FixIT

By

CSM Muhammad Zohaib Nazir

GC Muneeb Hassan Khan

GC Arslan Arshad

Submitted to the Faculty of Electrical Engineering Department
National University of Sciences and Technology, Islamabad
in partial fulfillment for the requirements of a B.E Degree in
Computer Software Engineering
May 2017

CERTIFICATE OF APPROVAL

This is to officially state that the thesis work contained in this report **"FixIT"**

Is carried out by:

Zohaib Nazir, Muneeb Hassan and Arslan Arshad

under my supervision and that in my judgment, it is fully ample, in scope andexcellence, for the degree of Bachelor of Computer Software Engineering from National University of Sciences and Technology (NUST).

Approved By:

Signature: ________________________

Supervisor: ***Lec Mobeena Shahzad***

MCS, Rawalpindi

ABSTRACT

Since mobile phones have become an intrinsic part of our daily lives, why not use them to save our time and effort in a new way. The traditional ways of reporting a problem to a concerned Govt. authority includes a visit to their offices, long waits in long queues and still you might return back unheard. FixIT shall enable the residents of a particular region to be able to report their problems to the server. The user of the app will be able to take the photo of the problems that he sees around in the town. These problems will be categorized in 3 different categories (i.e. Municipal, Police & Terrorism). The user will be registered first on the server. The photo will be sent to the server with GPS coordinates of Mobile Phone and a description about the photo. Then the problems can be shared on various social apps for further discussion via server so that only registered users can share their views and discuss. Mind here, that the user will request the server for the sharing of the problem on social media and it can't be done directly through app.

FixIT will be an Android based Application that shall provide a platform for users to send their queries to competent authority. People can also report about different problems they encounter in the society. Furthermore, the problems can be shared on different social media platforms only via requesting the server. It provides people to discuss their problems with the authorities which is the main problem nowadays. To resolve the communication gap between government and the people this app is being developed.

DECLARATION OF ORIGINALITY

We hereby declare that the work contained in this report and the intellectual content of this report are the product of our work. This thesis report has not been formerly published in any structure nor does it include any verbatim of the published resources which could be treated as violation of the international copyright decree. We also affirm that we do recognize the terms ‘plagiarism’ and ‘copyright’ and that in case of any copyright infringement or plagiarism established in this thesis, we will be held fully accountable of the consequences of any such violation.

DEDICATED WITH LOVE AND RESPECT TO MY PARENTS, TEACHERS AND FRIENDS

ACKNOWLEDGEMENTS

All glory goes to Almighty ALLAH, the most merciful,Who led us to this extent. May all glory, honorand Adoration be unto Thy Name.

Our special thanks go to our supervisor **Lec Mobeena Shahzad** for her guide line, in completing this project successfully. Wewould also like topay our thankfulness to the people from MCS, especially Dr Sherri, Dr. Hammad, Dr. Naima Altaf and A/P Bilal Rauf.

A deep gratitude towards **Dr. Adnan Ahmed Khan**(Head of Computer Software Department) for his guidance and facilitation for the Project.

At last, we are most obliged to our Parents,their support contributed immensely to the success of this project.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

Chapter 1

1 Introduction

FixIT is an application proposed as providing a platform to the residents of a particular area to be able to report their issues to the competent authorities in an efficient way, saving both their effort and time.

The Govt. established agencies do carry heavy responsibilities to facilitate the citizens and as a matter of fact it sometimes faces the problem of lack of manpower for handling the issues. Since, there are few employees against an enormous population, the optimum service and resolution of all the problems in time, cannot be guaranteed. Also the mundane means of handling all these complaints in a systematic way is sometimes troubling. As of now, there are a few websites but that are not supported by all the departments that are our covered in our App, and currently there is no automated system for any response or discussion on the complaint.

As per the ordinary customs, a citizen has either to directly walk in to the concerned department or make a phone call to lodge his/her complaint. The concerned staff then visits the site, after being informed about the complaint, makes an assessment of the work and comes back to seek the required manpower, material and equipment for resolving the complaint. This is so time taking. These issues mostly comprise of the disposal of solid waste, drainage of stagnant water, fixing up of worn-out street–lights, and repair of some broken piece of road, theft recovery, counter terrorism, rescue services, anti-bomb squad and unusual happenings etc.

It therefore urges the use of modern and systematic technology to minimize the effects this disproportion as of now, by building an App to enable the customers to register their complaints on the server where their complaints shall be taken care of, efficiently and promptly.

1.1 Intended Audience and Reading Suggestions

The Software Requirements Specification (SRS) document is meant for all the stake holders.

- **Project Supervisor:** It will help to supervise the project and guide the team in a better way.

- **Development Team:** It will help the developer to develop the product and to trace back the functional requirements.
- **Testing Team:** It will help the testers to understand the constraints.
- **Complainers (Users):** The potential stakeholders of the system, who are interested in lodging a complaint or discussion using there Android Phone.
- **UG Project Evaluation Team:** Evaluation committee which will evaluate the progress of UG Projects.
- **Staff:** The intended audience is both higher management and local officials of the different departments, who will update the complaint-status after its resolution.

1.2 Motivation

A group of young volunteers in Karachi have formed a mini organization FixIT. These people are doing a job that is solely the responsibility of the Government (fixing manholes, roads, plantation, cleanliness etc). This gave rise to a perception that the current methods of reporting such issues to the authorities are very inefficient, and there's a dire need to counter this problem. Why not develop a mobile application that can serve well both the people and the authorities.Project Scope

FixIT will be used by all the citizens to lodge their complaints to the Government departments by augmenting their opinion with the photographs of the site. It requires front–end design and implementation for Android smart phones, tablets; and synchronizing the data collected with cloud servers at the back end. The complaints, along with the photographs, will be marked to the concerned supervisor/staff as well as the overall admin. The staff will carry out necessary calculation of the condition and will resolve the complaint by taking along the required manpower, material and equipment to the site. The admin will also be able to monitor the progress of each complaint registered, view comments of other users, according to its status.

1.3 Project Vision

Users, who wish to complaint about the problems and the staff who are responsible for resolution of the complaints can keep track of the problems through FixIT. The FIXIT is an android-based application that offers the users with interface to lodge complaints and discussions, and while allowing the staff to get to know about it and reply to each complaint. FIXITisthe creation that is mainly categorized as a system for handling complaints and discussions, and is intended to assist both the complainers and the staff in easy and efficient reporting of the problems. Complainersdo not have to lodge the complaints of the problems in the old mundane ways anymore. In the existing practice of physical visit to the concerned department or make telephone call to the Office. By using FIXIT, user will be able to save time by rapid complaint-lodging/discussing..

1.4 Project Objective

1.4.1 Primary Objectives:

1. Mobile platform for lodging complaints concerned to different departments.
2. Narrow down the gap between citizens &authorities.
3. Minimized time & effort to lodge the complaint.
4. Share the complaints on social media.

1.4.2 Academic objectives:

1. Development of an android based application.
2. Use of cloud based service as back end support for data storage and computation.
3. Plotting of the complaints against concerned staff, based on the area of responsibility and type of complaint.
4. Introducing a sharing mechanism to enable the complainantto share the complaints on social media to create a sense of responsibility.

1.4.3 Application / End – goal objectives:

1. To facilitate the citizens to lodge their complaints efficiently and swiftly.
2. To enable the concerned staff to carry out the assessment of the scene and make needed estimates for approval.
3. To save the precious time and efforts by minimizing the need for pre-work survey.
4. To enable the staff to track the progress of complaints being registered.

1.5 Deliverables

1. Complete working project

2. Android application
3. Documentation
4. Video Tutorials

Chapter 2

2 Literature Review

2.1 Introduction

Conventional procedure of lodging complaints about civic issues ways back many years and different procedures and methods were followed in different times. FixIT will be an advancement of existing setup of complaint lodging, saving time and effort. The main intention of the project is to set up an ITsetup of Complaint Lodging handled by government departments and different towns/housing schemes. FixITwill bridge the gap between the complainantsand the authorities responsible. The product will enable the complainers to lodge their complaints using their smart phone and enter their location automatically or manually. The IT setup will assign the complaints resolution task to the concerned staff person according to the type of problem.

2.2 Problem Domain

As per the existing scenario, a citizen has either to walk in to office of the authorities (say in case of an issue with electricity, one has to visit the office of WAPDA)or make a phone call in order to register a complaint. The concerned staff visits the site, after being informed about the complaint, and then makes estimation of work and comes back to seek the required manpower, material and equipment for its solution–

This process incurs many problems, like:

1. Inability of the complainant to convey all the desired details to the staff (especially when he tries to register his complaint on phone call).
2. Inability of the staff to understand the exact nature of the job to be done, unless he visits the site.
3. Takes too much time and effort.
4. Inability of authorities to get the problem solved until social media hype is created.

It urges the use of modern technology to minimize the effects of this existing disparity by making an android–based application to enable the citizens to register their complaints to the concerned authorities for solving their problems swiftly, promptly and efficiently.

2.3 Related Work

Keeping in line with modern technological advancements, few of metropolitan organizations have started taking a step in automating their complaint management system by introducing android smartphone applications. “My Council” is one of such applications that helps the citizens of UK to immediately report a range of local issues to the local authority responsible. KPK government also launched and android application for the purpose of solving all problems remotely but most of them were unresolved and the app was unresponsive.

Project Objectives that distinguishes our solution from the others:

1. “FixIT” becomes the pioneer in Pakistan to provide all departments a proper complaint management system without interference with each other.
2. Our solution has the potential to be deployed in all the departments and government organizations wanting giving their services.
3. It will also be an important milestone for further broadening of the scope of facilities in latter versions, from mere complaint registration to obtaining birth/death certificates, paying bills and analysis on maps.
4. To bridge the gap between the complainant and the government authorities.
5. To develop a centralized platform for lodging complaints in different categories pertaining to different Govt. departments.
6. To minimize the time & effort to lodge complaint and to seek help of people in the departments.

To set up the location based system enabling the cantonment board employees to know the complaint area location along with picture.

2.4 Shortcomings/issues

1. Internet connection is essentialfor the working of FixITon android phone.

2. The application is for android based smart phones only.

3. Location accuracy depends upon the quality of the user’s mobile set.

4. For earlier versions, the contents of the application will be in English language only.

5. The server will not be available in case of maintenance and testing issues. No backup server configuration is provided.

2.5 Proposed Project

Every day, thousands of citizens have to face scores of problems related to basic civic facilities in their localities. These widely range from jammed sewers, worn roads, flooding and defective plumbing to dilapidated buildings with worn rooftops, security issues and so on.

The bane of the problem are the inefficient methods of reporting aforementioned problems to the concerned authorities. Visiting the offices or making phone calls, either go unresponsive most of the times or just take so much time that they need to be replaced by something worth saving time and effort. This is where the FixIT comes in to play.

FixITaims to eliminate the immense communication gap between the citizens facing problems and the authorities. The designs of the project will make it difficult for the authorities to hide institutional mismanagement and as well allow for greater transparency in the workings of these authorities. If utilized properly, the system could lead to an extremely positive change and help in the advancement of the standard of living in the populations that choose to employ and embrace the system. Furthermore, FixITwill be a new, standalone product.

2.6 Deliverables

2.6.1 Software Requirement Specification (SRS)

The purpose of the document is to present a detailed description of the FixIT. It explains the purpose and features of the system, the interfaces of the system, what the system will do, its processes and workings,also the constraints under which it must function and how the system will respond to external stimuli. This document is planned for both the developers and the stakeholders of the system. It shall explain how the system will principallyaidconcerned authorities to team up and cooperate with each other.

2.6.2 Software Architecture Document

In this document the overall architecture of the system is discoursed, also including the introduction of various components and subsystems. It is chieflybraced by system Architecture diagram which depicts an insider's viewpoint of the system by unfolding the high level software components that execute the major functions to make the system operational.

2.6.3 Software Design document

The Software design document takes into account all our functional requirements and shows how they interact with each other conceptually. The low level design also shows as to how actually we've been implementing how we are going to implement all of these requirements.

2.6.4 Implementation code Document

The implementation code document provides details about the pseudo code for the application and project prototype.

2.6.5 Software Testing Document

This document has testing modules in which there are certain test cases which depicts the correctness and accuracy of the project.

2.6.6 Final Project Report

This is the thesis report which compiles all the previous and current working for the project. Thesis report provides the whole summary for the project and also give details about each and every aspect of the project starting from introduction of the project, literature review, requirements leading to design discussions then testing and lastly future work and conclusion.

2.6.7 User Manual

User Manual gives details about the use of the end product. It contains details as how to use the product. Its functionalities and details of every aspect as how that works and how to use it. User Manual is for users to get to know the product.

2.7 Technological Requirements

FixIT requires following software and hardware requirements.

2.7.1 Software Interfaces

1. FixIT should be able to run on any version of the following Web browsers: Microsoft Internet Explorer, Mozilla Firefox, Opera and Google Chrome.
2. Primary Operating System supported by FIXIT Interface will be Windows 8.
3. FixIT should be work with MySQL database management system.
4. The app will require access to the device GPS, and request permission for location tracking via the Android Operating System
5. FixIT app should be able to run on all android devices with basic hardware requirements fulfilled that run Android OS 5.0 or above.
6. The app will require access to the device camera and request permission to access gallery or take pictures via the camera through Android Operating System.

2.7.2 Hardware Interfaces

2.7.2.1 Computer System

1. System shall have keyboard input.
2. System shall have mouse input.
3. System shall have a monitor.

4. System shall have a working internet connection.

2.7.2.2 Mobile Device

1. Android Device (Smartphone or Tablet) running Android 5.0 or later, color display.
2. Touch Screen with haptic feedback on key presses (Android Keyboard).
3. Device Camera.
4. Global Positioning System (optional).

2.7.2.3 Web and Database Server

1. To process requests and retrieve/store data.

2.7.3 Communications Interfaces

1. A connection shall be established between the System and the web services that we will create.
2. To access the complaints data from database, Grocery CRUD be used.
3. Communication between the Web Interface and the Server will be through HTTP over a Web browser.
4. Communication between the Android application and the Server will be through HTTP as well.

2.7.4 Programming Interface

Programming interfaces for project are:

1. Android Studio
2. PHP Storm
3. Visual Studio

Chapter 3

3 Overall Description

3.1 Product Perspective

Daily, thousands of citizens have to face scores of problems related to basic civic facilities in their localities. These widely range from jammed sewers, worn roads, flooding and defective plumbing to dilapidated buildings with worn rooftops, security issues and so on.

The bane of the problem are the inefficient methods of reporting aforementioned problems to the concerned authorities. Visiting the offices or making phone calls, both go unresponsive many times and just take so much time that they need to be replaced by something worth saving time and effort. This is where the FixIT comes in to play.

FixIT aims to eliminate the immense communication gap between the citizens facing problems and the authorities. The designs of the project will make it difficult for the authorities to hide institutional mismanagement and as well allow for greater transparency in the workings of these authorities. If utilized properly, the system could lead to a very positive change and help in the advancement of the standard of living in the populations that choose to employ and embrace the system. We will follow the client server architecture technique as mentioned in the diagram below:

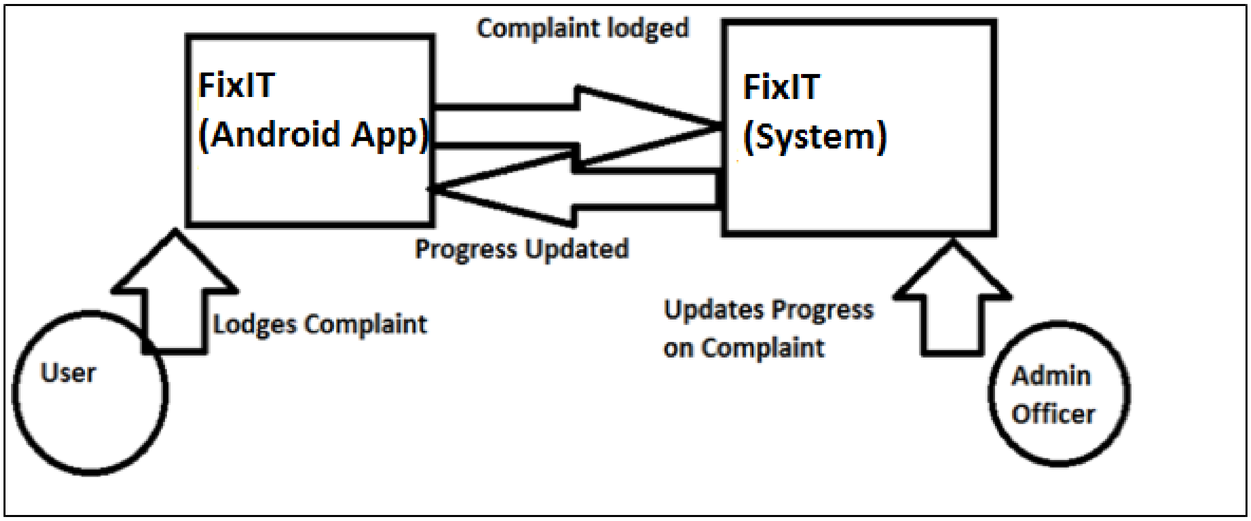


Figure 1 Abstract Diagram

3.2 Product Functions

Following are the key functions of the Town Management System (FIXIT):

3.2.1 User Profiles

Every user will register with FIXIT.

