automated feedback system is required to solve this issue in which the people can report about the non-functional streetlight so as to resolve the problem.

1.1 Overview

Street light plays an very important role when it comes to road safety and security. Road lighting improves safety for drivers, passengers, and pedestrians on rural roads, where road conditions are sometimes not up to standard. Street lights are installed in many cities and regional highways in Central India. However, in many cases these lights remain inactive due to the latest information about the exact location and unavailability of adequate service support .The software-based automated response system may allow service engineers to know the exact location of traffic lights that are not working in their area of responsibility.

1.2 Background and Motivation

The non-Functioning of street lights is a very serious problem for decades, various possible methods are done to solve this problem but with the passage of time, as technology took its growth to the extreme it becomes simpler to reduce this problem by using software application.

The main motivation behind picking up this topic is the safety on roads at the time of night not only for the vehicles that are being driven on the road but also the people living in the locality and using that road to commute by walking daily.

1.3 Problem Statement and Objectives

Good Street lighting systems in rural areas improve road safety, personal security and raise the standard of living. Street lighting also improves safety for drivers, riders, and pedestrians on rural roads, where the road conditions are sometimes not of the desired standards. Streetlights have been installed in many villages and state highways in Central India. However, many times these lights remain non-functional due to late information about the exact location and unavailability of sufficient service support. Consequently, in the absence of proper lighting, some of these areas become unsafe for people at night especially elders, women, and children. An automated feedback system based on a software may allow the service engineers to know the exact location of non-functional streetlights in their areas of responsibility. A solution may be sought from students through this hackathon regarding an automated feedback system.

Thus, the system implemented has the following objectives :

Objective 1: The goal of this project is to develop a software system that allows the user/people to report the non-functional street lights by themselves which helps in resolving the issue as soon as possible.

Objective 2: The objective is to ensure safety by the means of a software that will directly report to the concerned authority about the issue. Thus the road accidents will be reduced and safety will be improved very efficiently. The people and drivers will feel safe to travel at night. A central database will record all the information about the requests and the issues that got resolved

1.4 Scope of the Project

- This system bridges the gap between the people and the concerned authority and it will also create awareness among the people .
- This system will be implemented in some rural regions first and then will be expanded to other regions all over the state.
- This system will bring a remarkable change in the road safety and security .
- The size of the database may increase very rapidly so this system is made in such a way that it is scalable , reliable, easy to use, fast and efficient.
- The system will reduce road accidents and crime thus less reports registered to the police.
- This system will save time as compared to the manual system which requires going to the government offices to report about the issue and takes so much time to resolve.

1.5 Team Organization

1.6 Report Structure

The project **Automated Feedback System for Non-functional Streetlight** is primarily concerned with the **automation of the whole process of complaining about non functional streetlight** and whole project report is categorized into five chapters.

Chapter 1: Introduction- introduces the background of the problem followed by rationale for the project undertaken. The chapter describes the objectives, scope and applications of the project. Further, the chapter gives the details of team members and their contribution in development of project which is then subsequently ended with report outline.

Chapter 2: Review of Literature- explores the work done in the area of Project undertaken and discusses the limitations of existing system and highlights the issues and challenges of project area. The chapter finally ends up with the requirement identification for present project work based on findings drawn from reviewed literature and end user interactions.

Chapter 3: Proposed System - starts with the project proposal based on requirement identified, followed by benefits of the project. The chapter also illustrate software engineering paradigm used along with different design representation. The chapter also includes block diagram and details of major modules of the project. Chapter also gives insights of different type of feasibility study carried out for the project undertaken. Later it gives details of the different deployment requirements for the developed project.

Chapter 4: Implementation - includes the details of different Technology/ Techniques/ Tools/ Programming Languages used in developing the Project. The chapter also includes the different user interface designed in project along with their functionality. Further it discuss the experiment results along with testing of the project. The chapter ends with evaluation of project on different parameters like accuracy and efficiency.

Chapter 5: Conclusion - Concludes with objective wise analysis of results and limitation of present work which is then followed by suggestions and recommendations for further improvement.

Chapter 2 . Review of Literature

2.1 Preliminary Investigation

2.1.1 Current System

- The manual system one has to report about the non functional street light by going to the government offices where one has to submit an application about the issue which will be given to a government official and then all the applications keeps on getting piled up and no one takes the responsibility to resolve the issue even after so many attempts.

2.2 Limitations of Current System

The limitations of these are as follows :

- It takes long period of time to resolve the issue and people have to stand in long queues to file complaint .
- Even after filling complaint there is no guarantee that the problem will get resolved thus not an efficient way to solve the problem .
- Since it involves employees for manual work there is a meaningless usage of resource thus wasting time and money of people and the organization .
- To complaint about the issue in the manual system lots of documentation have to be done which is a hectic process .

2.3 Requirement Identification and Analysis for Project

2.3.1 Conclusion

The proposed Automated feedback system is a reliable platform for both the common people and the government officials as it provide proper data in an organized manner which ensures the smooth functioning of the department

responsible. It is an application that helps minimize human error and data redundancy issues. It is a fast and efficient method of communication between the common people and government bodies. The proper working of the street lights will avoid accidents and thus increasing the likelihood of saving lives.

Chapter 3 . Proposed System

Proposed System

3.1 The Proposal

This project is to develop a software system that allows the user/person to self-report the inoperative street lamps and helps to solve the problem as soon as

possible. This system is primarily based on the request generated by the user to solve a issue. For automatic feedback, the user needs to register on the platform using our app which will be available in both Hindi and English and then file a complaint about the non-functional street light. While registering the complaint, the user is asked for the image of the dysfunctional street lamp and after the image is approved, he/she has to fill in the location details. This location detail is matched with the address we will get from the user's mobile GPS coordinates to verify the exact location of the complaint. After the image is verified, it will be stored in the complaint database. This stored complaint will then be made visible to the relevant authorities or technicians and to the user who has filled out this complaint. After visiting the application, review all the complaints lists and select the complaint that was given to it. With the help of the coordinates, the technician will reach the exact location of the dysfunctional street light. After successful resolution of the complaint, the user will be notified that the complaint has been successfully resolved. This is how we automate the feedback process.

3.2 Benefits of the Proposed System

The current system had a lot of challenges that are overcome by this system :

- **Economic** : The proposed system is economic as there will be less usage of resource .
- **Man Power** : it requires less man power as compared to the existing system.
- **24 x 7 Availability** : no time boundation to file a complaint as the system is totally online.
- **Saves time**: it will save a lot of time as compared to the manual system.

