

---

# **Software Requirements Specification**

**For**  
**Society Complaint Portal**

**Prepared by:**

Hamza Maqsood (BSEF19M519)  
Zainab Tahir (BSEF19M522)  
Amina Ansar (BSEF19M527)  
Ahmed Raza (BSEF19M537)  
Danial Tahir (BSEF19M542)

**Submitted to:**

Prof. Junaid Nasir

**Submission Date:**

September 10, 2022

## Contents

1.	Introduction.....	3
1.1	Purpose.....	3
1.2	Document Conventions .....	3
1.3	Intended Audience and Reading Suggestions.....	4
1.4	Product Scope.....	4
1.5	References.....	5
2.	Overall Description.....	5
2.1	Product Perspective .....	5
2.2	Product Functions.....	7
2.3	User Classes and Characteristics .....	8
2.4	Operating Environment.....	8
2.5	Design and Implementation Constraints .....	9
2.6	User Documentation.....	9
2.7	Assumptions and Dependencies .....	9
3.	External Interface Requirements .....	11
3.1	User Interfaces:.....	11
3.2	Hardware Interfaces .....	19
3.3	Software Interfaces.....	19
3.4	Communications Interfaces .....	19
4.	System Features.....	20
4.1	User: Complaint Registration.....	20
4.1.1	Description and Priority .....	20
4.1.2	Stimulus/Response Sequences .....	20
4.1.3	Functional Requirements.....	20
4.2	User: Complaint Tracking .....	20
4.2.1	Description and Priority .....	20
4.2.2	Stimulus/Response Sequences .....	21
4.2.3	Functional Requirements.....	21
4.3	Admin: View and handle Complaints.....	21
4.3.1	Description and Priority .....	21

4.3.2 Stimulus/Response Sequences .....	21
4.3.3 Functional Requirements.....	22
4.4 Admin: Add User .....	23
4.4.1 Description and Priority .....	23
4.4.2 Stimulus/Response Sequences .....	23
4.4.3 Functional Requirements.....	23
4.5 Admin: Update / Delete user .....	24
4.5.1 Description and Priority .....	24
4.5.2 Stimulus/Response Sequences .....	24
4.5.3 Functional Requirements.....	24
4.6 Super Admin: View Complaints and monitor admin .....	25
4.6.1 Description and Priority .....	25
4.6.2 Stimulus/Response Sequences .....	25
4.6.3 Functional Requirements.....	25
5. Other Nonfunctional Requirements .....	26
5.1 Performance Requirements.....	26
5.2 Safety Requirements.....	26
5.3 Security Requirements.....	26
5.4 Software Quality Attributes.....	26
5.5 Business Rules .....	27
6. Other Requirements .....	27
Appendix A: Glossary.....	28
Appendix B: Analysis Models.....	29
Appendix C: To Be Determined List .....	30

# 1. *Introduction*

## 1.1 Purpose

- The purpose of building a web application named “**Society Complaint Portal**” is to manage the society's complaints online and establish peace in society.
- The idea is to manage records online and avoid manual “panchayat” systems.
- In the manual “panchayat” system, if someone complains against their neighbor, it will result in rivalry and things become more difficult.
- To avoid this behavior, the user who will register the complaint in the society complaint portal will be anonymous from complainee but not from the admin.
- The one who will create disturbance in society will be punished by the society admin according to the crime level.
- The admin will also face trouble, if a complaint will not be tackled within 7 days.

## 1.2 Document Conventions

SRS Conventions are:

- Simple Text with Arial font.
- Text font size is 11.
- Main headings (titles) are written in Arial with font size 23.
- Headings font family is Arial with font size 17.
- Product name “Society Complaint Portal” is bold and highlighted with orange color.
- To compare our product with the previous manual system, we use the word “panchayat” word.
- The term complainee used for the person to whom the complaint against.
- The term complainer is used for the person who registered the complaint.
- Functional requirements are written in bold.
- Product functions are written in bold.
- User classes are written in bold and uppercase letters.

## 1.3 Intended Audience and Reading Suggestions

This SRS document is for developers, project managers, designers, users, customers and for testers. The customer can review this document to ensure their needs along with the needs of their users are being met in the system. The development team will also use this document for guidance on overall design and implementation of the “Society Complaint Portal”. Clear and concise requirements will help the developers to build a proper product.