3.2.2 Complaint-lodging along with snapshot

The system will support the complaint-lodging process by allowing the user to add snapshot of the complained-area (along with complaint-description).

3.2.3 Task-assignment based on Complaint-category

The system will allow users to select category of complaint e.g. municipal, health, security etc. to enable the system to direct the task to the related staff.

3.2.4 Complaint-location identification

The system will get the location of the complaint-area, using Android device's Global Positioning System. In case of unavailability of GPS or otherwise too, the user can manually add the location in the complaint-location by inputting LAT-LONG coordinates.

3.2.5 Check and Balance on the progress

This feature will allow to view the status of the solved complaints alongside lodged complaints.

3.2.6 Login/Access Right

FIXIT will allow users to login, based on their roles. Rights of managers and users will not be the same.

3.2.7 Record maintenance and Report generation

This will allow the Administrator to maintain records of complaints registered and also of services performed by different staff. Different users will have different access level to database repository. The database will be designed in a way that protects data integrity and discretion.

3.2.8 Complaint-lodging medium

FIXIT will provide support for complaints to be sent by using internet. Users will be able to search for their complaint-status. The unsolved complaints will be marked by RED color and the solved by GREEN.

3.2.9 Comment on Issues

FIXIT will allow all the users to comment on the problems. A user can edit/delete his/her own comment.

3.3 User Classes and Characteristics

The software has three types of users, admin, managers and users. These three types of users have different access level to the system and its data and can perform functions assigned to their respective roles.

- **Administrator: -**The administration will use the web based interface of FIXIT in order to view the complaints of the public user and their progress logs with rights to modify anything (comments, users, complaints, statuses etc.) included in FIXIT.
- **Manager: -** The Manager can view the complaints which have been assigned to him/her and can also update the status of the resolved-complaints.
- **Public User: -** This will form the majority of the users of the system and comprises a wide multiplicity of people. This category of user will be able to lodge complaints and comment on different complaints, view the progress log for the complaint but will not be able to make any alterations to the application status and the progress log. Public users can view the news and updates of the disaster. They don’t require credentials to access different functionalities of system. They can only see the information that is publicly available to them.

3.4 Operating Environment

OE-1: FIXIT back-end utility i.e. online server that can be bought and database can be maintained there. All the data will be accessed on the cloud server and data manipulation can be done on the same server. Admin changes are done directly on the cloud without notifying any users.

OE-2: The Web based system of FIXIT shall run on the computer system with following specifications:

- Pentium 4 or Higher CPU
- At least 512MB of RAM
- At least 1 GB of free disk space
- Windows XP or later operating system
- Chrome, Firefox or Internet Explorer
- Color monitor and a working internet connection.

OE-3: FIXITwill be managed with MySQL database management system.

OE-4: FIXIT will run on any Android smart phone with a working internet connection.

3.4.1 Technology Platform:

3.4.1.1 Android-Based Front End:

FixIT's front-end would be developed for android-based smart phones, providing the users with the interface to get registered on the server, lodge and view complaints. Android development tools (Android Studio) would be used as the development environment.

3.4.1.2 Cloud Service:

Application's core part would be deployed on the cloud for the working out and data manipulation. Free web-hosting service like www.datacollection.com would be used for the database handling. App will carry out data mapping at server's end, based on the user-query for registration of the complaint or viewing the status of the complaint.

3.4.1.3 Programming languages:

1. Java

3.4.1.4 Programming Environment

1. Android Studio
2. PHP Storm
3. Visual Studio

3.4.1.5 Web Languages

1. HTML
2. CSS
3. PHP

3.4.1.6 Database

1. MySql

3.5 Design and Implementation Constraints

CO-1: Android compatible platform, Java based, is needed for the end-user.

CO-2: All HTML code should conform to the HTML 5 standard.

CO-3: Lack of user-expertise in using the applications on Android cellular device.

CO-4: Internet connection needed.

CO-5: Use of English language as the only means of communication in the system.

CO-6: Location access using GPS is required.

3.6 User Documentation

UD-1: Final release will be accompanied with a user guide to inform users how to use FixIT. User documentation that would be delivered along with the final product

- User manual

3.7 Assumptions and Dependencies

AS-1: Basic assumption for development of FixIT is that system should be available 24/7 since complaint can be lodged at any time.

AS-2: The users will not misuse the application to lodge false complaints or create complaints where there aren’t any.

AS-3: The administration officials will be honest and would notterminate the complaints without doing any work.

AS-4: The people are willing to take a moment to lodge a complaint using the application.

AS-5: The server will be able to take care of large numbers of requests especially when initially launched, as there will be loads of complaints.

AS-6: Users of (FIXIT) are assumed to have an Android smart phone with internet access.

D-1: There will be a permanent dependency on the internet and GPS, as without these complaints won't be able to be submitted and the administration won't be able to update the progress log at all.

D-2: System is entirely depending upon thecloud server for 24/7 access since our database and admin panel reside on that server.

Chapter 4

4 Software Requirements Specification

4.1 System Features

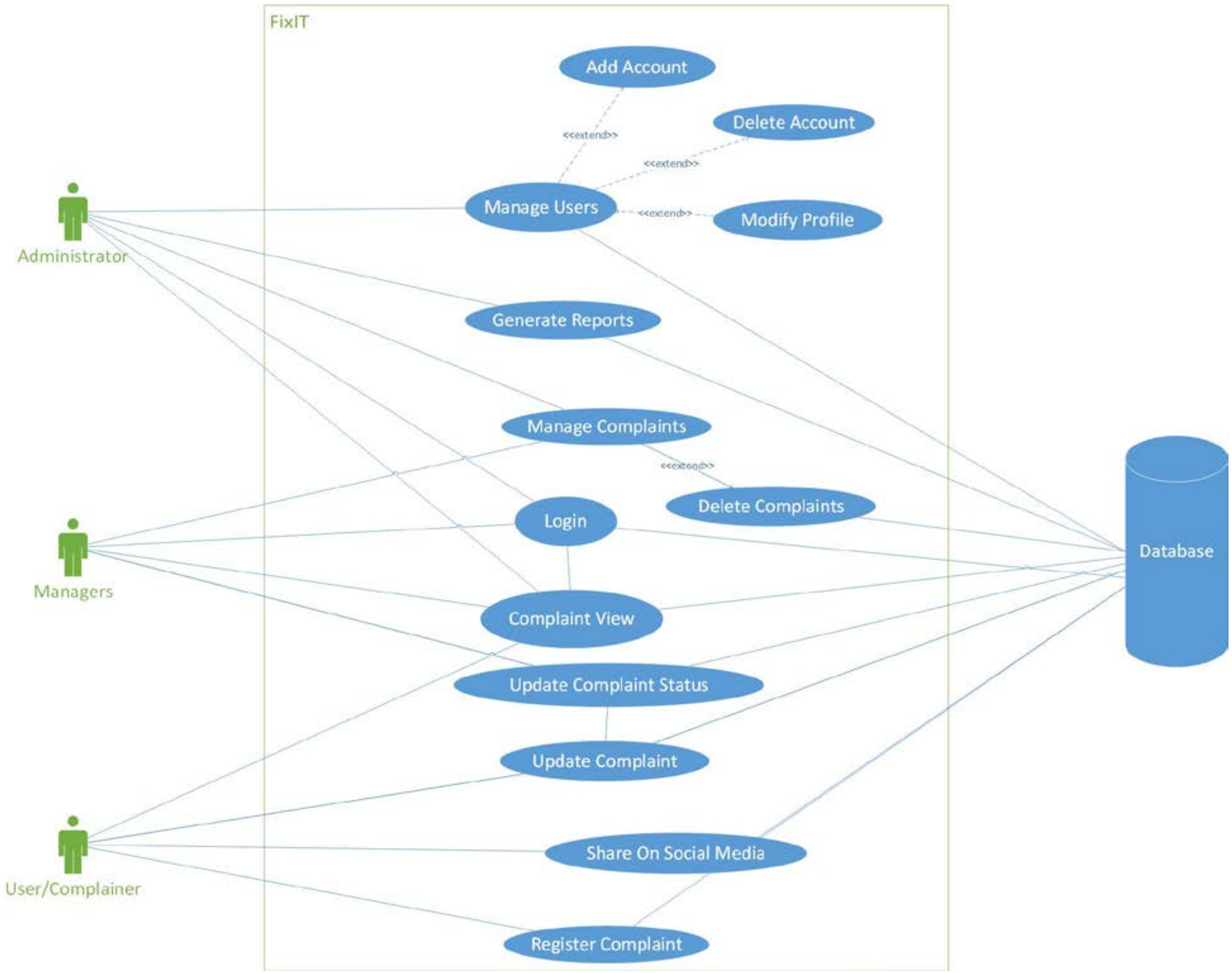


Figure 2 Use Case Diagram

4.1.1 Account creation

4.1.1.1 Description and Priority

The system will enable the all the users of the locality to create their user accounts, while providing them with the access rights based on their role in the system. The account creation with particular rights is for the managing staff only. This account will enable them to login, process the complaints and comment on them. The priority of this system feature is high.

4.1.1.2 Stimulus/Response Sequences

Input: The user will provide its credentials to create a user account.

Output: The account with specific rights to each person will be created.

4.1.1.3 Functional Requirements

REQ-1: FIXIT shall allow the users to create accounts.

REQ-3: FIXIT shall allow the Administrator to create and manage the accounts of users.

4.1.2 Login/access rights

4.1.2.1 Description and Priority

The system will enable the users to login. The users will provide their unique ID and password to access their account. Manager Users will be able to manage particular things based on their hierarchy and give rights accordingly.

4.1.2.2 Stimulus/Response Sequences

Input: User will enter login credentials to the system.

Output: The system will grant the valid user, the access to the features according to his access rights

4.1.2.3 Functional Requirements

REQ-1: FIXIT shall allow all the users to login to their account.

REQ-2: FIXIT shall grant the logged in user, the access to the system features.

4.1.3 User credentials entry

4.1.3.1 Description and Priority

The system will require the complainers to enter their certain credentials for their identification purposes, including name, passwords and basic info.

4.1.3.2 Stimulus/Response Sequences

Input: The user will enter his credentials in the FIXIT App.

Output: The system will accept the provided credentials and allow the user to perform required action.

4.1.3.3 Functional Requirements

REQ-1: FIXIT shall enable the user to enter his credentials.

4.1.4 Location determination

4.1.4.1 Description and Priority

The application will determine the user's location using the GPS. The user location will be marked using GPS which will help the manager staff to get the location of the complaint-area.

4.1.4.2 Stimulus/Response Sequences

Input: The user will run the app.

Output: The system will mark the location of the user where any problem is reported.

4.1.4.3 Functional Requirements

REQ-1: FIXIT shall determine the user-location.

4.1.5 Log maintenance

4.1.5.1 Description and Priority

A log of all complaints will be maintained on the server, accessible to the users. The log will contain the record of all the reported complaints and their resolution status.

4.1.5.2 Stimulus/Response Sequence

Input: Complainers and managing staff will populate the server with complaints-lodging and their related resolution-status.

Output: The log of the activities performed by the managing staff and the user will be maintained.

4.1.5.3 Functional Requirements

REQ-1: FIXIT will maintain the log of all the registered complaints.

REQ-2: FIXIT will allow the administrator to read all the logs maintained at server-end.

REQ-3: FIXIT will allow the complainer to check the status of his complaint.

4.1.6 Defining a complaint

4.1.6.1 Description and Priority

The complainant should be able to enter a description of his complaint and category including sanitation, solid-waste disposal, road maintenance, terrorism problem, kidnapping and bombing along with the snapshot.

4.1.6.2 Stimulus/Response Sequences

Input: The user will lodge a complaint and will add a description.

Output: The complaint will be registered along with the description and snapshot.

4.1.6.3 Functional Requirements

REQ-1: FIXIT shall allow the complainer to enter the definition of his complain.

REQ-2: FIXITshall allow the complainer to upload the snapshot with his complaint.

REQ-3:FIXIT shall allow the complainer to specify the category of service, under which the complaint falls.

4.1.7 Task assignment to the concerned staff

4.1.7.1 Description and Priority

Complaint lodged by the complainer will be assigned to the concerned staff, based on the area and nature of the complaint.

4.1.7.2 Stimulus/Response Sequences

Input: Complainers will populate the server with complaints.

Output: The server will assign the task to the concerned staff of the specific area.

4.1.7.3 Functional Requirements

REQ-1: FIXIT shall assign the registered complaint to the concerned staff.

4.1.8 Comments on the complaint

4.1.8.1 Description and Priority

The concerned people will be allowed to comment on the status of the complaint and discuss it on the forum directly.

4.1.8.2 Stimulus/Response Sequence

Input: The complaint will be pending but still people can comment and share their views.

Output: A comment will also be shown to all the users using the FIXIT.

4.1.8.3 Functional Requirements

REQ-1: FIXIT shall allow any user to comment on the problems.

4.1.9 Admin can delete the comments and complaints

4.1.9.1 Description and Priority

Admin of the system will be allowed to delete any comment.

4.1.9.2 Stimulus/Response Sequence

Input: Admin will delete the comment.

Output: Comment will be deleted.

4.1.9.3 Functional Requirements

REQ-1: FIXIT shall allow the admin to delete any comment and complaints.

4.1.10 Post Complaint-resolution Feedback

4.1.10.1 Description and Priority

When the complaint is resolved, the concerned staff (responsible for resolving the complaint) will update the complaint status as resolved, and will initiate a feedback report to the system database. The system will update its log and send the feedback on the complaint.

4.1.10.2 Stimulus/Response Sequence

Input: The Concerned Staff will update the system database by his feedback report on the resolved complaint.

Output: The system logs will be updated and the feedback report will be sent to complainant

4.1.10.3 Functional Requirements

REQ-1: FIXIT will allow the concerned staff to update the status of the resolved-complaint.

REQ-2: FIXIT will update the log file with the information of the resolved complaint.

REQ-3: FIXIT will send an update for the resolved complaint.

4.1.11 Complaint sharing on Social Media

4.1.11.1 Description and Priority

The users can share the complaints on different social media sites (Facebook, Twitter etc.) if they don’t find it appropriate to discuss anything on the forum.

4.1.11.2 Stimulus/Response Sequence

Input: The user will share the complaint on any social site.

Output: The complaint will be shared on the selected social site.

4.1.11.3 Functional Requirements

REQ-1: FIXIT will allow the users to share problems on social media.

4.2 Other Nonfunctional Requirements

4.2.1 Performance Requirements

Certain functionalities will be required, based on the performance and response of FIXIT. FIXIT will take less than 12 seconds to send data to the server (GPS coordinates and Photograph). Up to 75 users can lodge the complaint simultaneously.

4.2.2 Safety Requirements

SF-1: In case of data loss, system will back up the data and will restore it as per demand.

SE-2: System will be deployed on online cloud server platform so they’ve their inherent fault-tolerance capabilities.

4.2.3 Security Requirements

SE-1: Users shall be required to log into FIXIT for their own credentials.

SE-2: The system shall permit only authorized members to do administrator’s task.

SE-3: The system shall permit users to view only their own profile and data that are intended for them.

SE-4: The system must perform an encoding methodology such as hashing to save all passwords securely.

SE-5: The System will provide confidentiality and integrity.

4.3 Software Quality Attributes

Quality attributes of FIXIT are described below. By following these attributes, the quality of FIXIT will be improved.

4.3.1 Runtime System Qualities

At runtime FIXIT has to provide its users with functionalities so that they can publish and search for the desired services. Some of the runtime qualities that should be considered in the development of FIXIT are described here.

4.3.1.1 Functionality

FIXIT must provide functions to publish and search the different services. FIXIT must provide the functions of authentication of user.

4.3.1.2 Availability

FIXIT should be available 24/7 since the complaint can be lodged at any time. If at all system is down so the servers will take about 15 minutes to start the FIXIT again.

4.3.1.3 Usability

Usability is an important criterion in the development of FIXIT. The system should present all functionalities in such a way that nothing is missed by the user. The graphical user interface of app is to be designed with usability as the first priority. The app will be presented and organized in a manner that is both visually appealing and easy for the user to navigate.

4.3.1.4 Non-Runtime System Qualities

These are qualities of FIXIT which are required to make this software useful for further enhancements. It will also be helpful in future development as well as extending system to different environments.

4.3.1.5 Modifiability

FIXIT must support modifiability so any further improvements or features are easy to incorporate.

4.3.1.6 Portability

The system should work on WIFI as well as 3G network.

FIXIT should be able to run in different computer environments. The FIXIT server should be a platform-independent and should support interoperability

4.3.1.7 Interoperability

Mobile app will be interoperable with Google app engine.

4.3.1.8 Testability

Different quality tests should be performed so that FIXIT is free from faults and perform according to requirements.