3.3 Block Diagram

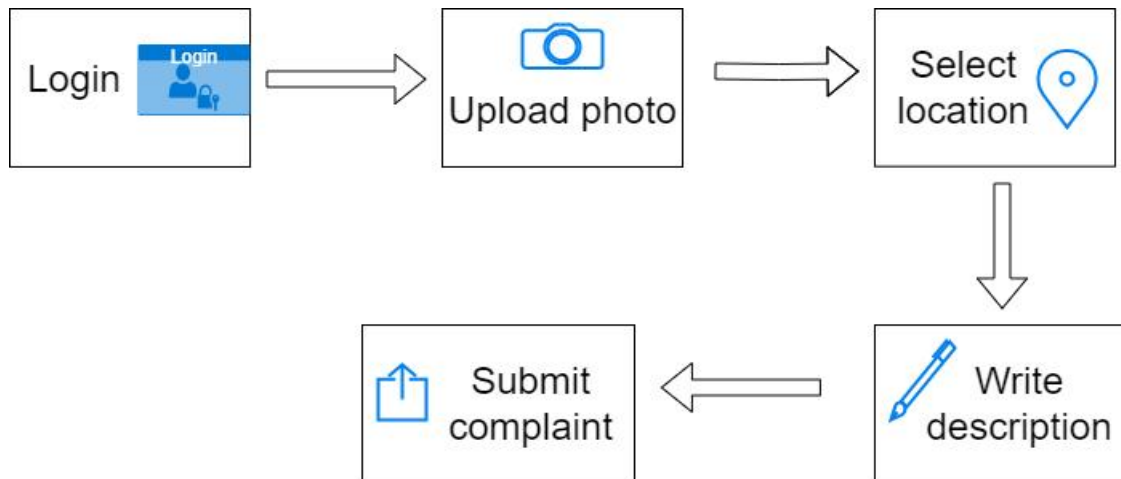


Figure 1

3.4 Feasibility Study

A feasibility study is an analysis of how successfully a system can be implemented, accounting for factors that affect it such as economic, technical and operational factors to determine its potential positive and negative outcomes before investing a considerable amount of time and money into it .

3.4.1 Technical

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA . On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, SQL tutorial provides basic and advanced concepts of SQL. Our SQL tutorial is designed for both beginners and professionals. SQL (Structured Query Language) is used to perform operations on the records stored in the database, such as updating records, inserting records, deleting records, creating and modifying database tables, views, etc. SQL is not a database system, but it is a query language.

SQLite is a C-language library that implements a small, fast, self-contained, high-reliability, full-featured, SQL database engine. SQLite is the most used database engine in the world. SQLite is built into all mobile phones and most computers and comes bundled inside countless other applications that people use every day.

3.4.2 Economical

For this feedback system a mobile with a good camera quality is required to click pictures of the non functional streetlight to that the request doesn't get rejected.

Since the system is completely automated, there is a need of internet connection for it to operate.

This application doesn't require high performance processor to run on hence this application is very economical for the user to use.

3.4.3 Operational

The main motto of our system is to reduce the manual efforts of checking the non functional streetlight by the technicians.

The system does all the task automatically hence making the system operationally feasible.