Programmers will use this SRS as a reference point to understand business needs and requirements. The test and verification team can reference this to ensure the requirements are being met for the customer. Testers will use this SRS and test the product on the basis of functionalities and requirements mentioned in it. Finally, the tech writer will use this to assist with user documentation. This document is designed to be reviewed from beginning to end.

## 1.4 Product Scope

The purpose of building “Society Complaint Portal” is to manage the society's complaints online rather than manually. Any person of society can register a valid complaint against a person who creates disturbance in society. For verification of genuine complaints, complainer will add the image or video as evidence. Only one society can use this software at a time.

The goal is to build software efficient enough to handle complaints online without any inconsistency. Our goal is that by using this software resident can register complaints, track complaints, get a penalty when registering a false complaint, admin tackle complaints within a specified period and in case of delay admin get a penalty from super admin.

### Objectives:

- Register complaint.
- View Complaint.
- Anonymous complainer.
- Maintain complaint records online.
- Notify/warning to the complainer and complainee.
- Verification of complaint.
- Penalty to complainee in case of false complaint.
- Tackle complaint within 7 days.
- Penalty to admin in case of delay.

- Track complaints.
- Admin can add, delete, update and view members.

### Benefits:

- Rid from manual “panchayat system”.
- Lower risk of data lost.
- Verification of complaints.
- Lower chance of favoritism.
- Peace in society.
- End of rivalry