Chapter 5

5 System Design Specifications

5.1 Overview of the module

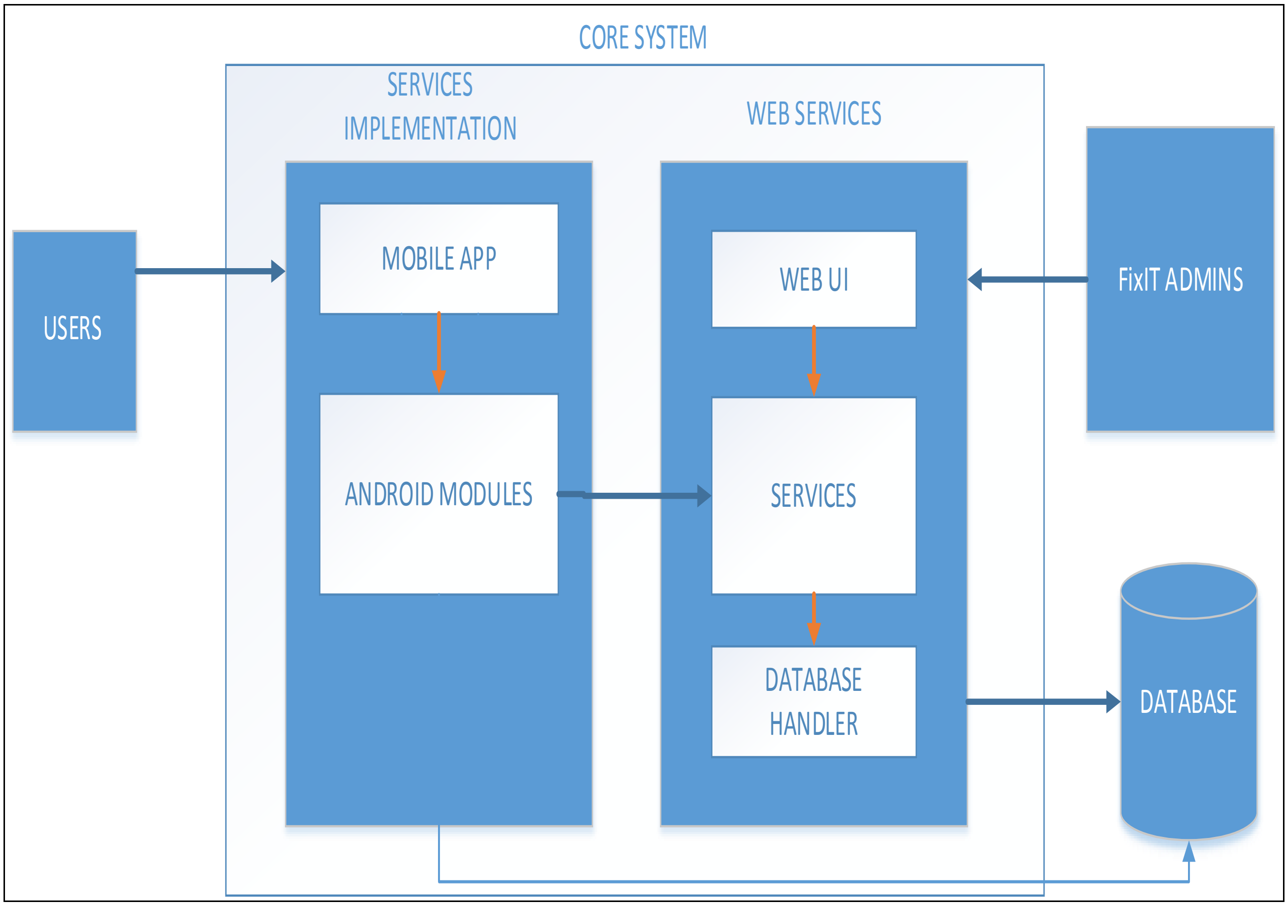


Figure 3 Module Overview

The system's architecture is mainly divided into four fundamental modules "Users", "Core system", "FixIT Administration", and the "database". It will further be having sub modules as shown in the following abstract diagram. Abstract diagram provides an overview of the system. Client accessing the system till the processing in Databases.

5.2 Architecture

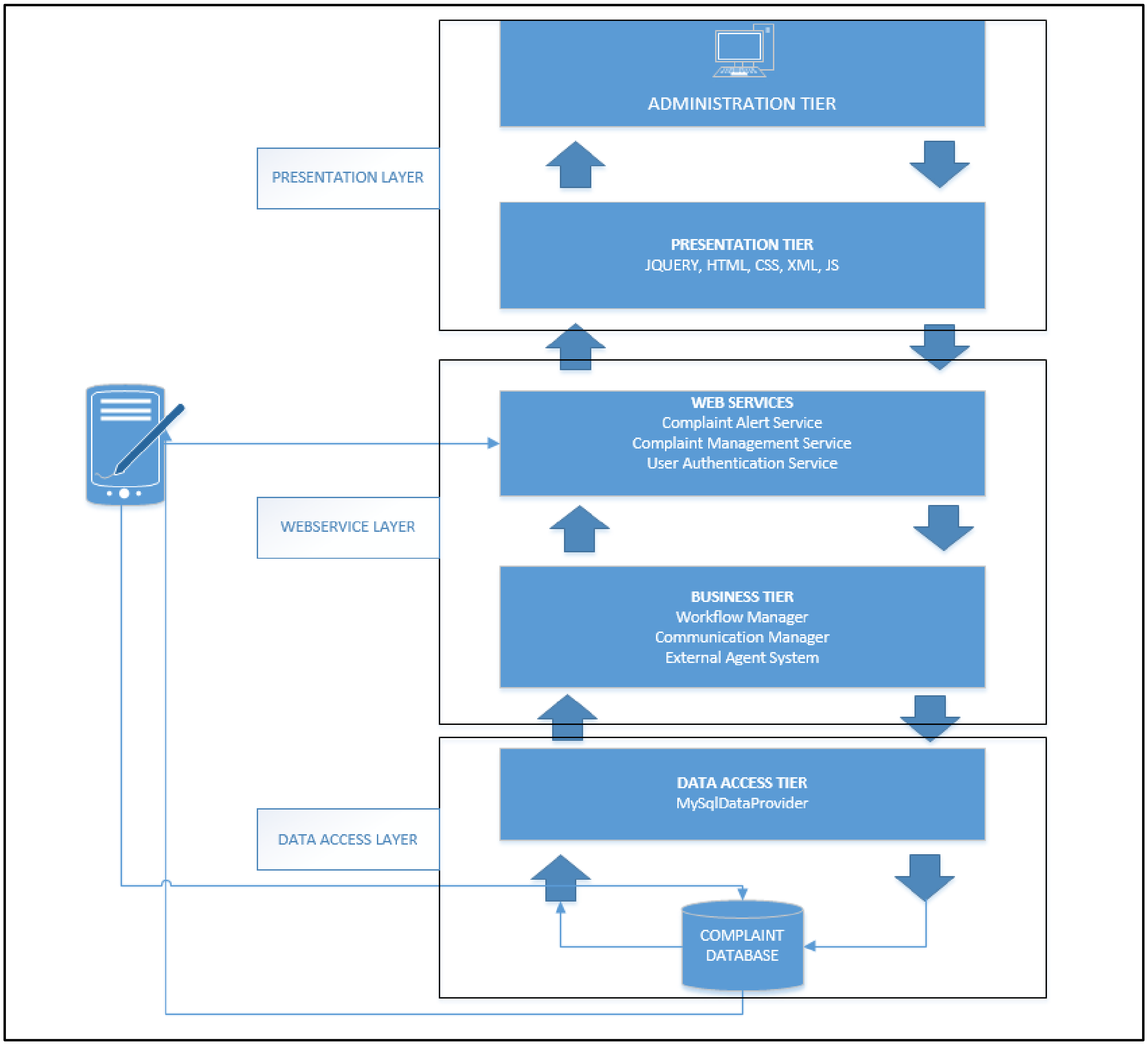


Figure 4 System Architecture

Once a user of FixIT Android app lodges a complaint, the web interface informs the managing authority that a new complaint has been lodged, and the members can access the said complaint. The complaint is automatically forwarded to the relative department. A complaint number is issued. The authority can update complaints with the action they take, and can close solved complaints as well (User will know if a problem is solved by seeing the status of the Problem being changed from RED to GREEN). A record of all complaints is maintained in the complaint database.

5.3 Class Diagram

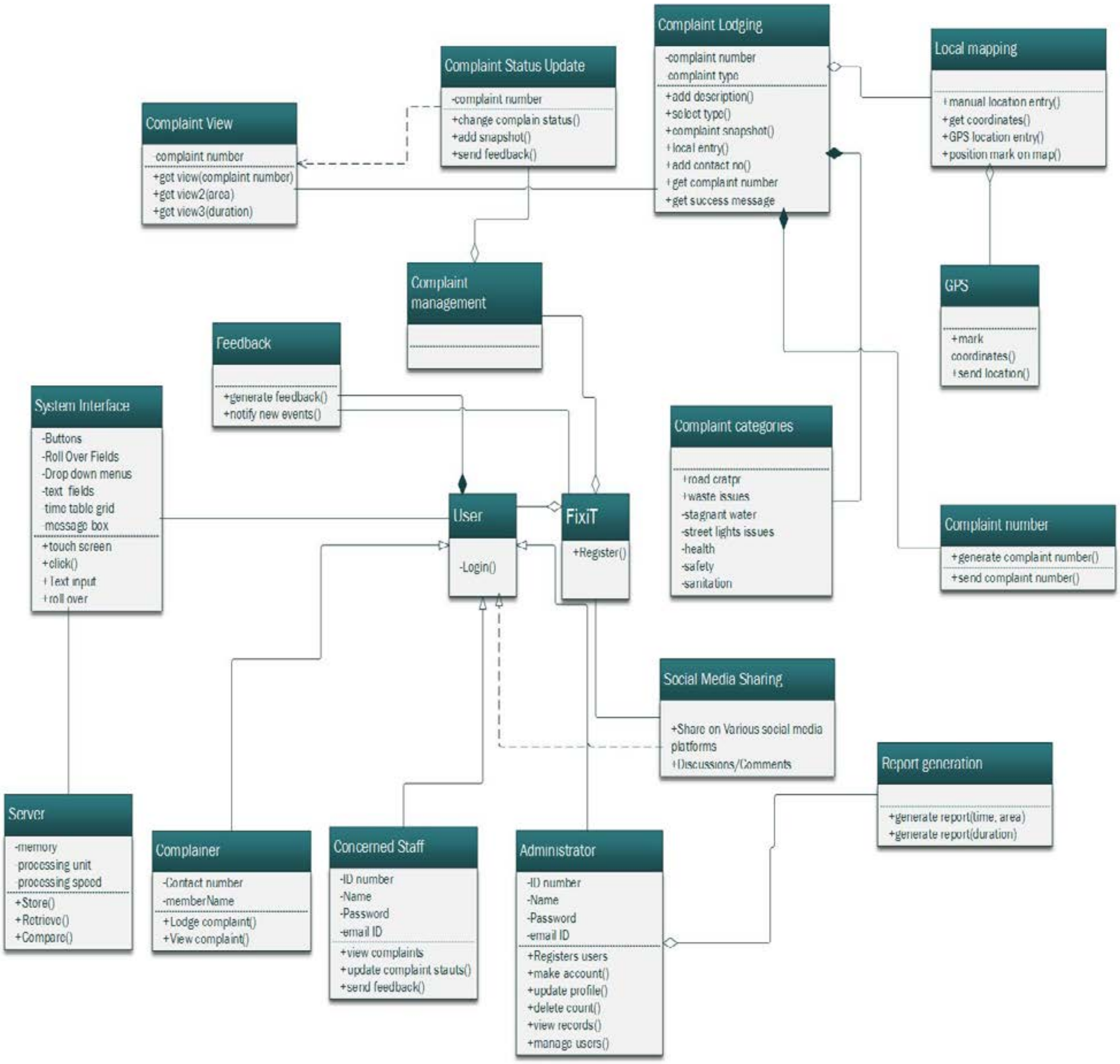


Figure 5 Class Diagram

The description for class diagram is shown on the next page:

Class name	Description
App	App class contains all the information that FixIT has to perform. It is the main class which will be acting as a gateway to all the other classes
Complaint management	Complaint management class will be handling all the functions and other classes related to complaints which include lodging, updating and viewing etc. It also provides a generalization to the classes as complaint lodging class viewing class and update class
User	User class contains all the information related to user management. Generalization with other classes of user categorization and the functions that performs all the user management functions.
System interface	It contains all the information to enable user to interact with FixIT. It has links to all the functions of different classes that on selection lead to different actions.
Server	This class contains the database objects and server information. It provides access to all the database objects and also saves the data
Complaint lodging	The class provides all the data and options to lodge a complaint.
Complaint viewing	This class handles the complaint viewing options and let users to view their complaints according to their required format.
Complaint status update	This class contains all the functionality to update the complaint status.
Location mapping	This class handles the location mapping system of FixIT as it allows the

	user to choose between options to enter location.
GPS	This class actually deals with the GPS system to store the coordinates of the person who enters location via GPS.
Concerned Staff	This class contains the information about the privileges and the functions that a concerned staff can perform.
Complainer	This class contains the information about the privileges and can perform the functions that a complainer is allowed to perform.
Administrator	The class handles all the functions that an administrator can perform and maintains the rights of an administrator.
Complaint categories	This class contains the information about the categories of complaints that a complainer can choose while lodging complaint.
Report generation	This class actually generates a report of complaints as if said by Admin.
Complaint number	This class contains the functions that generates the complaint numbers and send them back to complainer.
Feedback	The class generates feedback of actions performed by the users and also notify if users if there is some new news or event for them.

Table 1 System Features

5.4 Use Case Diagram

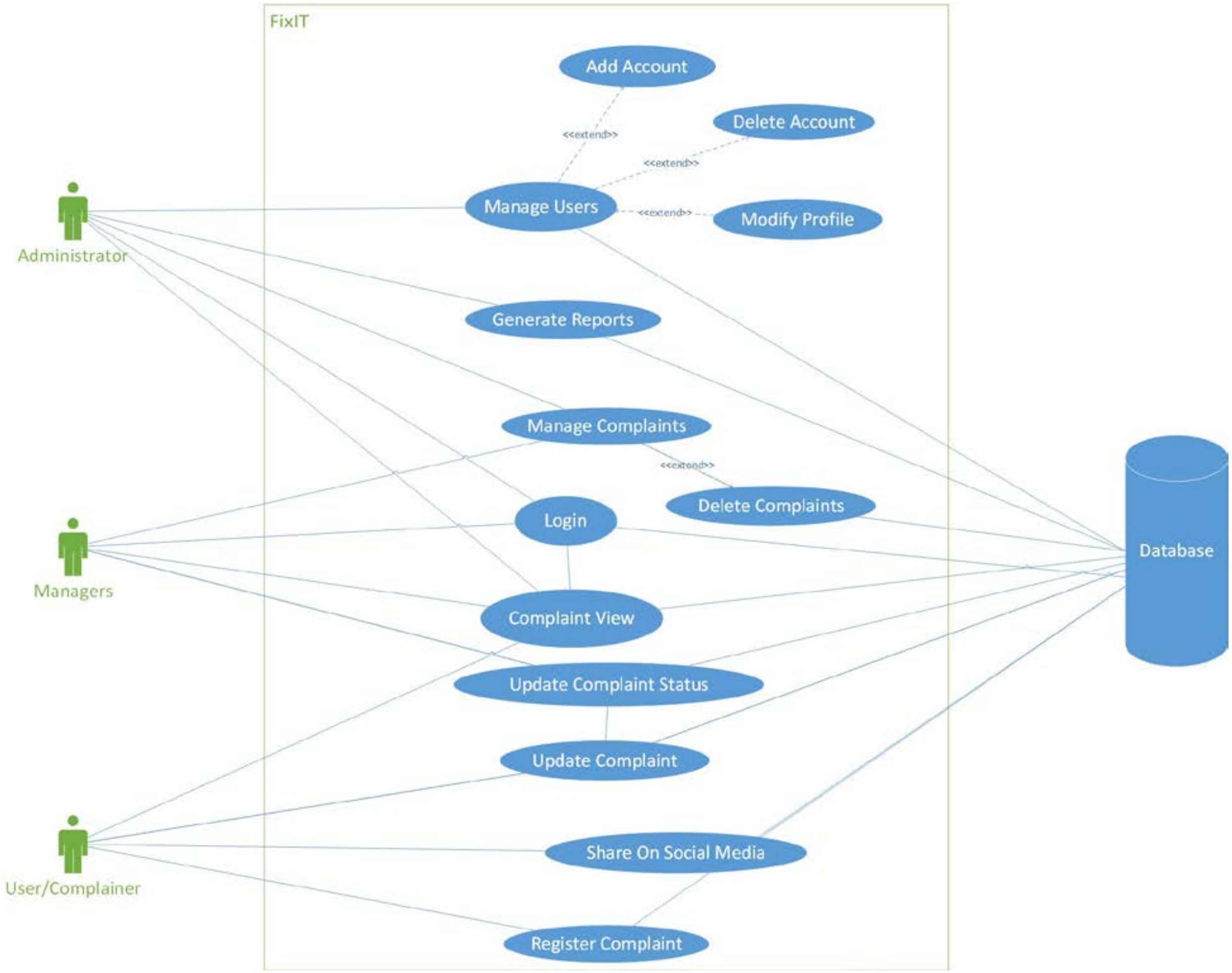


Figure 6 Use Case Diagram

5.5 Use Cases Description

5.5.1 Manage Users

Use Case ID:	1
Use Case Name:	Manage Users

Actors:	Administrator		
Created By:	Zohaib	Last Updated By:	Zohaib
Date Created:	27/10/2016	Date Last Updated:	27/10/2016
Description:	1. Admin has to login to the system to manage users i.e. create, read, update and delete user profiles.		
Preconditions:	1. Admin has to login.		
Post conditions:	1. The System must record the change.		
Normal Flow (primary scenario):	1. The actor creates, reads, updates and deletes the user details. 2. Click on create or read or update or delete button as required.		
Alternative Flows:	1. The actor will contact the system maintenance team to check if there is some error with database systems and has to resolve the error.		