3.5.1 Data Flow Diagrams

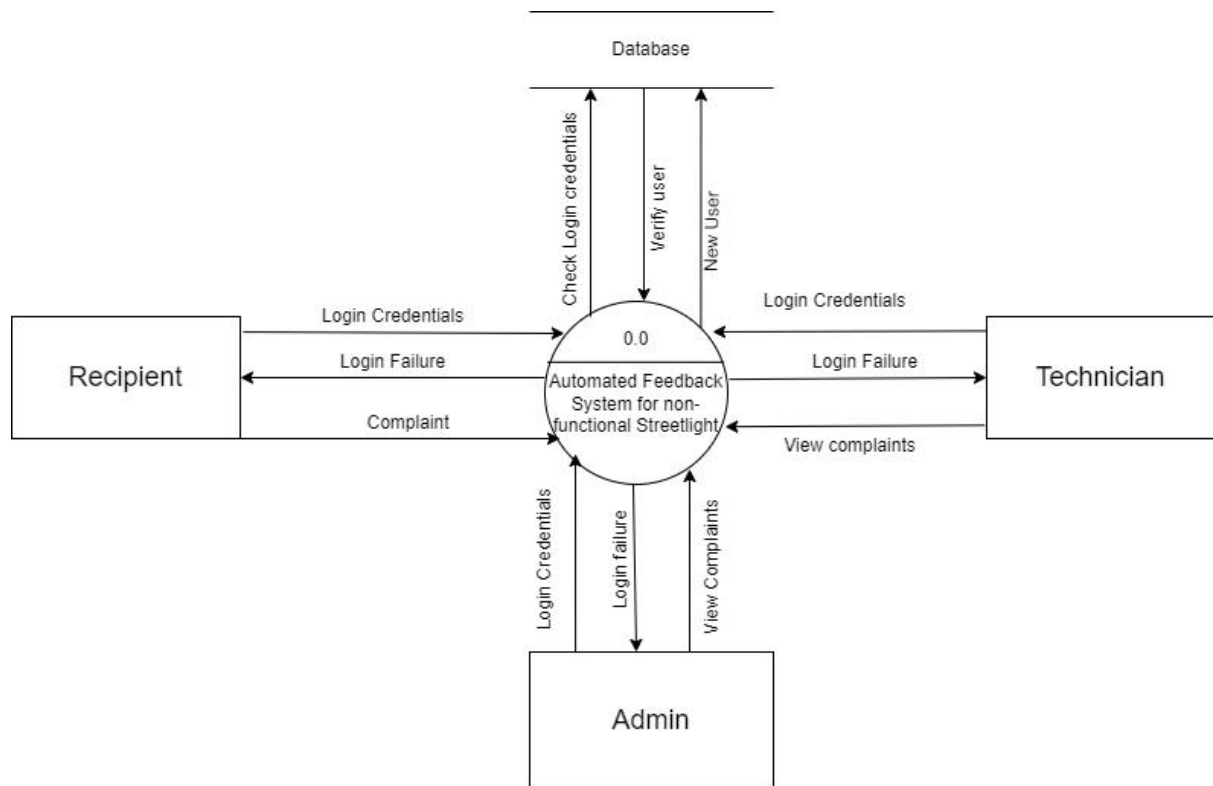


Figure 2. level 0 DFD

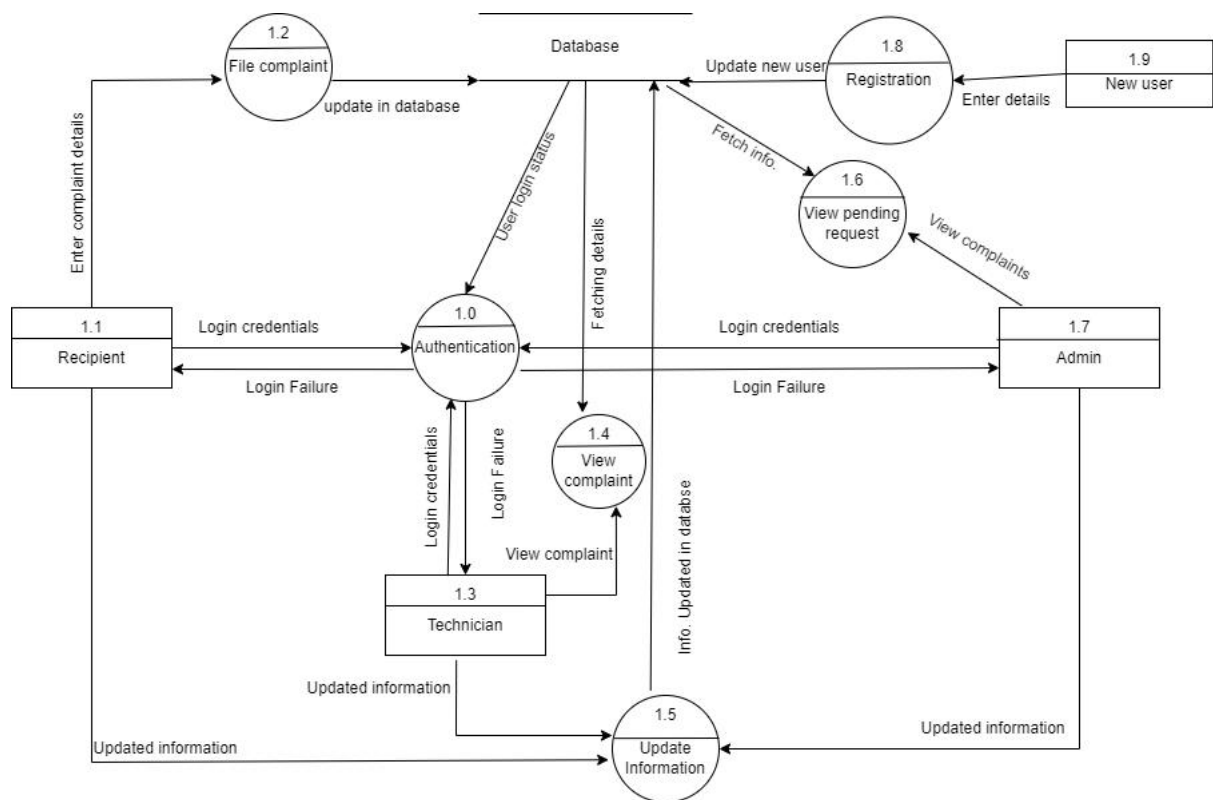


Figure 3. level 1 DFD

Figure 4. level 2 DFD

3.5.2 Activity diagram

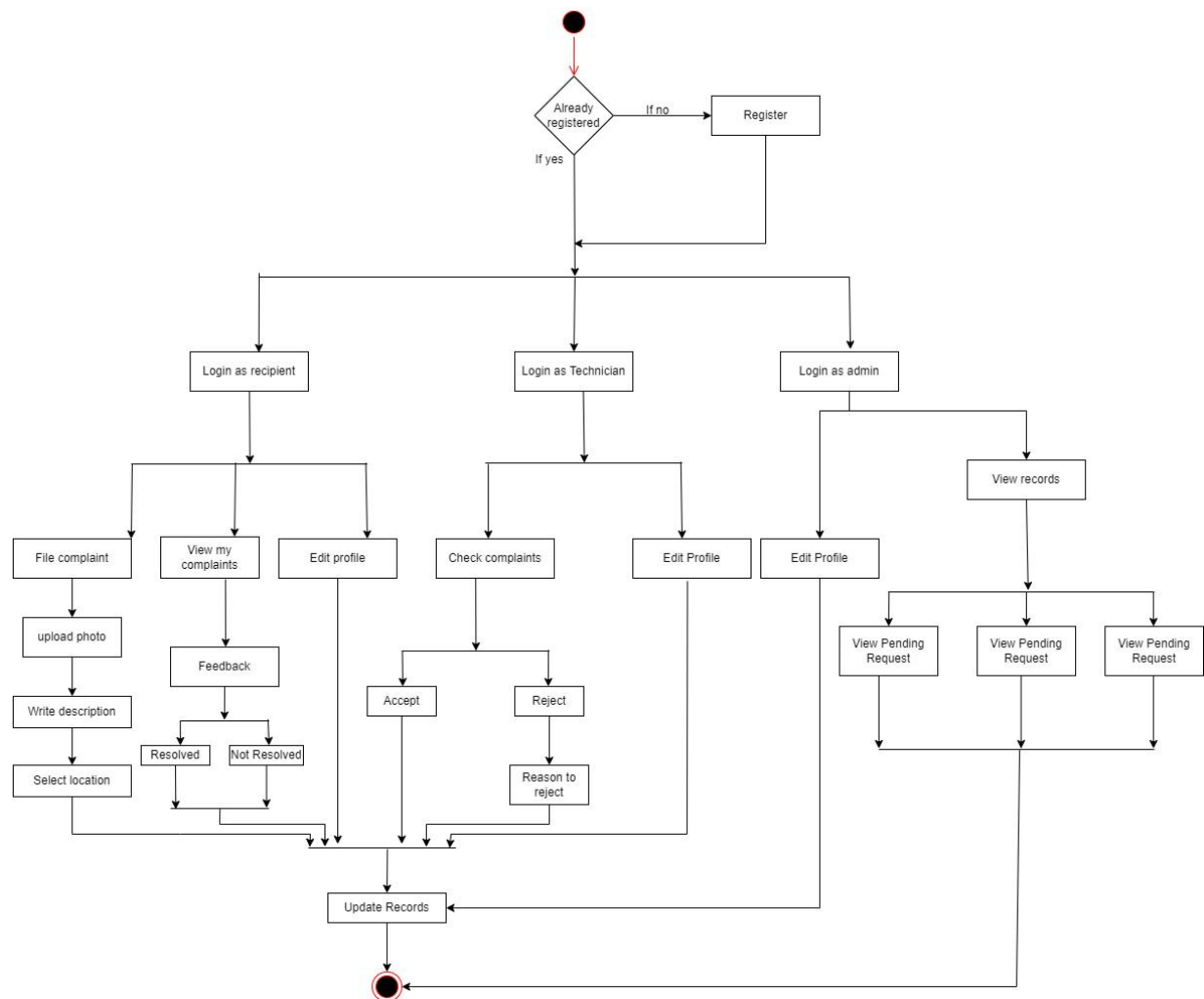


Figure 4. Activity diagram

3.5.4 State transition diagram

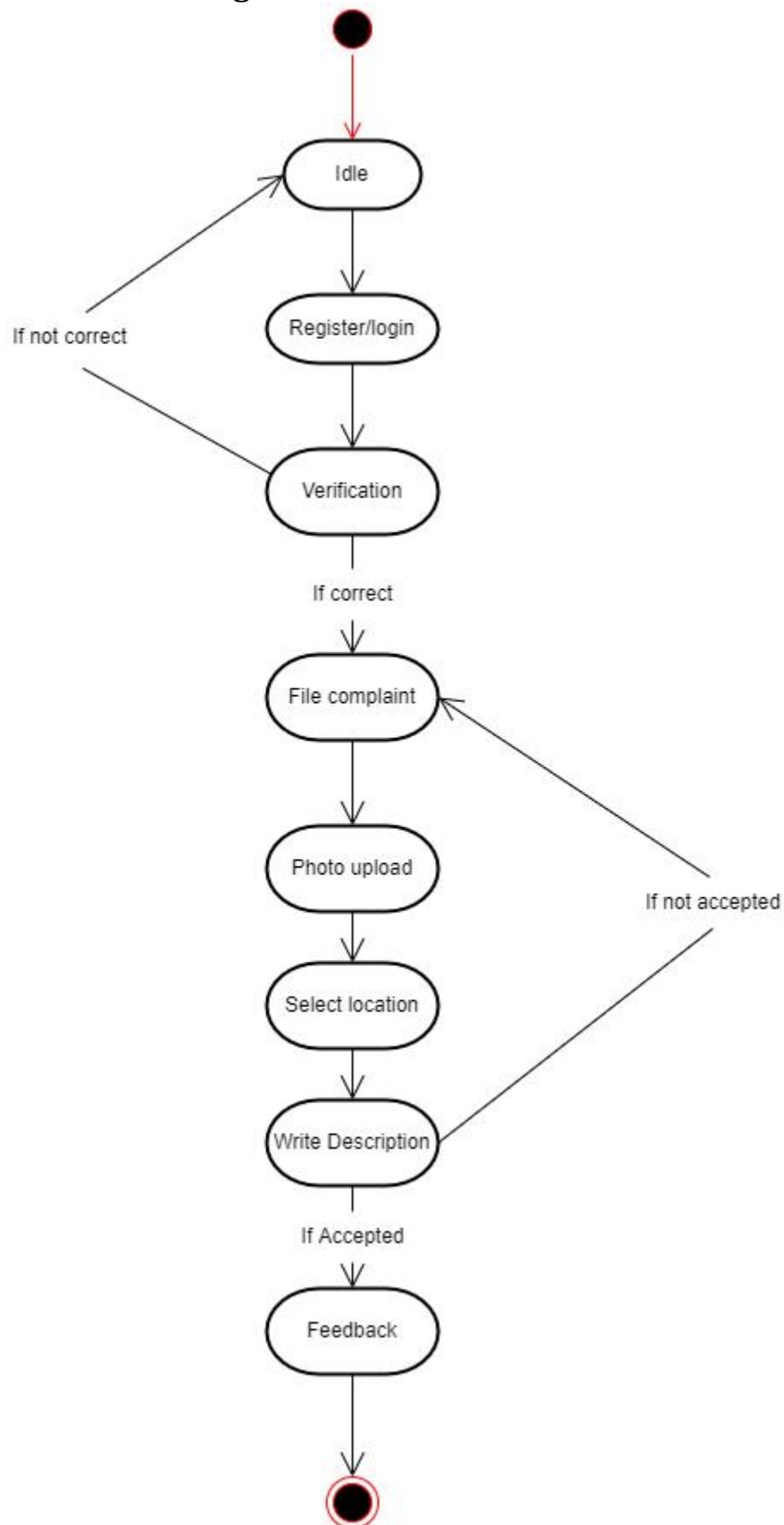


Figure 6. state transition for recipient

3.5.6 Sequence diagram