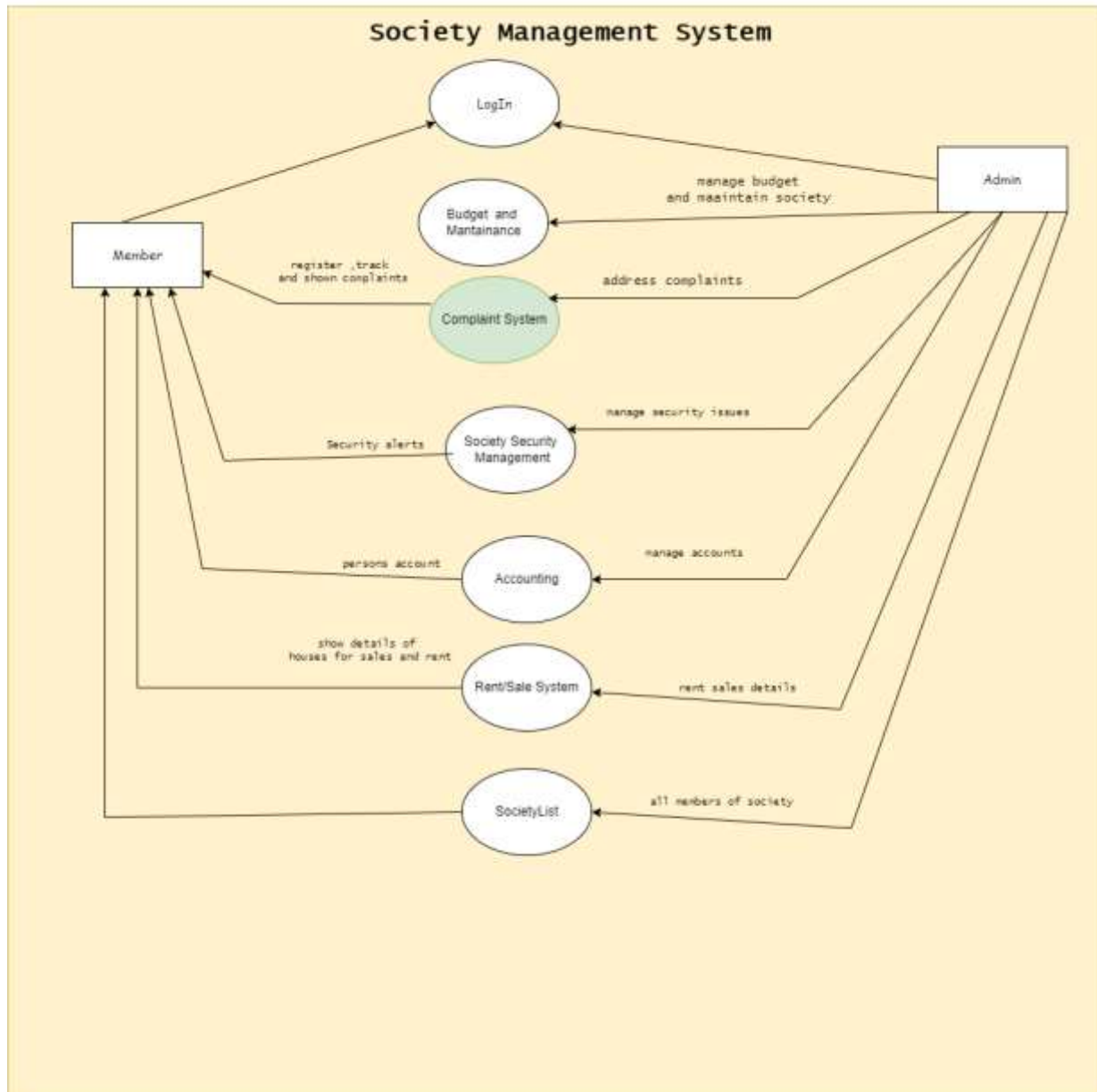
## 1.5 References

While writing the SRS of Society Complaint Portal, we didn't copy from any related SRS or from any website. The idea of interface design is our own. There is nothing copied from Google.

# 2. Overall Description

## 2.1 Product Perspective

Our “**Society Complaint Portal**” is a new project that is not a follow up of any project. It belongs to a family of Management systems. It isn't a replacement but an automated idea because this is still done manually today but with the revolution of the modern age, we now need an automated society portal. Your complaints will be taken by the software and the admin will handle this and you will be notified automatically. You can track your complaints. Our Project can be used by one society at a time. Our Product stakeholders are **Complainee, Complainer, Admin, Super Admin**. Complainer can complain, complainee will get notification, admin can handle the complaints and super admin may give penalties to admin. This is basically a module of a larger project Society Management system. We are not discussing every module but only the module that is our project- will handle complaints. The larger module will consist of all the information about society, its residents, sales, rents and workers. A simple diagrams of the larger module is discussed below:



## 2.2 Product Functions

Our **efficient** product “**Society Complaint Portal**” Product will provide a number of functions. They are listed below:

- It **maintains all records** of information of the society members.
- It shows an option of **Register Complaint** to society members.
- It shows an option of **View Complaint** to society members.
- While registering a complaint it shows a warning bar that in case you are registering a **wrong complaint**, then you will have to **pay the penalty** of false accusation.
- In registering a complaint the system will demand a proof, complaint description, name and address of the complaine.
- Product will give a **notification to the complainer** of penalty if complaints prove false.
- Product will give a **notification to the complaine** along with an email.
- Product will **maintain members' details** on Admin side for admin.
- It allows the admin to **add members' information**.
- It allows the admin to **delete members' information**.
- It allows the admin to **update members' information**.
- It allows the admin to **search members**.
- Incase admin doesn't handle the complaint within seven days, the product will notify him/her with a **notification** that Admin must pay a penalty now.
- Product **shows all complaints to admin** that are issued, that are handled and pending complaints.
- Product will provide an option to **super Admin to view complaints**.
- Product will allow **superAdmin to add Admins**.
- Using the product, the super admin can check if he wants to **select penalties for admin**.
- Product will allow the complainer to **track his/her complaints**.
- It allows the **admin to verify the complaints** in order to identify fake complaints.
- It allows the **admin to change the status of the complaint** and to save it along, as the actions on the complaint are taken.



## 2.3 User Classes and Characteristics

User classes for “Society Complaint Portal” are **ADMIN, SUPER ADMIN, COMPLAINER, COMPLAINEE**. Admin must be a technical person that knows how to handle the system. Admin will handle security issues instantly and will ensure the safety and privacy of complainee and complainer. All users must know the technology and websites. They should know how to use this web application.

The Complainer must know whether his/her complaints are addressed or not and the complainee must use his / her email or portal to see notification/ message. Complaints and complainee should also know how to use this software. Members should have a good understanding of how to use this software and at least one member of each member should know this and must have internet, mobile/ laptop to use it.