Table 2 Manage Users

5.5.2 Login

Use Case ID:	2		
Use Case Name:	Login		
Actors:	Administrator, Managers,Users		
Created By:	Arslan	Last Updated By:	Arslan
Date Created:	27/10/2016	Date Last Updated:	27/10/2016
Description:	A user tries to login to the system.		
Preconditions:	1. User has to open the login page first.		
Post conditions:	1. If the use case was successful, the actor is now logged into the system. If not the system state remains unchanged.		

Normal Flow (primary scenario):	This use case starts when an actor wishes to log into the System. 1. The system requests that the actor enter his/her name and password. 2. The actor enters his/her name and password. 3. The system validates the entered name and password and logs the actor into the system.
Alternative Flows:	1. **Invalid Name / Password** If in the *Basic Flow* the actor enters an invalid name and/or password, the system displays an error message. The actor can choose to either return to the beginning of the *Basic Flow* or cancel the login, at which point the use case ends.

Table 3 Login

5.5.3 Registration

Use Case ID:	3		
Use Case Name:	Registration		
Actors:	Managing Staff, Users		
Created By:	Zohaib	Last Updated By:	Zohaib
Date Created:	27/10/2016	Date Last Updated:	27/10/2016
Description:	A user tries to sign up in to the system.		
Preconditions:	1. User has to open the sign up page first.		
Post conditions:	1. The System must record the membership information of the new member.		

Normal Flow (primary scenario):	1. The member enters the membership details on the screen and clicks the sign up button. 2. The system checks for the availability of the username. 3. The system generates the membership ID. 4. The system records the membership information of the new member, in the database.
Alternative Flows:	1. The member enters the membership details on the screen and clicks the sign up button. 2. The system checks for the availability of the username. 3. The system displays an error report if the username is not available.
Non-functional Requirements	The system must perform an encoding technique such as hashing to save all passwords securely.

Table 4 Registration

5.5.4 Edit/View Profile

Use Case ID:	4		
Use Case Name:	Edit Profile		
Actors:	Administrator, Managers, Users		
Created By:	Arslan	Last Updated By:	Arslan
Date Created:	27/10/2016	Date Last Updated:	27/10/2016
Description:	A user tries to edit/view his profile.		
Preconditions:	1. User has to open the sign up page first.		
Post conditions:	1. The System must show the profile and record the changes made therein.		

Normal Flow (primary scenario):	1. The user login credentials and signs in to the system. 2. The user clicks profile options. 3. The system enables the user to view or modify his profile information. 4. The system records the changes made in the profile.
Alternative Flows:	1. The user tries to modify his profile name. 2. The system throws error message, mentioning that user cannot change his profile name.

Table 5 Edit/View Profile

5.5.5 Check Complaint Status

Use Case ID:	5		
Use Case Name:	Check Project Status		
Actors:	Administrator, Manager, User		
Created By:	Muneeb	Last Updated By:	Muneeb
Date Created:	27/10/2016	Date Last Updated:	27/10/2016
Description:	A user has to login to the system to check the status of the complaint.		
Preconditions:	1. User has to open Check Status page first. 2. User has to know the Date / ID of the complaint to check the status.		
Post conditions:	1. The System shows the status of complaint.		
Normal Flow (primary scenario):	1. User enters Date / ID of the complaint. 2. System shows the status of the complaint.		

Alternative Flows:	**Complaint Not Found** 1. User enters ID of the complaint. 2. Entered complaint ID is not found. So, system tells no record for this complaint number exists.

Table 6 Check Complaint Status

5.5.6 Update Complaint Status

Use Case ID:	6		
Use Case Name:	Update Complaint Status		
Actors:	Concerned Manager		
Created By:	Zohaib	Last Updated By:	Zohaib
Date Created:	27/10/2016	Date Last Updated:	27/10/2016
Description:	The system will enable the concerned supervisor to update the status of the complaint after having it resolved.		
Preconditions:	1. User has to login to the system. 2. User has to open the complaint status form of the concerned complaint.		
Post conditions:	1. The system will update the status of the complaint.		
Normal Flow (primary scenario):	1. The user selects the complaint and updates its status. 2. The system records the changes.		
Alternative Flows:	**Limited Connectivity** In case of connectivity issues with the database, the system will allow the user to update complaint status.		

Table 7 Update Complaint Status

5.5.7 Lodge Complaint

Use Case ID:	7		
Use Case Name:	Lodge Complaint		
Actors:	Complainer		
Created By:	Zohaib	Last Updated By:	Zohaib
Date Created:	27/10/2016	Date Last Updated:	27/10/2016
Description:	User intends to lodge a complaint.		
Preconditions:	1. User accesses the complaint registration form.		
Post conditions:	1. The system registers the complaint.		
Normal Flow (primary scenario):	1. The user fills in his credentials including name, phone number in the specified fields. 2. The user types in the complaint-description field, all the details about his complain in 500 characters. 3. The system enables the user to attach a snapshot of the complaint area.		
Alternative Flows:	**Limited Connectivity** In case of connectivity issues with the database, the system will allow the user to resubmit the complaint.		

Table 8 Lodge Complaint

5.5.8 Complaint number generation

Use Case ID:	8

Use Case Name:	Complaint number generation		
Actors:	Complainer		
Created By:	Zohaib	Last Updated By:	Zohaib
Date Created:	27/10/2016	Date Last Updated:	27/10/2016
Description:	User will register a complaint in reply will get a complaint registration number.		
Preconditions:	1. User has to access the application 2. User has to open the complaint registration page. 3. User has to register his complaint successfully.		
Post conditions:	1. User has positively registered complaint and acquired a complaint registration number.		
Normal Flow (primary scenario):	1. User accesses the app. 2. User lodges a complaint. 3. After registration user acquires a complaint registration number.		
Alternative Flows:	**Limited Connectivity** In case of connectivity issues with the database, the system will permit the user to retry submitting the complaint information.		

Table 9 Complaint Number Generation

5.5.9 Location Tracking

Use Case ID:	9
Use Case Name:	Location Tracking
Actors:	Complainer, Manager

Created By:	Zohaib	Last Updated By:	Zohaib
Date Created:	27/10/2016	Date Last Updated:	27/10/2016
Description:	The user location will be marked using GPS which will help the Manager staff to acquire the location of the complaint-area.		
Preconditions:	1. User will lodge a complaint and permit the allow button for location marking in app.		
Post conditions:	1. User location will be marked by the app using GPS.		
Normal Flow (primary scenario):	1. User access the system 2. User lodge a complaint 3. User permits the app to mark his location.		
Alternative Flows:	1. User access the system. 2. User manually enters his location while lodging complaint.		

Table 10Location Tracking

5.5.10 Comment Section

Use Case ID:	9		
Use Case Name:	Comments		
Actors:	Complainer, Manager		
Created By:	Zohaib	Last Updated By:	Zohaib
Date Created:	27/10/2016	Date Last Updated:	27/10/2016
Description:	1. The user shall be able to comment on the complaints submitted by the users.		
Preconditions:	1. The complaint must be in the database before we are done with the comment.		

Post conditions:	1. Comment will be exposed on the complaint.
Normal Flow (primary scenario):	1. User access the system 2. User Comment on a complaint 3. Comment is shown with the complaint.
Alternative Flows:	1. User comments. 2. No comment is over due to being offline.

Table 11 Comment Section

5.5.11 Social Sharing

Use Case ID:	10		
Use Case Name:	Social Sharing		
Actors:	Complainer, Manager		
Created By:	Zohaib	Last Updated By:	Zohaib
Date Created:	27/10/2016	Date Last Updated:	27/10/2016
Description:	1. The user shall be able to share photos and explanation of complaint on social media.		
Preconditions:	1. The complaint must be in the database before the comment is completed.		
Post conditions:	1. The photo along with the explanation will be shared on Social media.		
Normal Flow (primary scenario):	1. User access the system 2. User share a complaint on Social Media 3. Photo and explanation are shared on social media		
Alternative Flows:	1. User shares. 2. Not shared because of being offline.		

Table 12 Social Media Sharing

5.6 Sequence Diagrams

5.6.1 Complaint lodging

The underneath diagram describes the sequence of actions that occurs when a complaint has been lodged by a complainer. All the actions are in time lined.

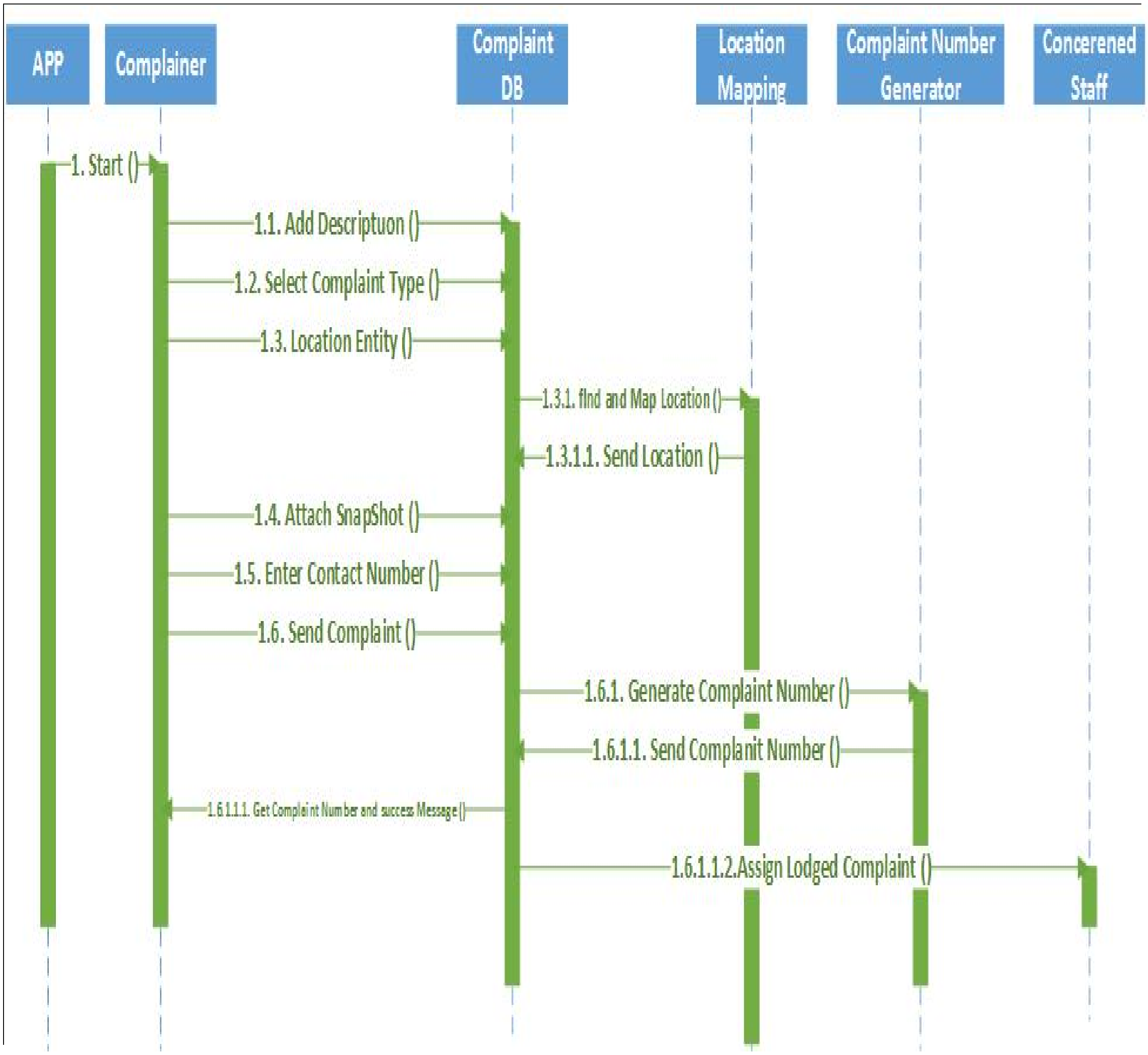

APP
Complainer
Complaint DB
Location Mapping
Complaint Number Generator
Concerened Staff
1. Start ()
1.1. Add Descriptuon ()
1.2. Select Complaint Type ()
1.3. Location Entity ()
1.3.1. flnd and Map Location ()
1.3.1.1. Send Location ()
1.4. Attach SnapShot ()
1.5. Enter Contact Number ()
1.6. Send Complaint ()
1.6.1. Generate Complaint Number ()
1.6.1.1. Send Complanit Number ()
1.6.1.1.1. Get Complaint Number and success Message ()
1.6.1.1.2.Assign Lodged Complaint ()

Figure 7 Sequence diagram Complaint lodging

5.6.2 Location Mapping Manually

The underneath diagram describes the order of actions that occurs when a complainer while lodging complaint choose to enter his/her location manually not through GPS.

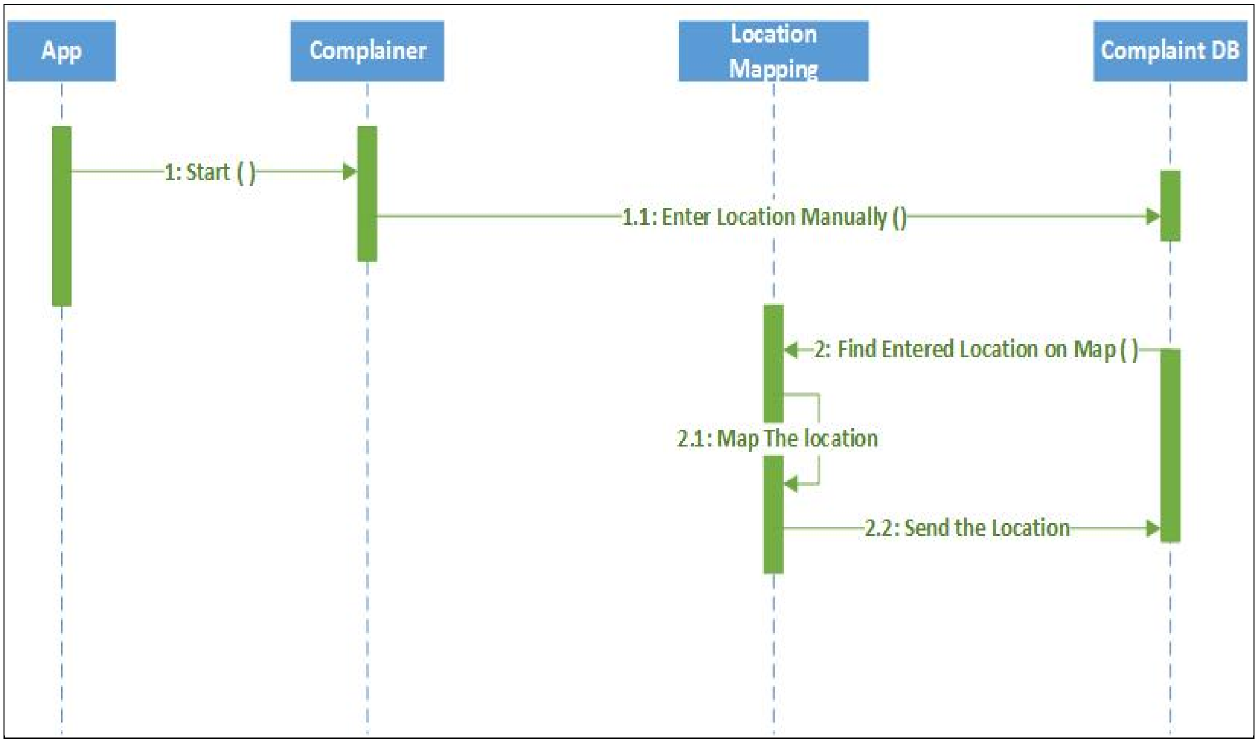


Figure 8 Sequence diagram Location Mapping manually

5.6.3 Edit/View Profile

The underneath diagram describes the order of actions that occurs when a User attempts to edit or view his/her profile. There is also clear optional outlook in the order of editing or viewing profiles.