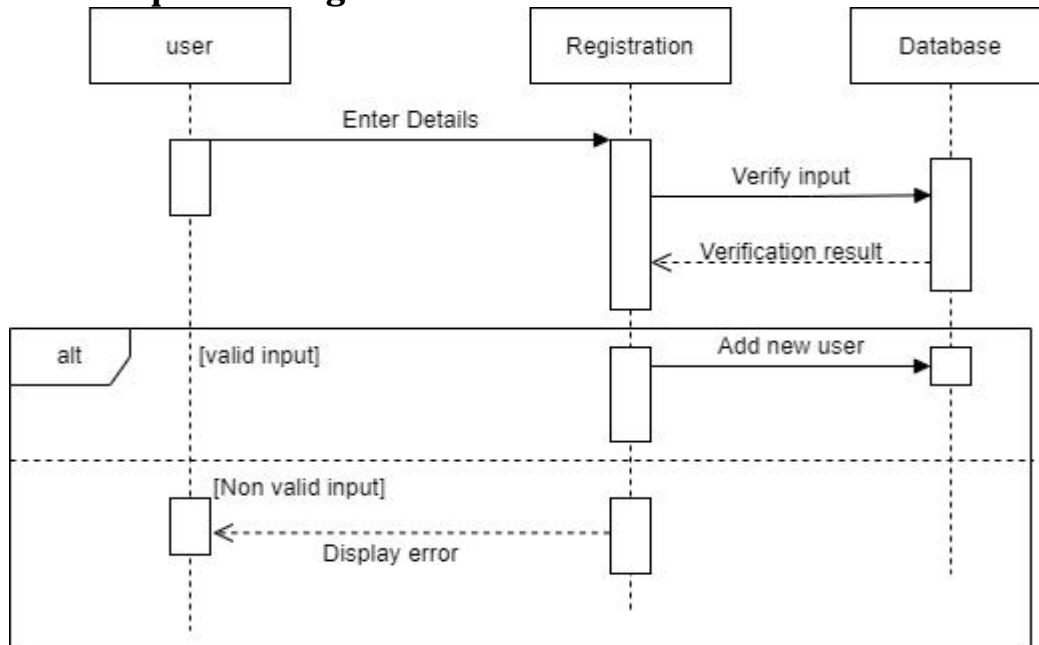


Figure 7.1 Sequence diagram for registration

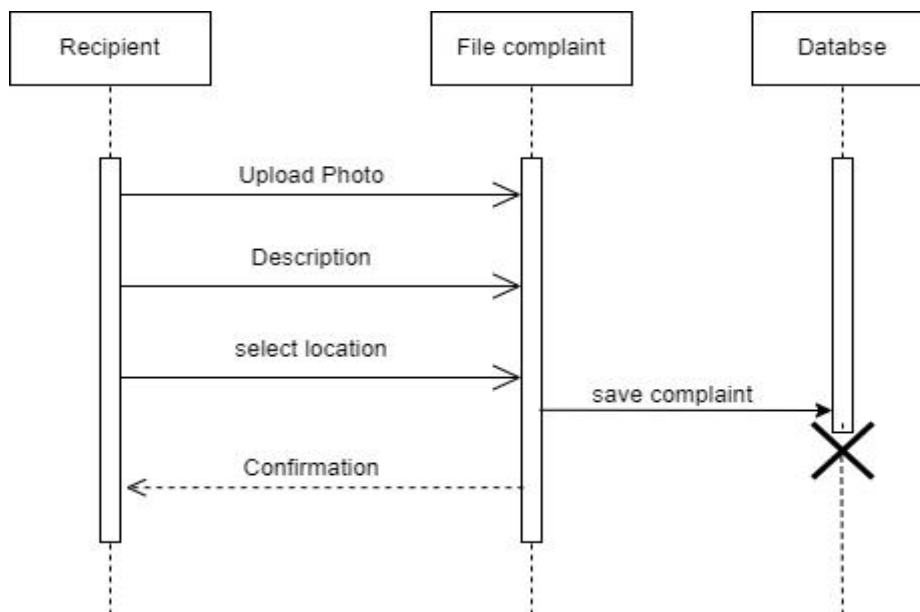


Figure 7.2 Sequence diagram for filing complaint

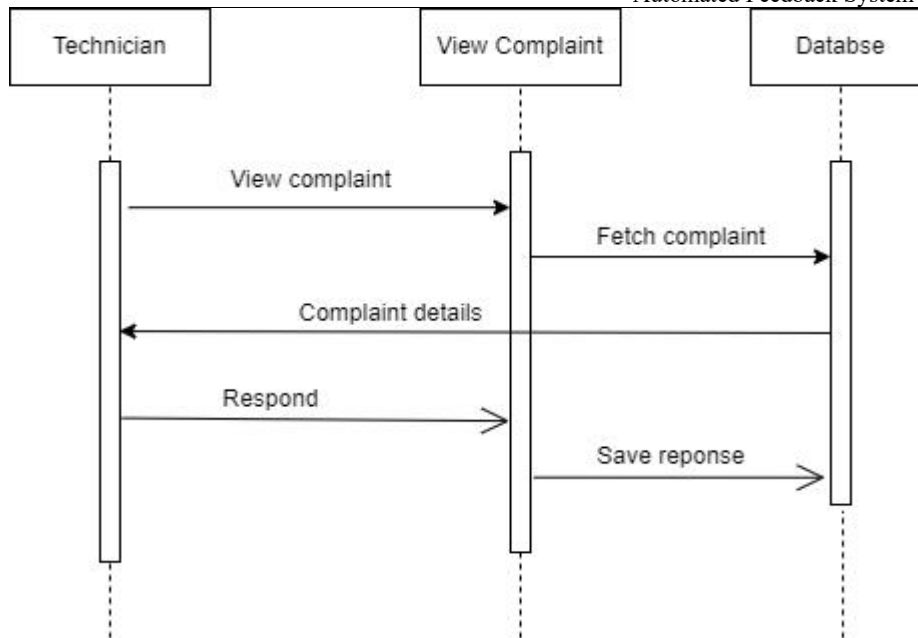


Figure 7.3 Sequence diagram for technician

3.5.7 Use Case Diagram

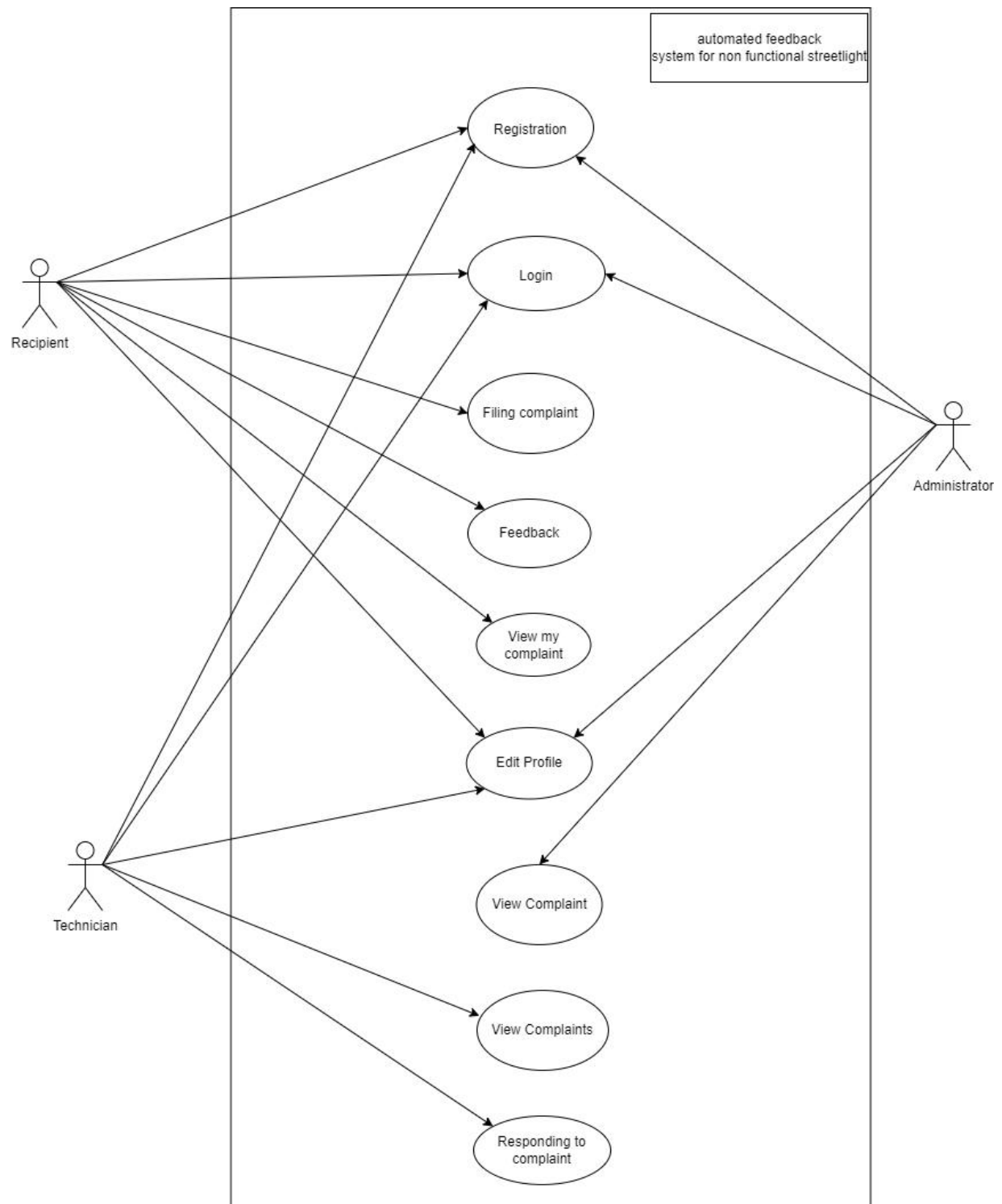


Figure 8. use case diagram

3.5.3 Database Structure

The name of the database created is “MY_DB” which will contain 4 tables in it and is created using SQLite in android studio. The four table are as follows:

- **Recipient information table:** It has 2 columns name and mobile number.
- **Technician information table:** It has 2 columns empId and name.
- **Complaint information table:** it has 2 columns empId and name.
- **Administrator information table:** it has 4 columns photo, location, description, mobile.