## 2.4 Operating Environment

- This is a web application .It can operate on Android, IOS, Windows, Mac and Linux.
- There will be two sides: client side and server side.
- All users should have efficient mobiles and PCs with core i5 having at least 8GB of RAM and at least a 14” screen.
- On client side, users should have a Browser with SSL(Secure Socket Layer) cryptographic protocols at a minimum encryption level of 128 bits. Browser can be:
  - Google chrome 40+
  - Firefox 40+
  - Opera
  - Internet Explorer 9+
- On server side, MERN stack technology is used
  - MangoDb
  - Express
  - React
  - Nodejs

## 2.5 Design and Implementation Constraints

- There should be enough memory to handle records of each and every society member.
- MongoDB is not resilient against crashes so you have to take care of this.
- Developer of this software should use MERN technology.
- Products should be easy to use by a common person.
- You should check compatibility of browsers while developing the product.
- This is a web application and will not be in working condition without javascript support.
- Security will be handled by admin not by super admin or developer.

## 2.6 User Documentation

Our target audience is members of the society, admin and super admin. We'll provide them with:

- User Manual that will help all the users to understand how they should use the product.
- Shall provide a website link so that they can use that link to access the portal.
- A tutorial link will also be given to all users.
- Online Help option is available also.
- A special guide for admin that includes how to install the system and maintain its security.

## 2.7 Assumptions and Dependencies

We are assuming that:

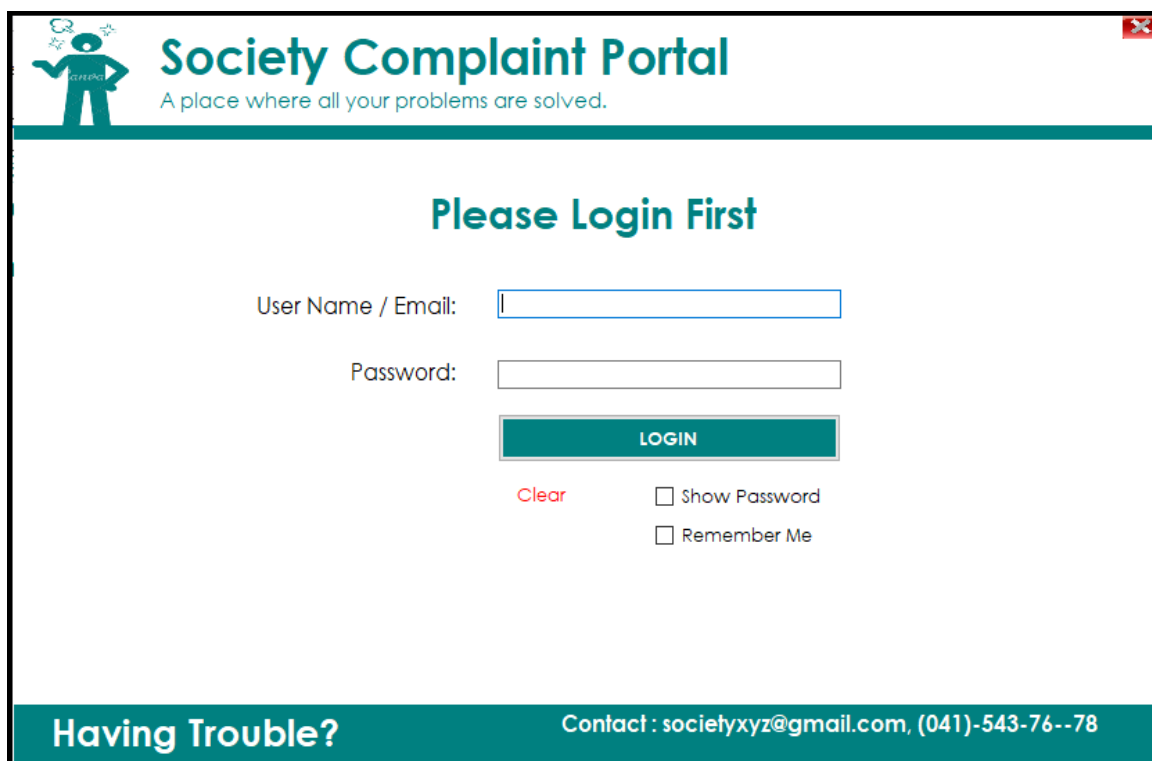
- All the members of the society are well educated and know how to use "Society Complaint Portal".
- Admin and super admin also belong to the same society.
- All members and users of this software have 24 hrs internet access.

- There is only one admin we are assuming and only one super admin.
- SignUp (i-e members data) is already hard coded when any new member comes into the society.
- This software will only have login.
- Admin or super admin cannot be a complainer or complaine.
- Portal will work efficiently and never be down.
- It will work on any browser.
- Complainer will be anonymous so that there will be no chance of rivalry.
- Complaints can be registered any time ,any hour.
- Username and email of all users will always be unique.
- All the databases and servers for the application are fully functional and efficient.