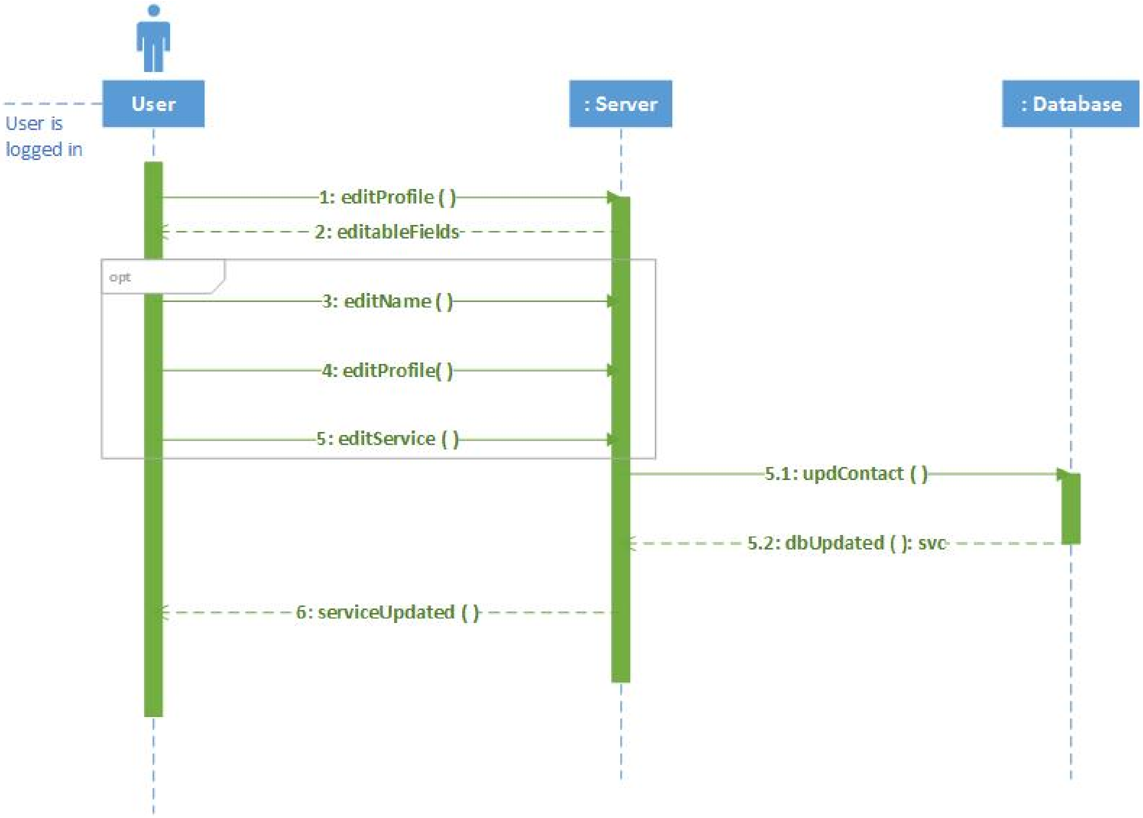


Figure 9 Sequence diagram Edit/View Profile

5.6.4 Location mapping through GPS

The underneath diagram describes the order of actions that occurs when a complainer while lodging complain enters his/her location through GPS not manually.

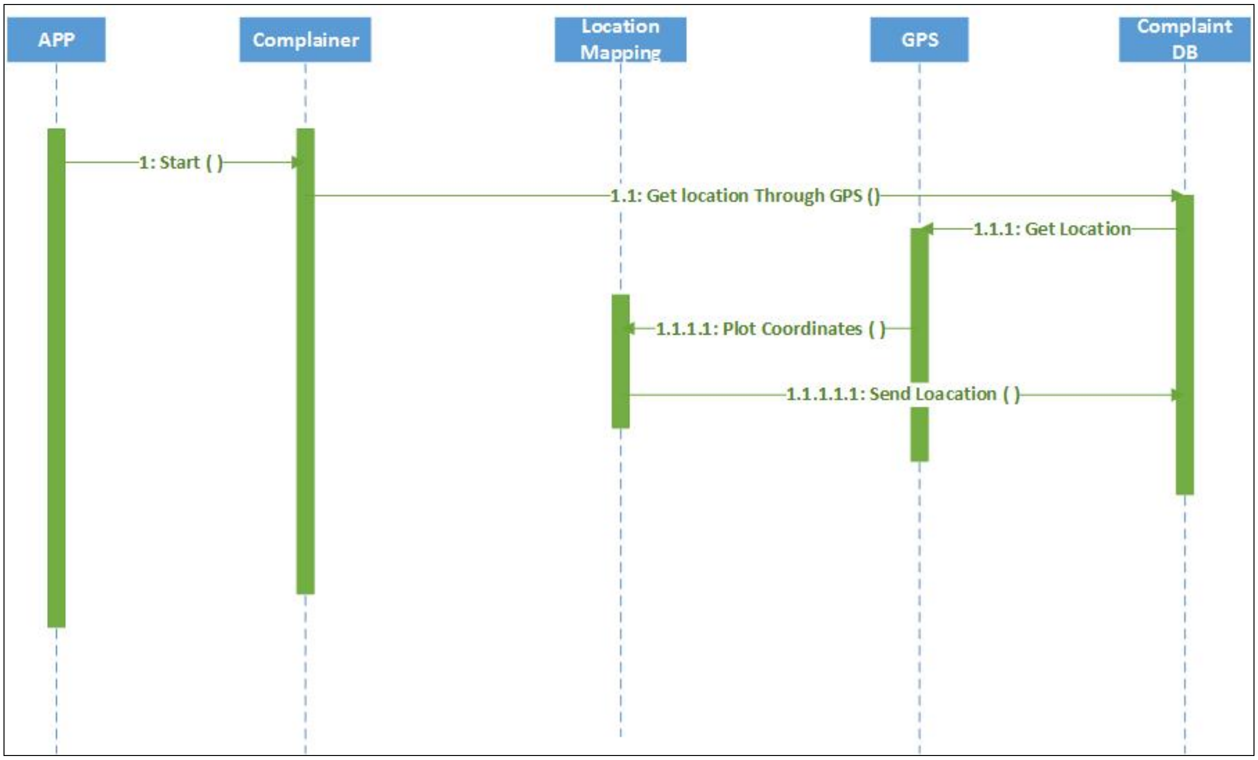

APP
Complainer
Location Mapping
GPS
Complaint DB
1: Start ( )
1.1: Get location Through GPS ()
1.1.1: Get Location
1.1.1.1: Plot Coordinates ( )
1.1.1.1.1: Send Loacation ( )

Figure 10 Sequence diagram Location mapping through GPS

5.6.5 Sharing on Social Media:

The diagram underneath explains the order of events that will take place when the user desires to share a complaint's content on various social media platforms. The appeal to share on social media will be launched from Server to the concerned managers who will approve the state of complaint, consequence to which Server will grant or deny user's request.

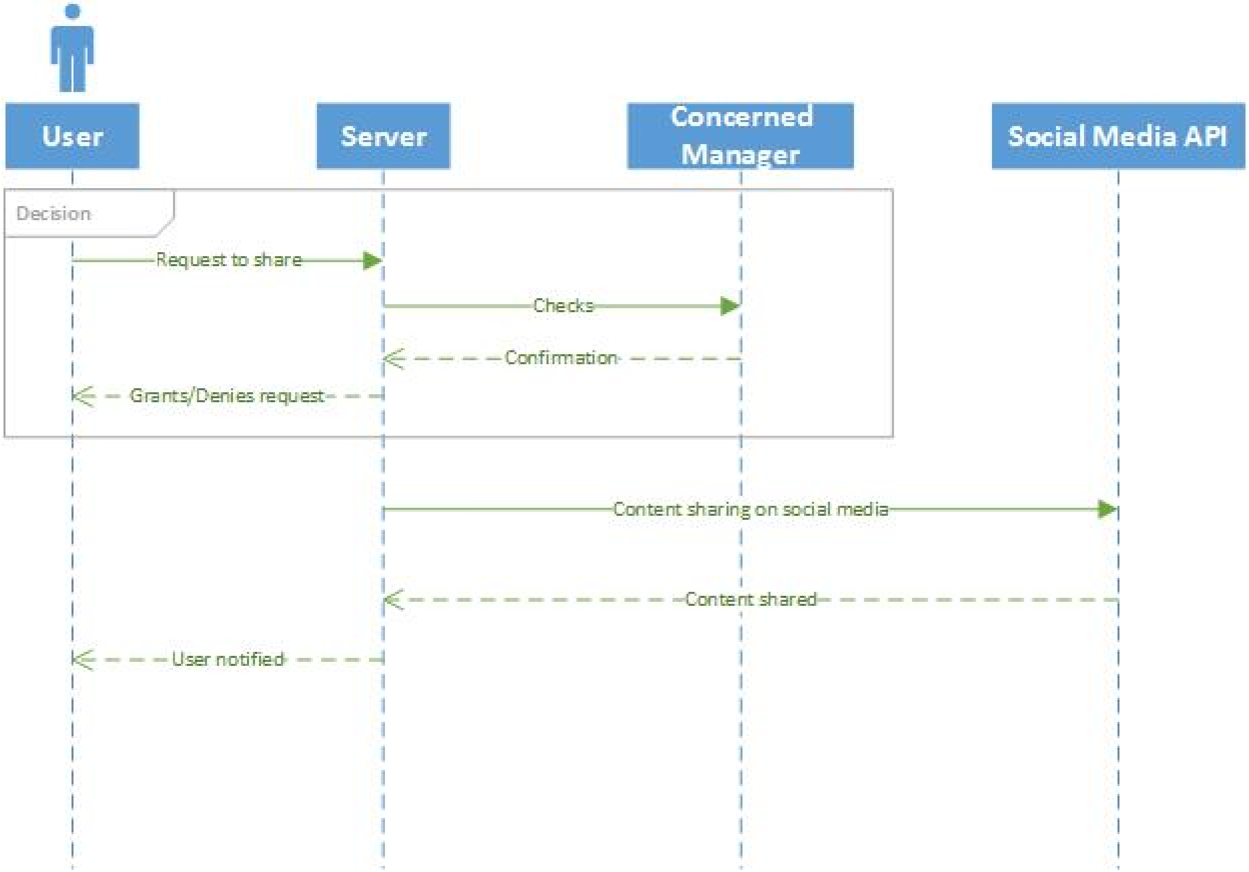


Figure 11 Social media sharing

5.6.6 Update complaint status

The underneath diagram describes the order of actions that occurs when a Concerned Staff updates the status of a resolved complaint. The elective factor of the update complaint status has been also stated that is of snapshot addition.

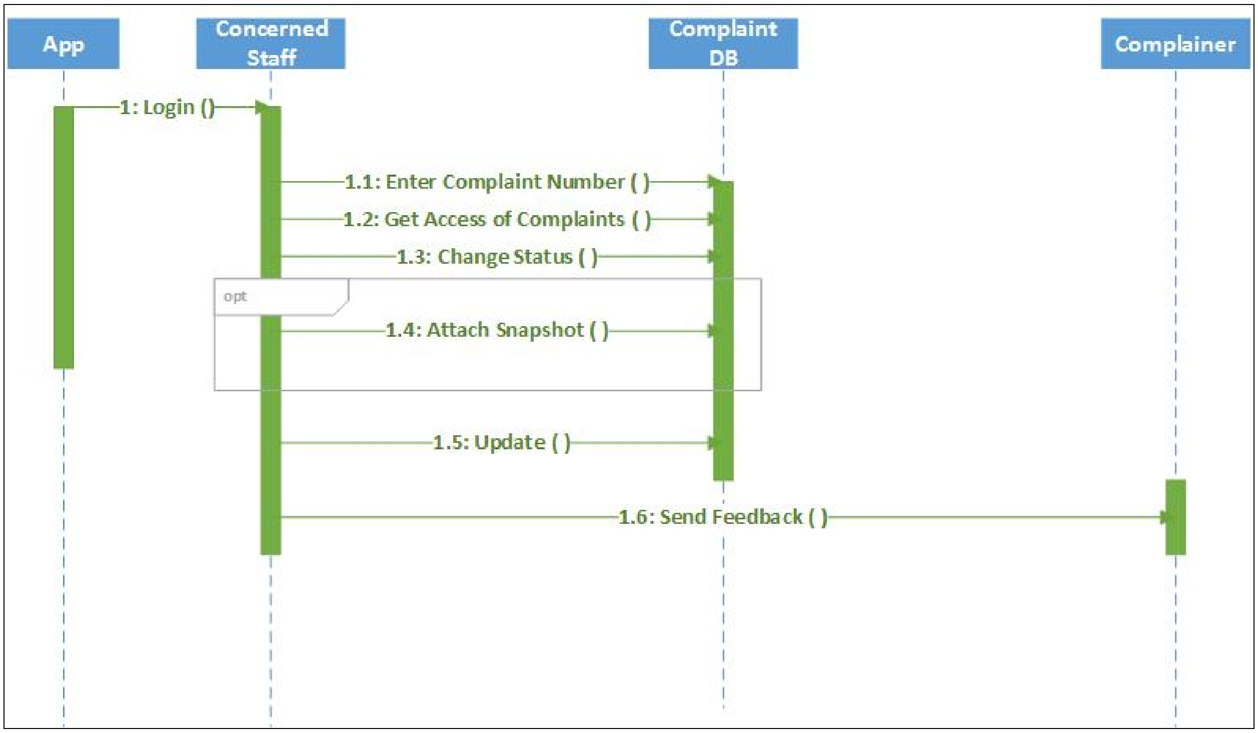


Figure 12 Sequence Diagram update complaint status

5.6.7 View complaint for Admin

The underneath diagram outlines the categorization of actions that occurs when Admin views complaints. The conclusion factor among different viewing options has been declared as in which form the complaints required to be viewed. The optional prospect of the events have also been provided for in case the Admin desires to print report about the complaints.

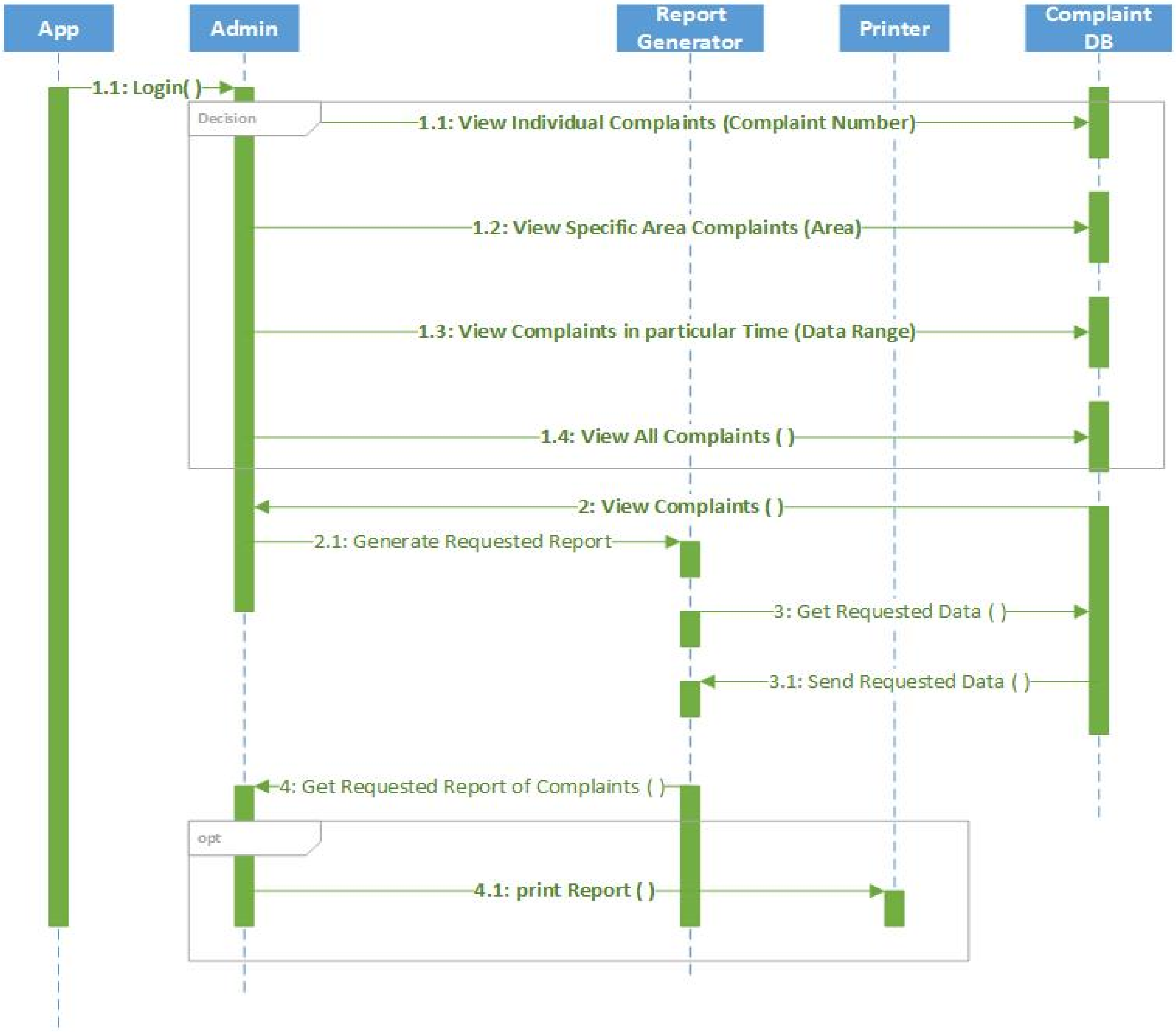

App
Admin
Report Generator
Printer
Complaint DB
1.1: Login( )
Decision
1.1: View Individual Complaints (Complaint Number)
1.2: View Specific Area Complaints (Area)
1.3: View Complaints in particular Time (Data Range)
1.4: View All Complaints ( )
2: View Complaints ( )
2.1: Generate Requested Report
3: Get Requested Data ( )
3.1: Send Requested Data ( )
4: Get Requested Report of Complaints ( )
opt
4.1: print Report ( )

Figure 13 Sequence diagram view complaint for CEO

5.6.8 View complaint for complainer

The under diagram outlines the order of actions that occurs when a complainer views his/her lodged complaint by giving complaint number.