3.6 Deployment Requirements

There are various requirements (hardware, software and services) to successfully deploy the system. These are mentioned below :

3.6.1 Hardware

- 64-bit, x86 Processing system
- Windows 8 or later operating system
- High processing computer system without GPU or with GPU(high performance).

3.6.2 Software

- Android studio version 2021.2.1.
- JAVA and its supported extensions and libraries .
- XML latest version for UI/UX designing .
- SQL for database building .

Chapter 4 . Implementation

Implementation

For the problem of counting the number of students and vehicles entering the college campus manually, the system is designed in such a way so as to automate the process by placing a camera at the entrance gate so that students, bikes and cars getting inside the college campus can be identified and counted.

4.1 Technique Used

4.1.1 SQLite

This SQLite tutorial teaches you everything you need to know to start using SQLite effectively. In this tutorial, you will learn SQLite step by step through extensive hands-on practices.

This SQLite tutorial is designed for developers who want to use SQLite as the back-end database or to use SQLite to manage structured data in applications including desktop, web, and mobile apps.

SQLite is an open-source, zero-configuration, self-contained, stand-alone, transaction relational database engine designed to be embedded into an application.

4.2 Tools Used

4.2.1 Android studio

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA . On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, such as:

- A flexible Gradle-based build system
- A fast and feature-rich emulator
- A unified environment where you can develop for all Android devices
- Apply Changes to push code and resource changes to your running app without restarting your app
- Code templates and GitHub integration to help you build common app features and import sample code
- Extensive testing tools and frameworks
- Lint tools to catch performance, usability, version compatibility, and other problems
- C++ and NDK support
- Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine

4.2.2 Draw.io

Designed by Seibert Media, draw.io is proprietary software for making diagrams and charts. The software lets you choose from an automatic layout function, or create a custom layout. They have a large selection of shapes and hundreds of visual elements to make your diagram or chart one-of-a-kind. The drag-and-drop feature makes it simple to create a great looking diagram or chart.

Draw.io has options for storing saved charts in the cloud, on a server, or network storage at a data center, depending on your needs

4.3 Language Used

JAVA

Java is one of the powerful general-purpose programming languages, created in 1995 by Sun Micro systems (now owned by Oracle). Java is Object-Oriented. However, it is not considered as pure object-oriented as it provides support for primitive data types (like int, char, etc). Java syntax is similar to C/C++. But Java does not provide low-level programming functionalities like pointers. Also, Java code is always written in the form of classes and objects. Android heavily relies on the Java programming language all the SDKs required to build for android applications use the standard libraries of Java. If one is coming from a traditional programming background like C, C++, Java is easy to learn. So in this discussion, there is a complete guide to learn Java specifically considering Android App Development.

Object Oriented :

JAVA supports procedure-oriented programming as well as object-oriented programming. In procedure-oriented languages, the program is built around procedures or functions which are nothing but reusable pieces of programs. In object-oriented languages, the program is built around objects which combine data and functionality. JAVA has a very powerful but simple way of doing object-oriented programming, especially, when compared to languages like C++ or Java.

4.5 Testing

Testing is the process of evaluation of a system to detect differences between given input and expected output and also to assess the feature of the system. Testing assesses the quality of the product. It is a process that is done during the development process. .

4.5.1 Strategy Used

Tests can be conducted based on two approaches –

- Functionality testing
- Implementation testing

The testing method used here is Black Box Testing. It is carried out to test functionality of the program. It is also called ‘Behavioral’ testing. The tester in this case, has a set of input values and respective desired results. On providing input, if the output matches with the desired results, the program is tested ‘ok’, and problematic otherwise.

4.5.2 Test Case and Analysis

Chapter 5.Conclusion

Conclusion

5.1 Conclusion

The proposed Automated feedback system is a reliable platform for both the common people and the government officials as it provide proper data in an organized manner which ensures the smooth functioning of the department responsible. It is an application that helps minimize human error and data redundancy issues. It is a fast and efficient method of communication between the common people and government bodies. The proper working of the street lights will avoid accidents and thus increasing the likelihood of saving lives.

The work done manually can now be completely replaced by this automated system and it can reduce all the extra efforts of maintain the records.

5.2 Limitations of the Work

- One of the main limitation of these project is that the application will not be able to work without proper internet connection because the database has to be updated frequently for proper functioning of the system
- Other problem of this system is that old people or some people in the rural area who are not familiar to the modern technology will face some problems in using the application.
- It is also possible that people upload fake complaint which will cause misuse of resources and wastage of time of the officials and technician.

5.3 Suggestion and Recommendations for Future Work

Bibliography

Project Plan

Guide Interaction Sheet

| Date | Discussion | Action Plan |
|------------|---|--|
| 4/01/2018 | Discussed about the title of the Project | Real-time object detection and Recognition was decided as the title. |
| 10/01/2018 | Discussion on the technology to be used for object detection in real-time | Tensorflow , OpenCV and other tools were finalized |
| 14/01/2018 | Discussion of the creation of synopsis of the project | Gathering of information for synopsis creation |
| 17/01/2018 | Suggestions on how to do a literature survey and preliminary investigation on the topic | Many research papers were read , understood and their abstract were to be written. |
| 22/02/2018 | Discussion on the implementation of the project | Using tensorflow and other tools, we decided to implement detection. |
| 15/03/2018 | Discussion on the objective of the project(counting of students at the entrance gate of college in real-time) | Decided to Include the logic of counting students in the program |
| 26/03/2018 | Suggestion for counting the number of vehicles like cars, bikes, buses also at the college entrance | Took steps for adding and modifying the program for counting vehicles also |
| 10/04/2018 | For generation of log files and storing the result, database was advised to be added | Action taken that for each user an entry must be made in the database so that count can be made easy |
| 15/04/2018 | Discussion on project documentation | Decided to write the content and integrate it in the proper format of the report |