## 3. External Interface Requirements

### 3.1 User Interfaces:

- Login:



The screenshot displays the login interface of the 'Society Complaint Portal'. The header features a logo on the left and the portal's name and tagline on the right. The main content area prompts the user to login, providing input fields for username/email and password, a login button, and options to clear the form, show the password, or remember the user. The footer contains a link for users having trouble and contact information.

**Society Complaint Portal**  
A place where all your problems are solved.

**Please Login First**

User Name / Email:

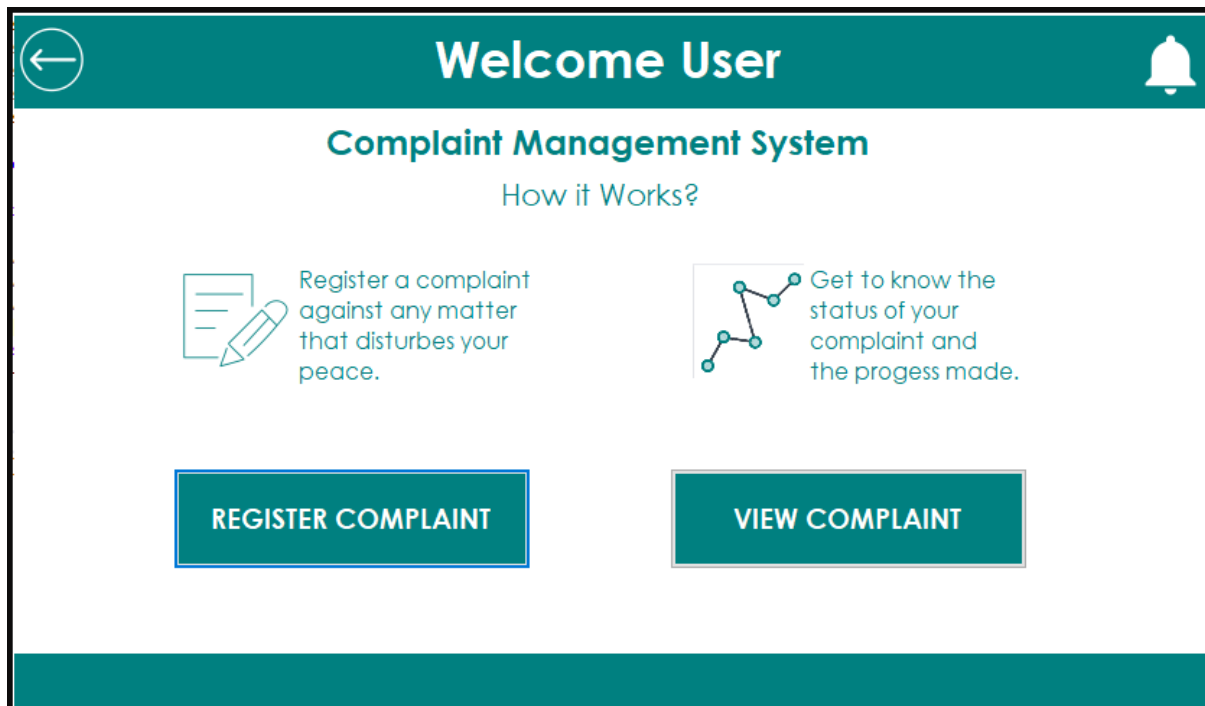
Password:

**LOGIN**

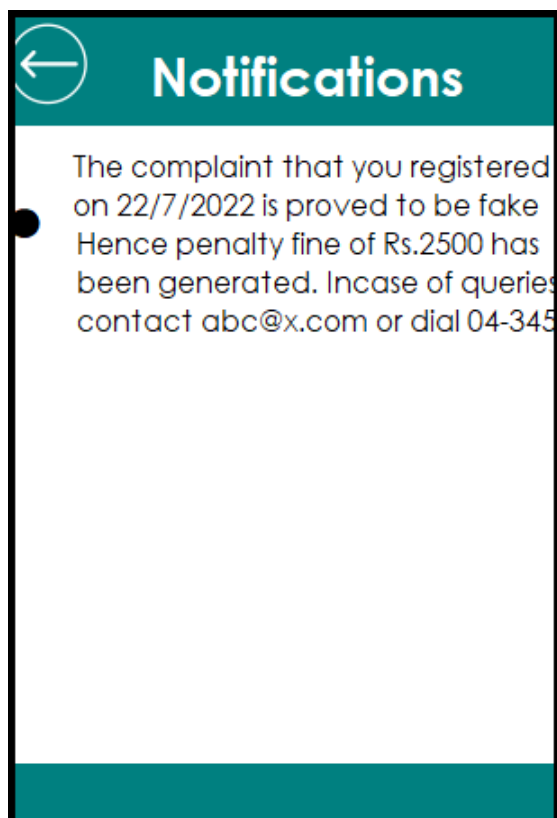
[Clear](#) ☐ Show Password ☐ Remember Me