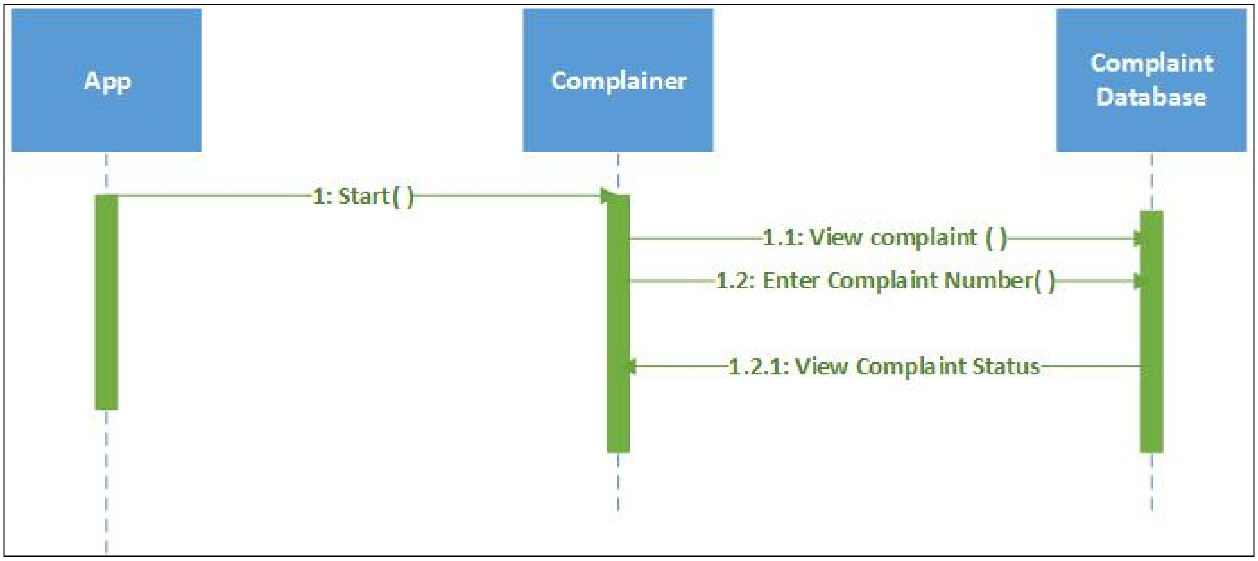


Figure 14 Sequence diagram view complaint for complainer

5.6.9 View complaints for concerned staff

The following diagram expresses the sequence of actions that materializes when a concerned staff wants to views complaints. The decision prospect for viewing options has been also satisfied in the diagram as the user desires to see complaint in which format either all either in certain time duration etc.

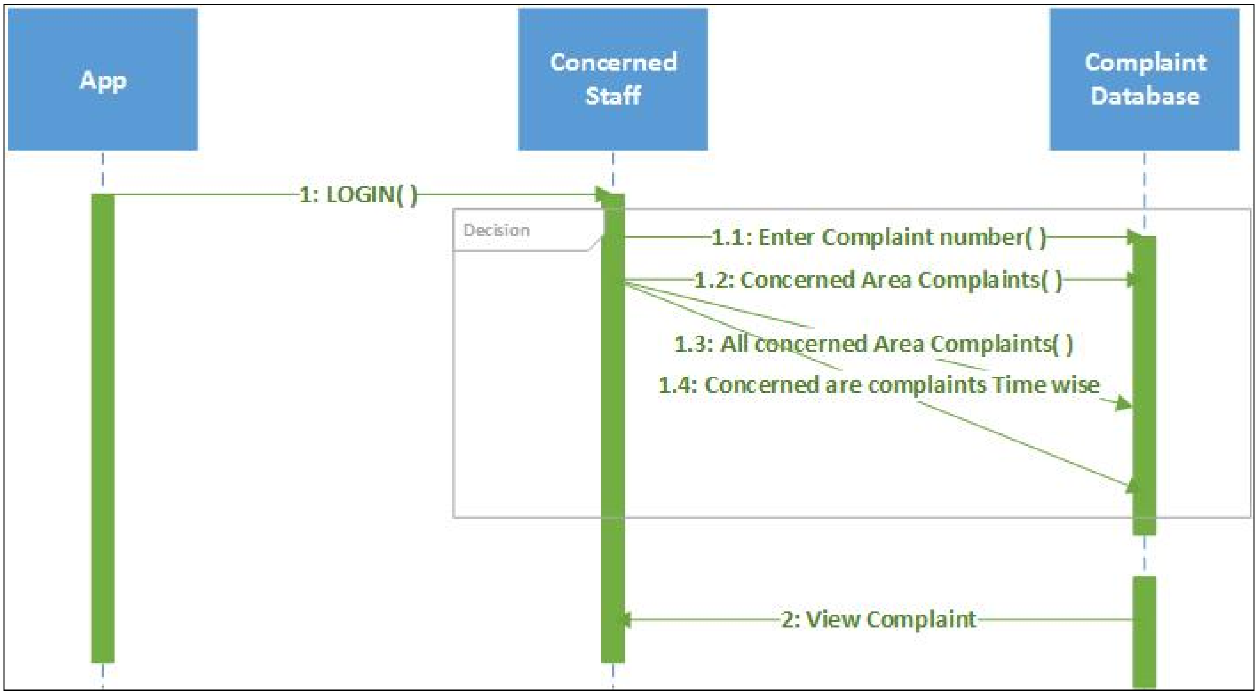

App
Concerned Staff
Complaint Database
1: LOGIN( )
Decision
1.1: Enter Complaint number( )
1.2: Concerned Area Complaints( )
1.3: All concerned Area Complaints( )
1.4: Concerned are complaints Time wise
2: View Complaint

Figure 15 Sequence diagram view complaint for concerned staff

5.7 Activity Diagrams

5.7.1 Location Mapping

The following diagram describes the stream of activities that a user need to execute while entering his/her location during complaint lodging.

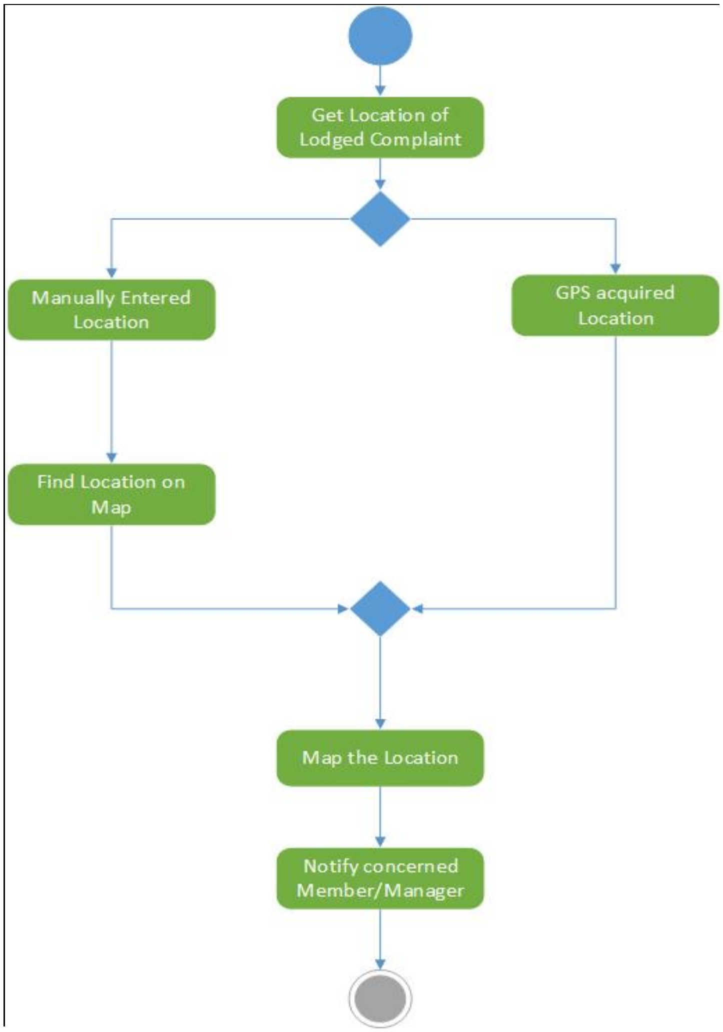

Get Location of Lodged Complaint
Manually Entered Location
GPS acquired Location
Find Location on Map
Map the Location
Notify concerned Member/Manager

Figure 16 Location Mapping

5.7.2 Complaint Lodging

The following diagram describes the stream of activities that a user wants to perform while lodging complaint. Certain requirements are there that complainer needs to fill as shown in diagram.

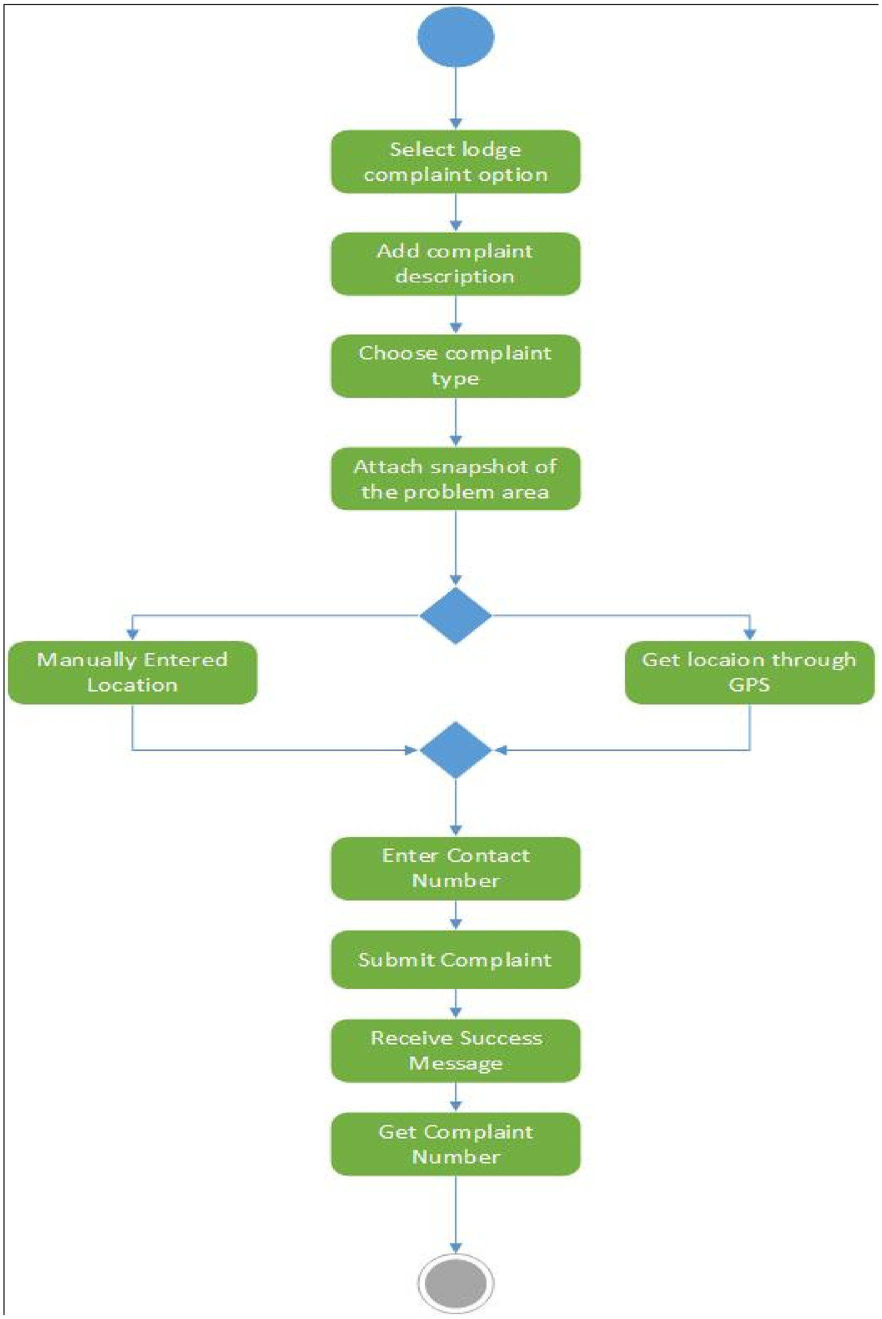
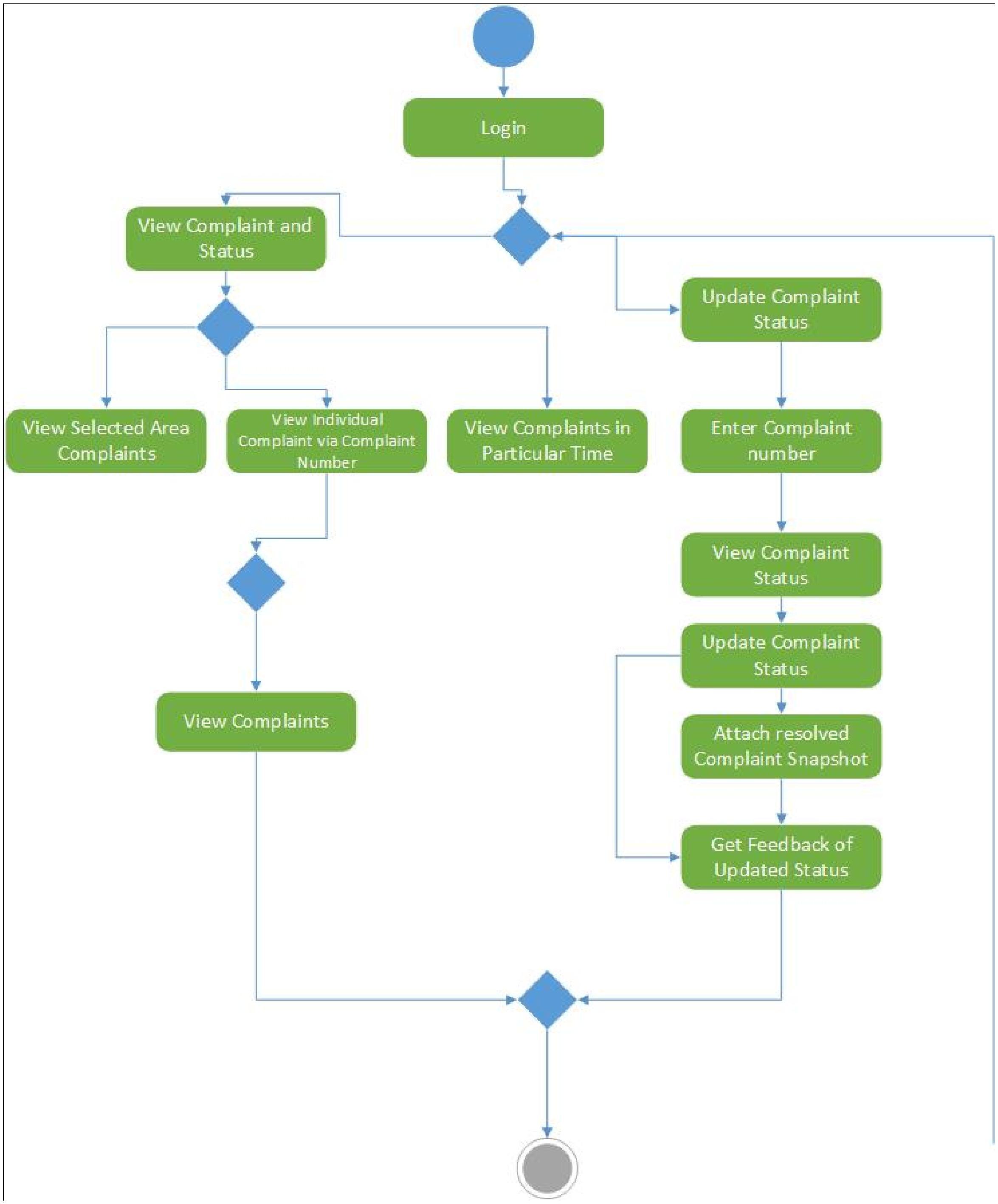


Figure 17 Complaint Lodging

5.7.3 Manage Comp for Admin and Concerned Staff

The following diagram describes the stream of activities that the advantaged users as CEO and concerned Staff performs to get the wanted operation done



Login
View Complaint and Status
Update Complaint Status
View Selected Area Complaints
View Individual Complaint via Complaint Number
View Complaints in Particular Time
Enter Complaint number
View Complaint Status
Update Complaint Status
View Complaints
Attach resolved Complaint Snapshot
Get Feedback of Updated Status