Source code

MainActivity.java

```
package com.example.samadhan;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

public class MainActivity extends AppCompatActivity {

    Button btnLogin,btnReg;

    Intent iLogPg,iRegPg;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        btnLogin=findViewById(R.id.btnLogin);

        btnReg=findViewById(R.id.btnReg);

        btnLogin.setOnClickListener(new View.OnClickListener() {

            @Override

            public void onClick(View view) {
```

```

        iLogPg = new
Intent(getApplicationContext(),Login_page.class);

        startActivity(iLogPg);

    }

});

btnReg.setOnClickListener(new View.OnClickListener() {

    @Override

    public void onClick(View view) {

        iRegPg=new
Intent(getApplicationContext(),Register_page.class);

        startActivity(iRegPg);

    }

});

}

}

```

Activity_main.xml

```

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout

xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:app="http://schemas.android.com/apk/res-auto"

    xmlns:tools="http://schemas.android.com/tools"

```

```
android:layout_width="match_parent"
```

```
android:layout_height="match_parent"
```

```
tools:context=".MainActivity">
```

```
<Button
```

```
    android:id="@+id/btnLogin"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_marginTop="256dp"
```

```
    android:fontFamily="sans-serif"
```

```
    android:text="LOGIN"
```

```
    android:textSize="20sp"
```

```
    android:textStyle="bold"
```

```
    app:layout_constraintEnd_toEndOf="parent"
```

```
    app:layout_constraintHorizontal_bias="0.498"
```

```
    app:layout_constraintStart_toStartOf="parent"
```

```
    app:layout_constraintTop_toTopOf="parent" />
```

```
<Button
```

```
    android:id="@+id/btnReg"
```

```
    android:layout_width="wrap_content"
```

```

    android:layout_height="wrap_content"

    android:layout_marginTop="65dp"

    android:height="10dp"

    android:text="REGISTER"

    android:textSize="20sp"

    android:textStyle="bold"

    app:layout_constraintBottom_toBottomOf="parent"

    app:layout_constraintEnd_toEndOf="parent"

    app:layout_constraintHorizontal_bias="0.498"

    app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toBottomOf="@+id/btnLogin"

    app:layout_constraintVertical_bias="0.007" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

Login_page.java

```

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout

xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:app="http://schemas.android.com/apk/res-auto"

    xmlns:tools="http://schemas.android.com/tools"

    android:layout_width="match_parent"

    android:layout_height="match_parent"

```

```
tools:context=".MainActivity">
```

```
<Button
```

```
    android:id="@+id/btnLogin"

    android:layout_width="wrap_content"

    android:layout_height="wrap_content"

    android:layout_marginTop="256dp"

    android:fontFamily="sans-serif"

    android:text="LOGIN"

    android:textSize="20sp"

    android:textStyle="bold"

    app:layout_constraintEnd_toEndOf="parent"

    app:layout_constraintHorizontal_bias="0.498"

    app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toTopOf="parent" />
```

```
<Button
```

```
    android:id="@+id/btnReg"

    android:layout_width="wrap_content"

    android:layout_height="wrap_content"

    android:layout_marginTop="65dp"
```

```
android:height="10dp"

android:text="REGISTER"

android:textSize="20sp"

android:textStyle="bold"

app:layout_constraintBottom_toBottomOf="parent"

app:layout_constraintEnd_toEndOf="parent"

app:layout_constraintHorizontal_bias="0.498"

app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/btnLogin"

app:layout_constraintVertical_bias="0.007" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

Activity_login_page.xml

```
<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout

xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout_width="match_parent"

android:layout_height="match_parent"

tools:context=".Login_page">
```

<Button

```
android:id="@+id/btnLogRec"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="180dp"
android:text="LOGIN AS      RECIPIENT"
android:textSize="20sp"
android:textStyle="bold"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.494"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent" />
```

<Button

```
android:id="@+id/BtnLogTec"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="56dp"
android:layout_marginEnd="6dp"
android:text="LOGIN AS      TECHNICIAN"
android:textSize="20sp"
```

```
android:textStyle="bold"
```

```
app:layout_constraintEnd_toEndOf="parent"
```

```
app:layout_constraintHorizontal_bias="0.498"
```

```
app:layout_constraintStart_toStartOf="parent"
```

```
app:layout_constraintTop_toBottomOf="@+id/btnLogRec" />
```

```
<Button
```

```
android:id="@+id/BtnLogAd"
```

```
android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"
```

```
android:layout_marginTop="56dp"
```

```
android:text="LOGIN AS ADMINISTRATOR"
```

```
android:textSize="20sp"
```

```
android:textStyle="bold"
```

```
app:layout_constraintEnd_toEndOf="parent"
```

```
app:layout_constraintHorizontal_bias="0.472"
```

```
app:layout_constraintStart_toStartOf="parent"
```

```
app:layout_constraintTop_toBottomOf="@+id/BtnLogTec" />
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

Register_page.java

```
package com.example.samadhan;
```



```
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;


public class Register_page extends AppCompatActivity {

    EditText edtRegName,edtRegNo;

    Button btnRegister;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_register_page);

        edtRegName=findViewById(R.id.edtRegName);

        edtRegNo=findViewById(R.id.edtRegNo);

        btnRegister=findViewById(R.id.btnRegister);

    }

}
```

Activity_register_page.xml

```
<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Register_page">

    <EditText
        android:id="@+id/edtRegName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="160dp"
        android:ems="10"
        android:hint="Enter name"
        android:inputType="textPersonName"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.502"
```

```
app:layout_constraintStart_toStartOf="parent"
```

```
app:layout_constraintTop_toTopOf="parent" />
```

```
<EditText
```

```
    android:id="@+id/edtRegNo"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_marginTop="56dp"
```

```
    android:ems="10"
```

```
    android:hint="Enter mobile number"
```

```
    android:inputType="phone"
```

```
    app:layout_constraintEnd_toEndOf="parent"
```

```
    app:layout_constraintHorizontal_bias="0.497"
```

```
    app:layout_constraintStart_toStartOf="parent"
```

```
    app:layout_constraintTop_toBottomOf="@+id/edtRegName" />
```

```
<EditText
```

```
    android:id="@+id/editTextTextPersonName2"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_marginTop="56dp"
```

android:ems="10"

android:inputType="textPersonName"

android:text="password/otp"

app:layout_constraintEnd_toEndOf="parent"

app:layout_constraintHorizontal_bias="0.502"

app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/edtRegNo" />

<Button

android:id="@+id/btnRegister"

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:layout_marginTop="60dp"

android:text="REGISTER"

android:textSize="20sp"

android:textStyle="bold"

app:layout_constraintEnd_toEndOf="parent"

app:layout_constraintHorizontal_bias="0.498"

app:layout_constraintStart_toStartOf="parent"

```
app:layout_constraintTop_toBottomOf="@+id/editTextTextPersonName2" />
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