**Having Trouble?** Contact : societyxyz@gmail.com, (041)-543-76--78

- **User Dashboard:**



- **Notification Dashboard (User) :**



- **Register Complaint:**




## Register Complaint

Fill this form to register a complaint (Mention the person's details you are reporting against)\*

Name:

Address:

Description:


Photo/Video Proof:  / 

**Register Complaint**

**Warning:** Fake Complaints can lead to penalties.


- **Warning :**

Confirm Submission ×

 Submitting FAKE complaints can lead to penalties

**OK** Cancel


- **View & Track Complaint:**




## Complaint Details and Status



	ID	Name	Date	Description	Status
»					

- **Admin Dashboard:**



## Welcome Admin

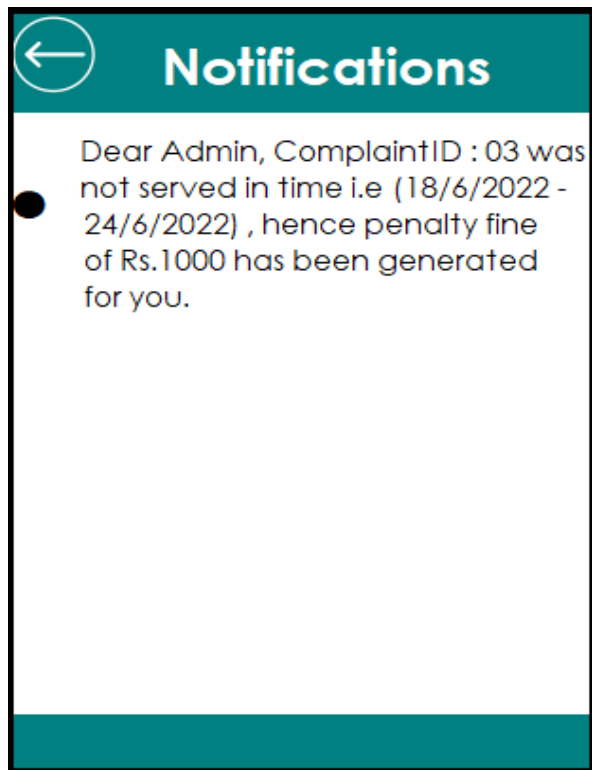