Figure 18 Manage complaints for Admin and Concerned Staff

5.7.4 Update Complaint status

The following diagram describes the stream of activities that a concerned Staff wants to execute to update complaint status after the complaint has been fixed

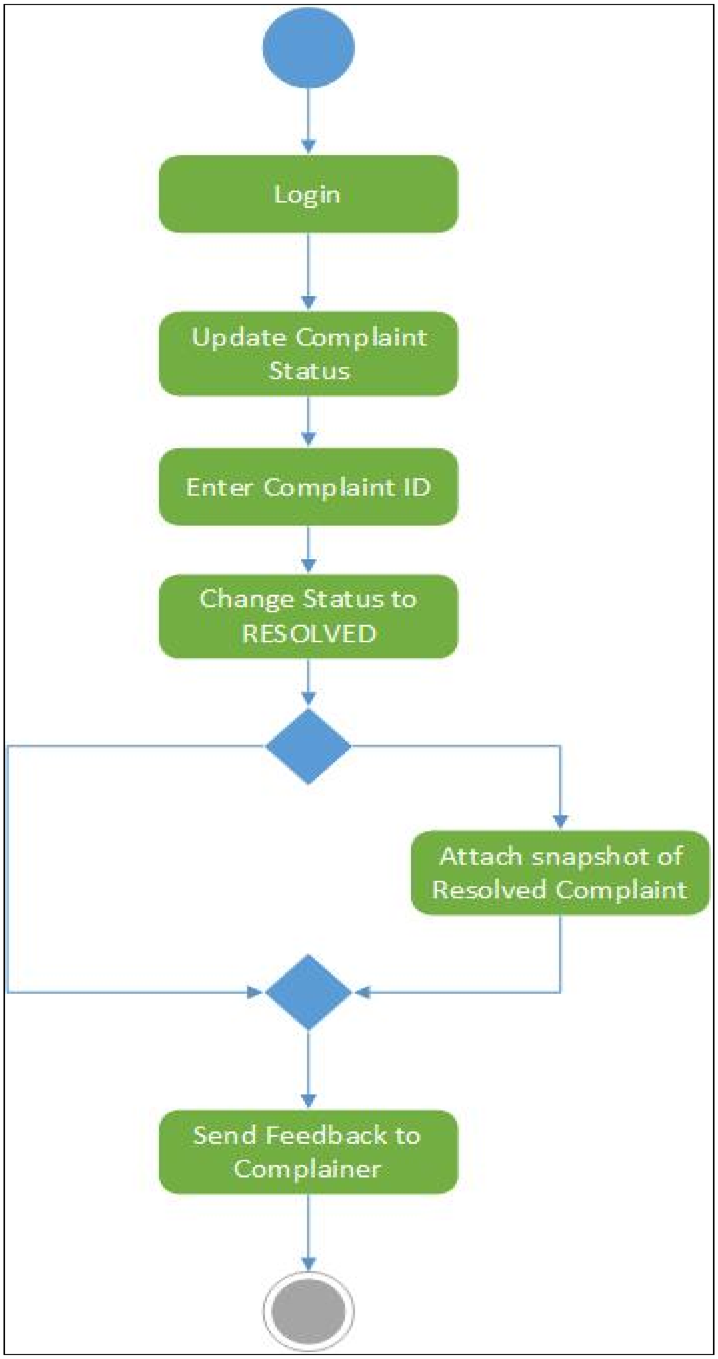

Login
Update Complaint Status
Enter Complaint ID
Change Status to RESOLVED
Attach snapshot of Resolved Complaint
Send Feedback to Complainer

Figure 19 Update complaint status

5.7.5 Sharing on Social Media:

The Activity diagram following shows the flow of activities when a User needs to share any complaint's content on various Social media platforms

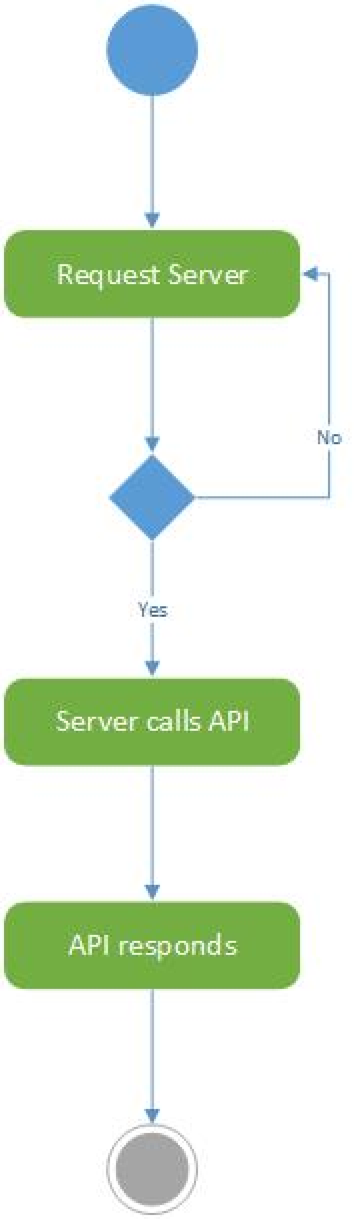


Figure 20 Social media sharing

5.7.6 Update complaint Status in Web Interface

The following diagram describes the stream of activities that a Concerned Staff needs to perform in a web user interface for updating the complaint status

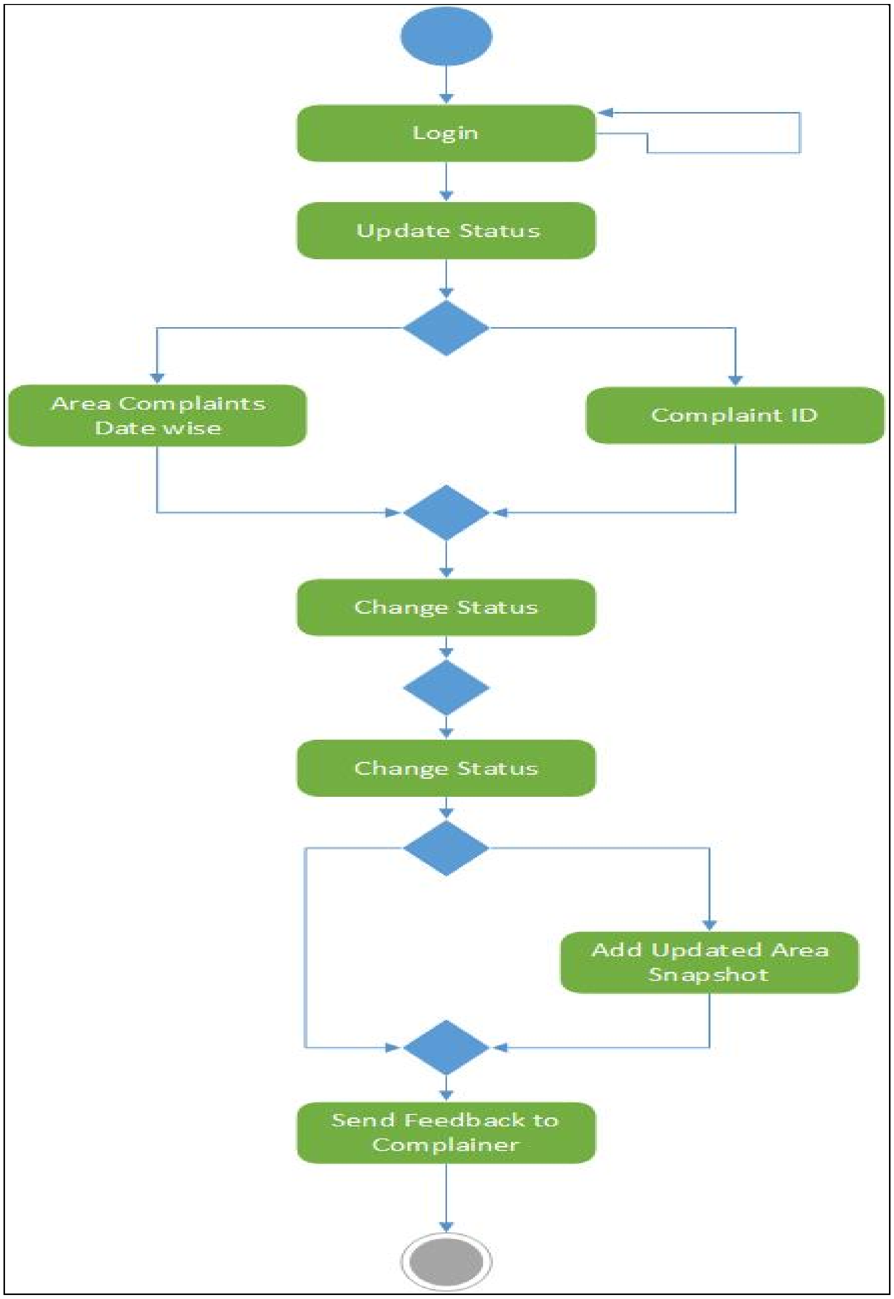

Login
Update Status
Area Complaints Date wise
Complaint ID
Change Status
Change Status
Add Updated Area Snapshot
Send Feedback to Complainer

Figure 21 Changing Status on Web

5.7.7 View Complaint for Admin

The following diagram describes the stream of activities that Admin wants to perform for viewing complaints with different options available in different formats.

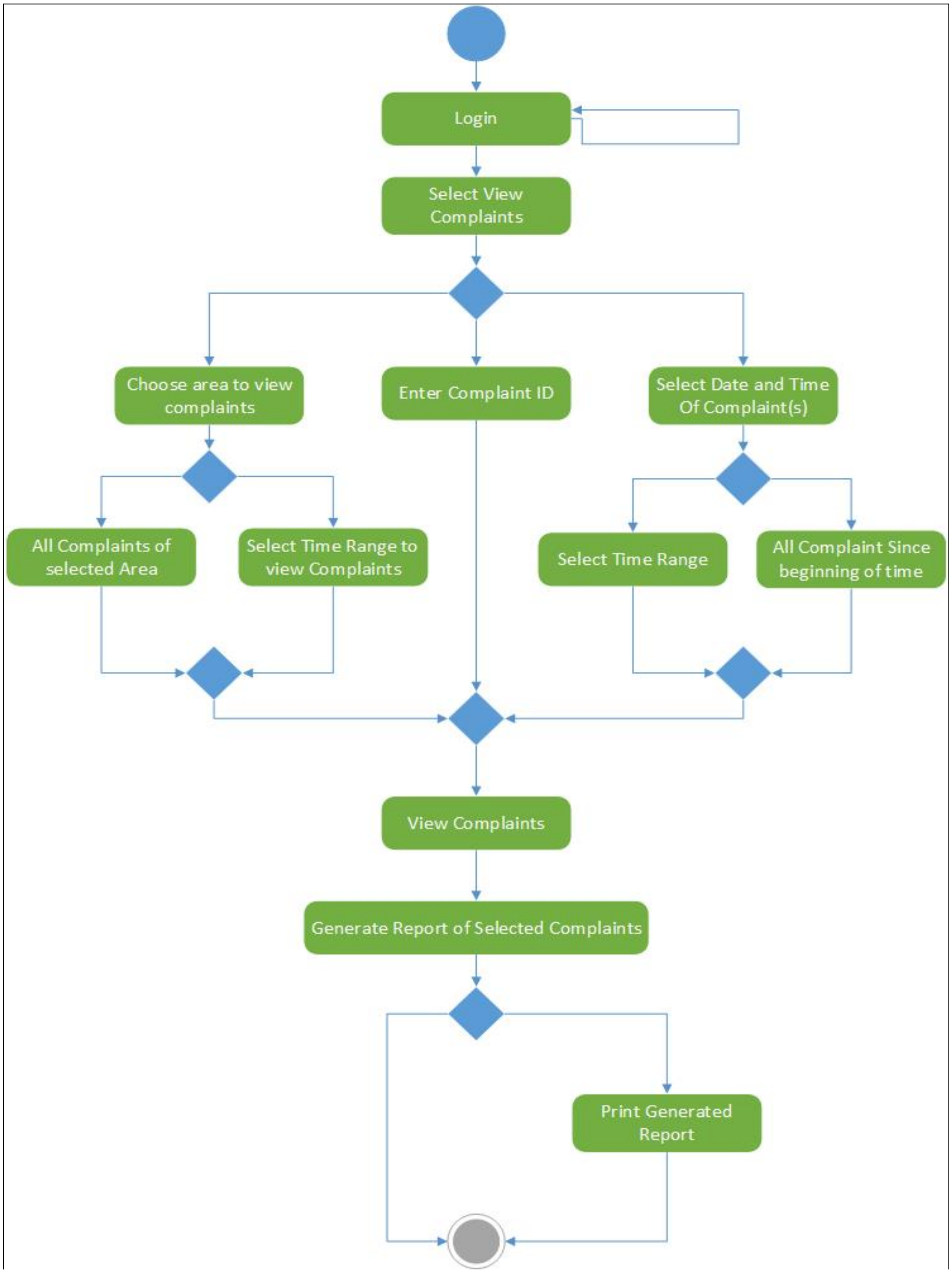

Login
Select View Complaints
Choose area to view complaints
Enter Complaint ID
Select Date and Time Of Complaint(s)
All Complaints of selected Area
Select Time Range to view Complaints
Select Time Range
All Complaint Since beginning of time
View Complaints
Generate Report of Selected Complaints
Print Generated Report

Figure 22View Complaints for Admin

5.7.8 Web Interface complaint view

The following diagram describes the stream of activities that a user need to implement to view complaints in web user interface. It requires login and it shows lot many options of viewing the complaints.

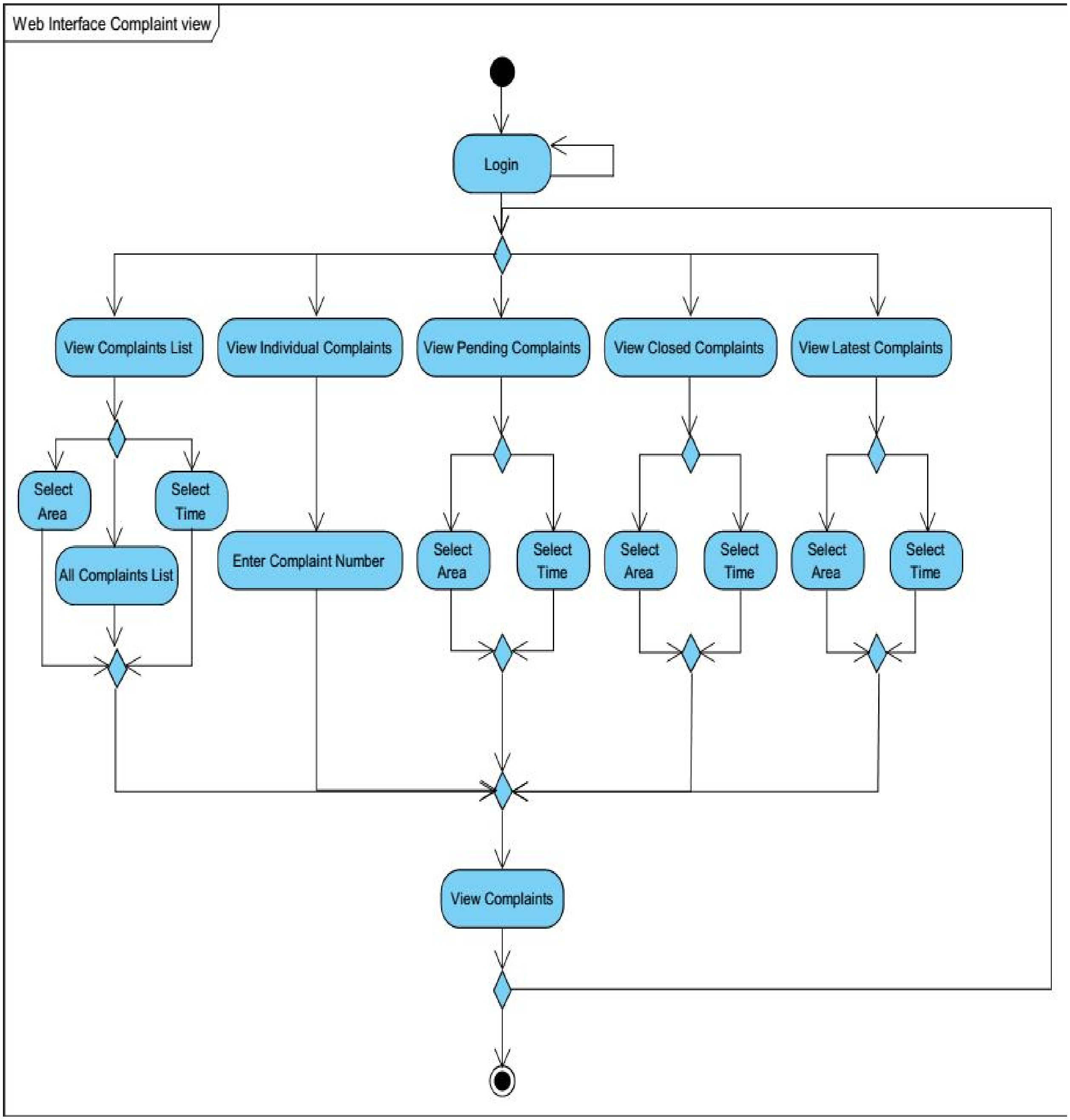


Figure 23 Web Interface view complaints

5.7.9 View Complaint for Complainer

The following diagram describes the stream of activities that a complainer wants to execute to view his/her complaint status by entering complaint number.