Recipient_login_page.java

```
package com.example.samadhan;
```

```
import androidx.appcompat.app.AppCompatActivity;
```

```
import android.content.Intent;
```

```
import android.os.Bundle;
```

```
import android.view.View;
```

```
import android.widget.Button;
```

```
import android.widget.EditText;
```

```
public class Recipient_Login_Page extends AppCompatActivity {
```

```
    Button btnRecipientLog;
```

```
    EditText edtLogNo,edtLogOtp;
```

```
    Intent iRecHome;
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```

super.onCreate(savedInstanceState);

setContentView(R.layout.activity_recipient_login_page);

btnRecipientLog=findViewById(R.id.btnRecipientLog);

edtLogNo=findViewById(R.id.edtLogNo);

edtLogOtp=findViewById(R.id.edtLogOtp);

btnRecipientLog.setOnClickListener(new
View.OnClickListener() {

    @Override

    public void onClick(View view) {

        iRecHome=new
Intent(getApplicationContext(),Recipient_Home_Page.class);

        startActivity(iRecHome);

    }

});

}

}

```

Activity_recipient_login_page.xml

```

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout

xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:app="http://schemas.android.com/apk/res-auto"

```

```
xmlns:tools="http://schemas.android.com/tools"
```

```
android:layout_width="match_parent"
```

```
android:layout_height="match_parent"
```

```
tools:context=".Recipient_Login_Page">
```

```
<EditText
```

```
    android:id="@+id/edtLogOtp"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_marginTop="80dp"
```

```
    android:ems="10"
```

```
    android:hint="Enter OTP"
```

```
    android:inputType="textPassword"
```

```
    app:layout_constraintEnd_toEndOf="parent"
```

```
    app:layout_constraintHorizontal_bias="0.497"
```

```
    app:layout_constraintStart_toStartOf="parent"
```

```
    app:layout_constraintTop_toBottomOf="@+id/edtLogNo" />
```

```
<Button
```

```
    android:id="@+id/btnRecipientLog"
```

```
    android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"

android:layout_marginTop="80dp"

android:text="LOGIN"

android:textSize="20sp"

android:textStyle="bold"

app:layout_constraintEnd_toEndOf="parent"

app:layout_constraintHorizontal_bias="0.498"

app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/edtLogOtp" />
```

<EditText

```
android:id="@+id/edtLogNo"

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:layout_marginTop="200dp"

android:ems="10"

android:hint="Enter Mobile Number"

android:inputType="number"

app:layout_constraintEnd_toEndOf="parent"

app:layout_constraintHorizontal_bias="0.497"

app:layout_constraintStart_toStartOf="parent"
```



```
app:layout_constraintTop_toTopOf="parent" />  
</androidx.constraintlayout.widget.ConstraintLayout>
```

Recipient_home_page.java

```
package com.example.samadhan;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
  
public class Recipient_Home_Page extends AppCompatActivity {  
    Button btnUpCom;  
    Intent iComPg;  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_recipient_home_page);  
        btnUpCom=findViewById(R.id.btnUpCom);  
        btnUpCom.setOnClickListener(new View.OnClickListener() {
```

@Override

```

    public void onClick(View view) {

        iComPg=new
Intent(getApplicationContext(),Complaint_Upload_Page.class);

        startActivity(iComPg);

    }

});

}

}

```

Activity_recipient_home_page.xml

```

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout

xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:app="http://schemas.android.com/apk/res-auto"

    xmlns:tools="http://schemas.android.com/tools"

    android:layout_width="match_parent"

    android:layout_height="match_parent"

    tools:context=".Recipient_Home_Page">

    <Button

        android:id="@+id/btnUpCom"

```

```
        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:layout_marginTop="240dp"

        android:text="UPLOAD A          COMPLAINT"

        android:textSize="20sp"

        android:textStyle="bold"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintHorizontal_bias="0.498"

        app:layout_constraintStart_toStartOf="parent"

        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Complaint_upload_page.java

```
package com.example.samadhan;

import androidx.annotation.Nullable;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.graphics.Bitmap;

import android.os.Bundle;

import android.provider.MediaStore;
```

```
import android.view.View;

import android.widget.ImageView;


public class Complaint_Upload_Page extends AppCompatActivity {

    ImageView imgUpload;

    Intent iImage;

    private final int CAMERA_REQ_CODE=100;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_complaint_upload_page);

        imgUpload=findViewById(R.id.imgUpload);

        imgUpload.setOnClickListener(new View.OnClickListener() {

            @Override

            public void onClick(View view) {

                iImage=new

                Intent(MediaStore.ACTION_IMAGE_CAPTURE);

                startActivityForResult(iImage,CAMERA_REQ_CODE);

            }

        });

    }
```

@Override

```
protected void onActivityResult(int requestCode, int resultCode,
@Nullable Intent data) {

    super.onActivityResult(requestCode, resultCode, data);

    if(resultCode==RESULT_OK)

    {

        if(requestCode==CAMERA_REQ_CODE)

        {

            Bitmap img = (Bitmap)data.getExtras().get("data");

            imgUpload.setImageBitmap(img);

        }

    }

}
```

Activity_complaint_home_page.xml

```
<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout

xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"
```

```
xmlns:tools="http://schemas.android.com/tools"
```

```
android:layout_width="match_parent"
```

```
android:layout_height="match_parent"
```

```
tools:context=".Complaint_Upload_Page">
```

```
<ScrollView
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
    android:layout_marginEnd="4dp"
```

```
    app:layout_constraintBottom_toBottomOf="parent"
```

```
    app:layout_constraintEnd_toEndOf="parent"
```

```
    app:layout_constraintTop_toTopOf="parent"
```

```
    app:layout_constraintVertical_bias="0.0">
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:orientation="vertical">
```

```
<ImageView
```

```
    android:id="@+id/imgUpload"
```

```
android:layout_width="400dp"
```

```
android:layout_height="400dp"
```

```
android:contentDescription="TODO"
```

```
android:foregroundGravity="center_horizontal"
```

```
android:scaleType="fitXY"
```

```
app:srcCompat="@drawable/click_here_to_upload_photo_min" />
```

```
<EditText
```

```
    android:id="@+id/edtDesc"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="150dp"
```

```
    android:ems="10"
```

```
    android:hint="Enter description"
```

```
    android:inputType="textPersonName"
```

```
    android:textAlignment="center"
```

```
    android:textSize="24sp"
```

```
    android:textStyle="bold" />
```

```
<Button
```

```
    android:id="@+id/btnSubCom"
```

```
android:layout_width="match_parent"
```

```
android:layout_height="wrap_content"
```

```
android:text="SUBMIT COMPLAINT"
```

```
android:textSize="20sp"
```

```
android:textStyle="bold" />
```

```
</LinearLayout>
```

```
</ScrollView>
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```