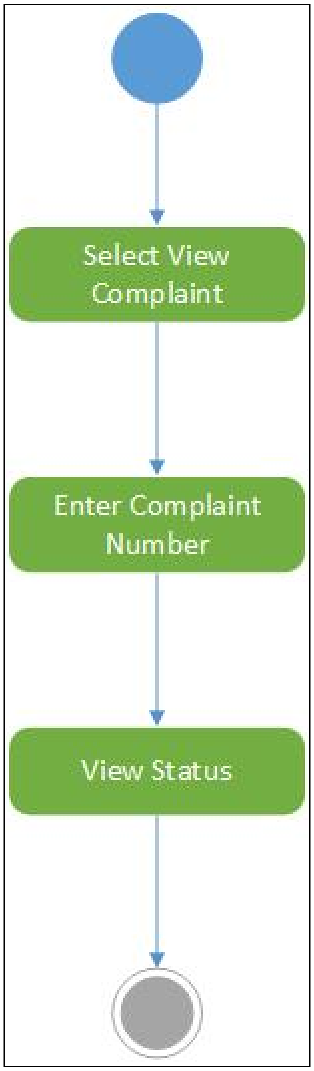


Figure 24 View Complaint for Complainer

5.7.10 View complaint concerned staff

The following diagram describes the stream of activities that a concerned Staff wants to accomplish to view Complaints of his own domain.

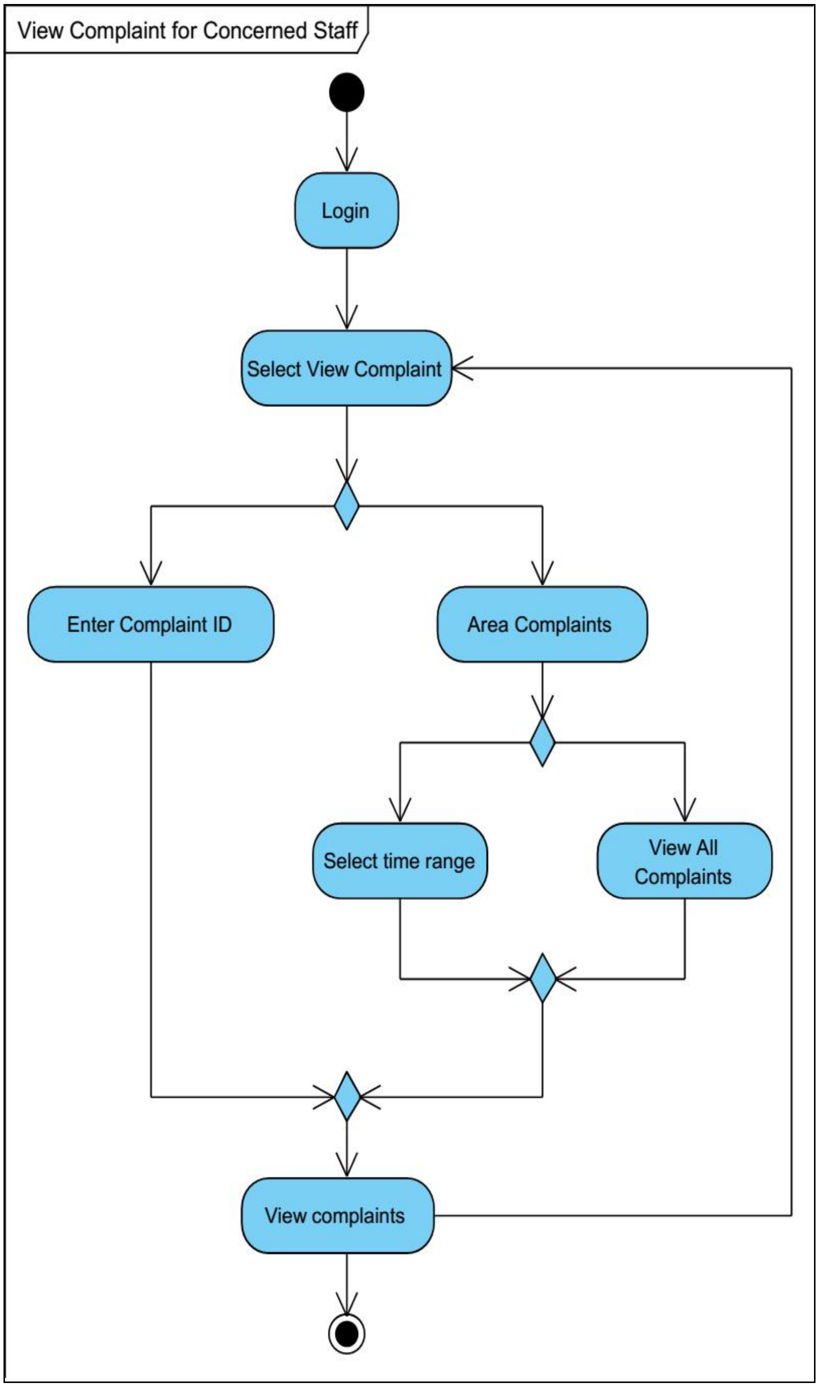


Figure 25 View Complaints concerned Staff

Chapter 6

6 System Implementation

6.1 Technology Used

6.1.1 Programming Language Used

The Web Interface of the system was constructed using Grocery Crud and MVC framework, with PHP used as a side scripting language. MySQL was used for handling all the data that was to be stored by the system. The android application for the project was written using Java and XML.

6.1.2 Development Tools

Android Studio was used for building the Android application, whereas Jet Brains Php Storm were used in the development of the Web Interface.

6.1.3 Database

The systems Database was established and achieved using MySQL

6.1.4 Operating System

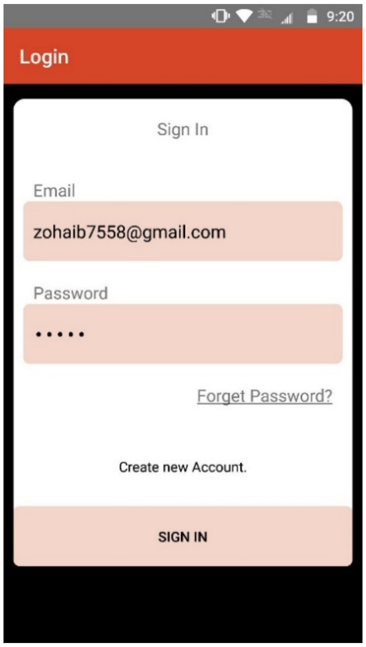
The Web Interface shall be running on Microsoft Windows and can run on Windows XP and all future iterations of it. The Android application shall be executing on Android, and works on versions later than 5.0 KitKat.

6.2 Complete System Implementation

The system comprises of two main components. An android application which the users will practice to lodge complaints and a web interface used by the Admin to reply to/ accept complaints. The major modules of The Android Application will be discussed in detail in the succeeding sections.

6.2.1 Login Module

Module is linked with the database and forms the data access layer of the application. Model Layer consists of all the functions that are then accessed from the view. Usernames and Passwords stored in the data base are compared with Username and Password entered by the user. Also, a new account can be created by choosing sign up.



Email
zohaib7558@gmail.com

Password
•••••

Figure 26 Login Screen

6.2.2 Home Module

This is the main module, which greets the user following successful login. This is used to access all the different functionality of the application and can be considered as a main menu. It can be used to Lodge complaints, View Complaints lodged by the user and to logout.

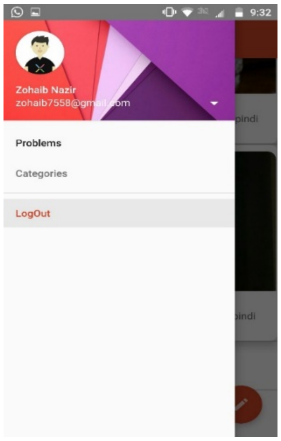
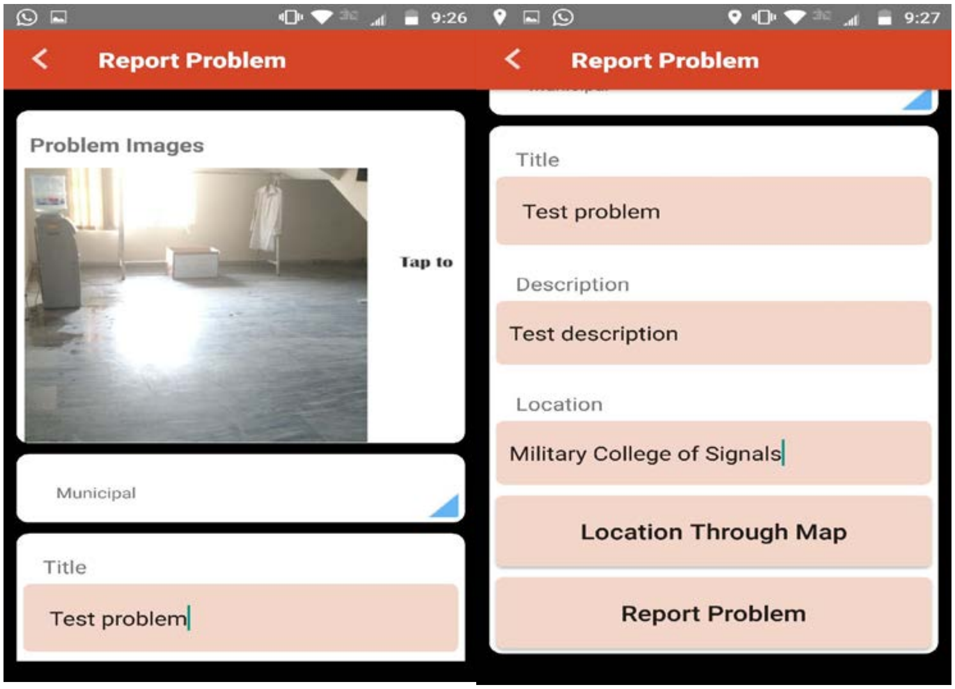


Figure 27 Home Screen

6.2.3 Lodge Complaint Module

This is the module which comprises all the functionality concerning to lodging of a complaint, including acquiring GPS location data and permitting adding a picture to the complaint. Location data can also be entered manually. All these particulars are stored in the database, and a unique tracking Id for the complaint is delivered to the user.



Municipal

Title

Test problem

Title

Test problem

Description

Test description

Location

Military College of Signals

Figure 28 Complaint Lodging Screen

6.2.4 View Complaints Module

This module allows logged in users to view the complaints that they have lodged and view the development that has taken place for those specific complaints. It does so by seeing the tracking Id’s connected with the account currently logged in to the application, then shows these. Upon selecting one, the details of that complaint are showed.

Chapter 7

7 System Implementation

7.1 Overview

Testing of software projects include different levels of testing to ensure that the software which is being developed is error and fault free. The different levels at which testing was performed is argued here.

7.1 Unit Testing

It includes the testing of each module at completion.

7.1.1 Login Feature Testing

Test Case Name:	Application Login Feature testing
Test Case ID:	1
Description:	This feature requests the user to enter his/her credentials for login. If not already signed up, user can sign up. This test case aimed to checked that feature works conferring to user requirement
Testing technique used:	Black Box Testing
Preconditions:	System is running and linked to database.
Input values:	Username String Password string
Valid Inputs:	Valid or legal username Valid or legal password

Steps	1. Enter Email 2. Enter password 3. Tap SIGN IN button
Expected Output	The user identifications will be passed to the server for verification. The valid users will be directed to home screen after login.
Actual Output	Successful login. User is directed to home screen.
Status	PASS

Table 13 Login Feature Testing

7.1.2 Sign up feature Testing

Test Case Name:	Sign Up Feature Testing
Test Case ID:	2
Description:	This feature requests the user to provide his/her credentials for signup. This test case is used to check the registration of a new user.
Testing technique used:	Black Box Testing
Preconditions:	System is executing and linked to database.
Input values:	First Name

	Last Name Email Password
Valid Inputs:	Alphanumeric First Name Alphanumeric Last Name Email id in the format: abc@xyz.com Alphanumeric password
Steps	1. Enter First Name 2. Enter Last Name 3. Enter Email 4. Enter Password 5. Tap SIGN UP button
Expected Output	User data is kept in the database.
Actual Output	User data is kept in the database.
Status	PASS

Table 14 Sign up Feature Testing

7.1.3 Data Insertion Testing

Test Case Name:	Report problem

Test Case ID:	3
Description:	This feature lets the user to enter the problem in the database.
Testing technique used:	Black Box Testing
Preconditions:	System is running, user is logged in and linked to database.
Input values:	Image File Category Selection from database Title of problem Description of problem Location
Valid Inputs:	.png/.jpeg image Alphanumeric characters title Alphanumeric characters explanation GPS coordinates of Location
Steps	1. Add a Photo 2. Select a Category 3. Enter Tittle 4. Enter Description 5. Input Location 6. Tap REPORT PROBLEM
Expected Output	User data is kept in the database.

Actual Output	User data is kept in the database.
Status	PASS

Table 15 Data Insertion Testing

7.1.4 Social Media Sharing Testing

Test Case Name:	Social Media Sharing feature Testing
Test Case ID:	6
Description:	This feature allows the user to share a Problem on social media. This test case is used to enquire the sharing of a Problem on social media (E.g. Facebook, Twitter).
Testing technique used:	Black Box Testing
Preconditions:	System is running and linked to database. Minimum one problem is added.
Input values:	Problem
Valid Inputs:	Valid Problem parameters

Steps	1. Tap the Share Icon 2. Choose the Social media platform 3. Write Post 4. Tap Share
Expected Output	Problem is shared on Social Media.
Actual Output	Problem is shared on Social Media.
Status	PASS

Table 16 Social Media Sharing Testing

Chapter 8

8 Conclusion and Future Work

8.1 Conclusion

Our team set out to develop a system that would find an innovative and creative solution to a problem many of us suffer daily. We did so by establishing a system that allows common people to lodge complaints in a formal manner, while allowing the concerned authorities to accept these complaints and respond to them in a methodical and ordered way.

We accomplished our objectives, successfully developing an Android Application that lets users to lodge complaints wherever they are, whenever they want as long as they have a data connection, alongside a web interface that allows respective authorities to receive these complaints and respond to them accordingly while updating the progress for the complainer to view.

Due to constraints of time and team size, the scope of the project was kept small. However, in the future flag markings can also be done on the maps as to have an analysis of which areas are hit by what kinds of different problems.

We firmly believe that our project can genuinely bring about a significant change in the lives of people.

8.2 Future work

Due to certain intrinsic limits in terms of project development time and team size, a lot of things had to be omitted from the scope of this project. However, this leaves room for a horde of enhancements, expansions and functionality add-on’s.

First of all, at the moment the Project only caters for any specific town management authority like Bahria Town or Green valley, therefore the user base for the system will be extremely limited. In the future however, functionality could be expanded to include all major cities of Pakistan.

At the moment, the system only caters for limited categories i.e. road works, sanitation, security and medical issues. In the future this could be extended to other fields, such as reporting electricity theft, gas problems etc. The program might be expanded to entertain complaints of other kinds, for example a similar system could be developed for the police for users to report criminal/suspicious activity. A system could be developed for the Public Transport agency to register complaints with railway, busses etc.

Currently, the application only works on Android devices, further limiting its potential user base. In the future, an application can be developed for iOS, Windows Mobile, Ubuntu Mobile and

Firefox OS. This would enable smartphone users on all major platforms to take advantage of the application.

Among minor changes, the application's User Interface could be modified to be even more user friendly and the application could be improved to run faster and enhance performance on lower end devices.

Glossary

API	Application Programming Interface
App	Application
AS	Assumption
Black box Testing	Testing emphasizes on the external behaviour of the software entity
CO	Constraints
App	Application
CEO	Chief Executive Officer
DBMS	Database Management System
DEP	Dependency
FRs	Functional Requirements
GUI	Graphical User Interface
HTML	Hyper Text Markup Language
HTTP	Hypertext Transfer Protocol
IDE	Integrated Development Environment

iOS	Mobile Operating System created and developed by Apple
JavaScript	Client side scripting language used to create dynamic web content and user interface
MCS	Military College of Signals
NFRs	Non Functional Requirements
NUST	National University of Science and Technology
OE	Operating Environment
OS	Operating System
Parse	Cloud Server
REQ	Requirement
SQL	Structured Query Language
SR	Safety Requirements
SRS	Software Requirements Specification
UD	User Documentation
UML	Unified Modelling Language
White Box Testing	Testing emphasizes on the internal behaviour of the software entity

Table 17 Glossary

Bibliography

https://www.coursera.org/

https://www.quora.org/

https://developer.android.com

https://bytehost.com

https://www.w3schools.com

https://www.teachmephp.com