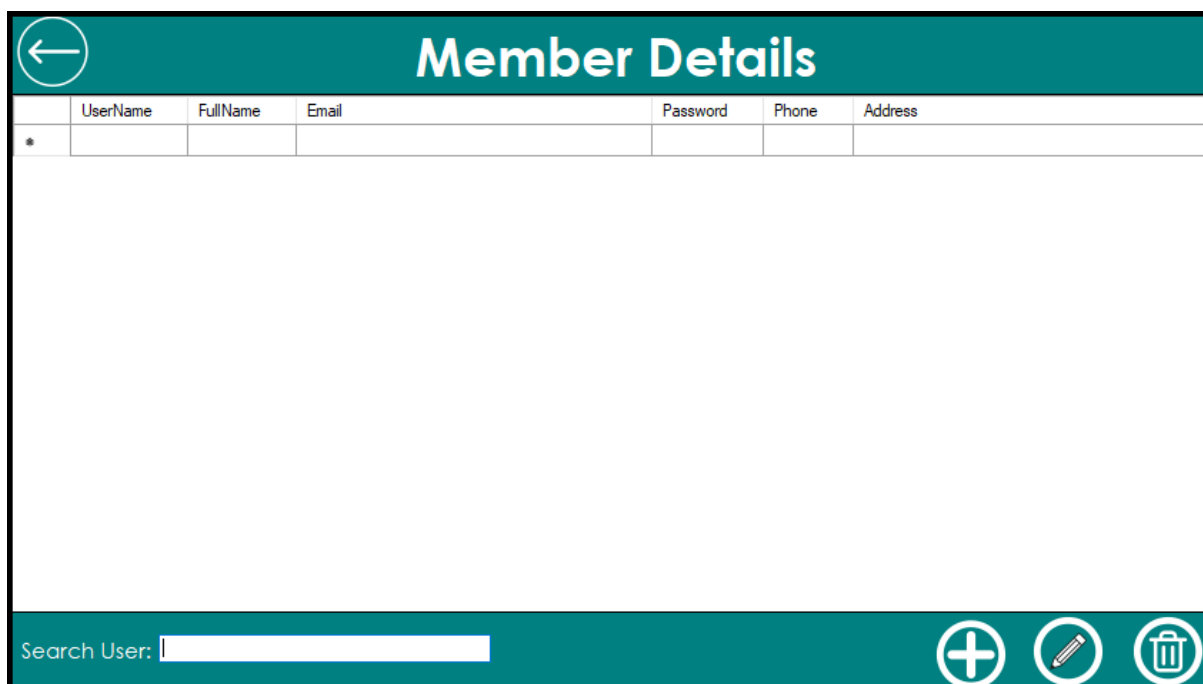
COMPLAINTS

MEMBERS

- **Notification Dashboard (Admin) :**




- **Members ( Users ) :**





- **Add Member (Member Module) :**

 **Member Module**

User Name:

Full Name:

Email:


Password:

Confirm Password:

Phone:

Address:

- **Update Member:**

 **Member Module**

User Name:

Full Name:

Email:


Password:

Confirm Password:

Phone:

Address:


- **Delete Member:**





**Member Module**

User Name:




**Delete**

- **Complaints:**


**Complaints**

	ID	Name	Date	Description	Address	Proof	Verification	Status	Save Changes
*									

Search Complaint:

Take Action:   

- **Complaint Action Form:**

**Complaint Action Form**

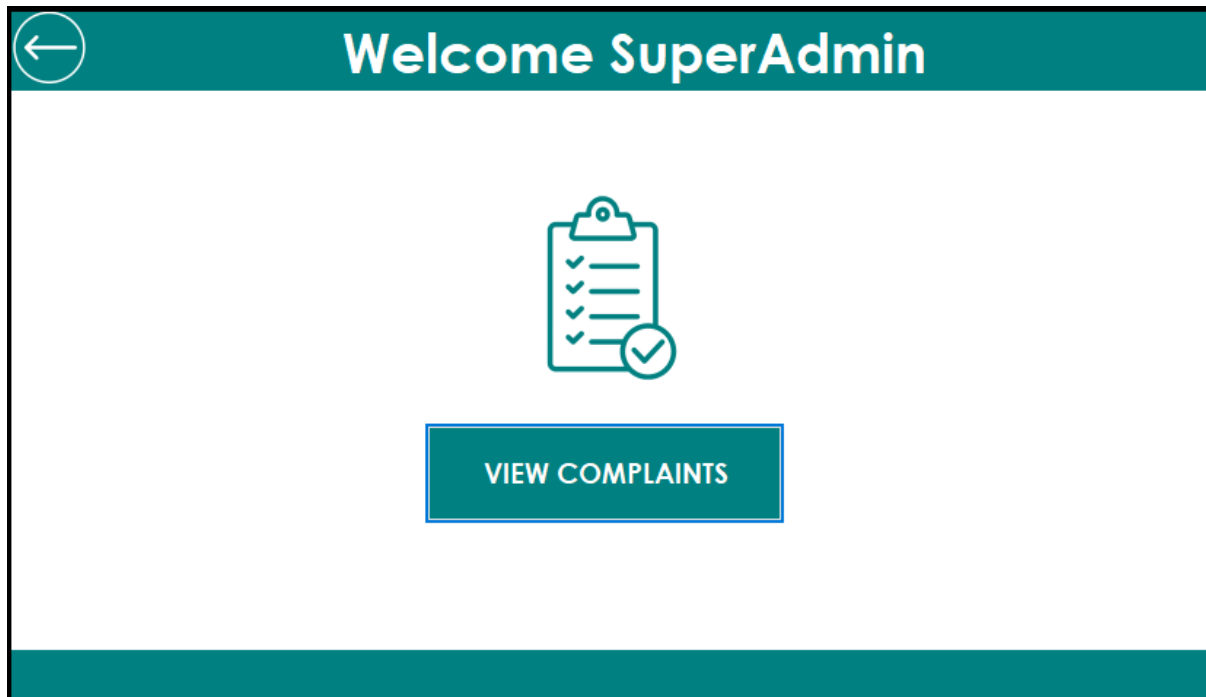
User Name:

Email:

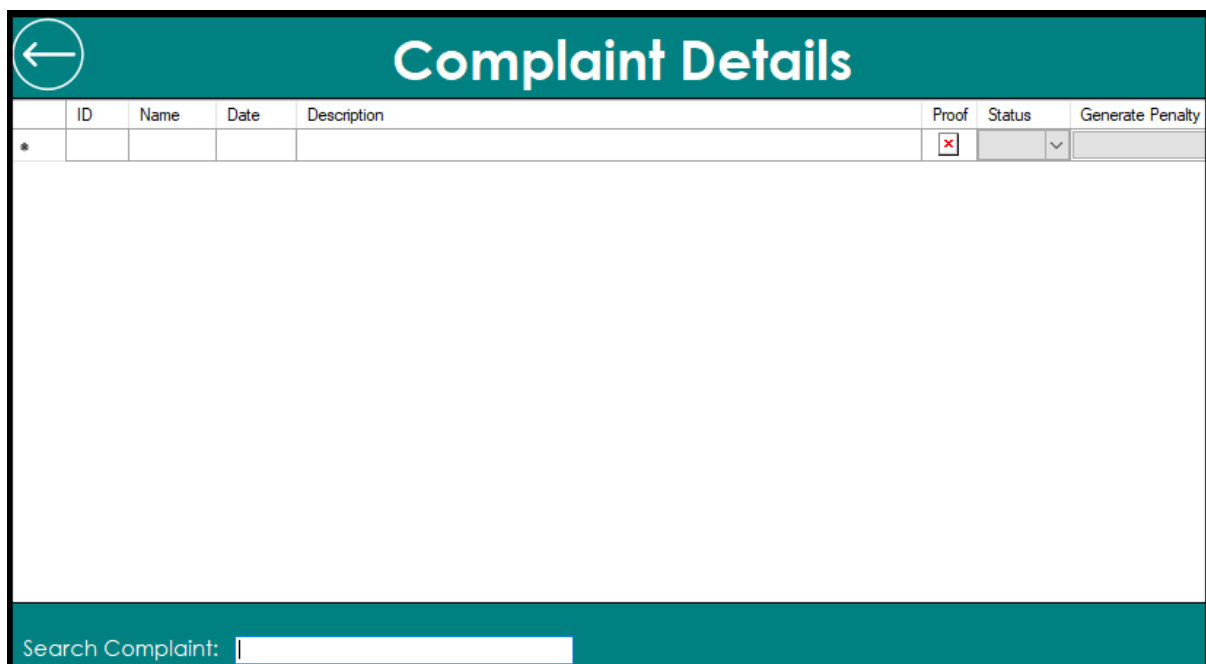
Complaint Details:

**Generate**



- **SuperAdmin Dashboard:**



- **Complaint Details:**



The Complaint Details UI features a teal header bar with a back arrow icon on the left and the text "Complaint Details" in white. Below the header is a table with the following columns: ID, Name, Date, Description, Proof, Status, and Generate Penalty. The table has one row with a red 'x' in the Proof column, a dropdown arrow in the Status column, and a text input field in the Generate Penalty column. Below the table is a large white area for details. At the bottom is a teal bar with the text "Search Complaint:" followed by a white text input field.

ID	Name	Date	Description	Proof	Status	Generate Penalty
*						<input type="text"/>

Search Complaint:

## 3.2 Hardware Interfaces

- All users should have efficient mobiles and PCs with core i5 having at least 8GB of RAM and at least a 14" screen.
- The product will only recognize the user if correct input ( email/username and password) are entered through the keyboard.
- The product will only run on mobile devices having android version greater than 6.0 and IOS greater than 13.0.

## 3.3 Software Interfaces

- This is a web application .It can operate on Android, IOS, Windows, Mac and Linux.
- On client side, users should have a Browser with SSL(Secure Socket Layer) cryptographic protocols at a minimum encryption level of 128 bits. Browser can be:
  - Google chrome 40+
  - Firefox 40+
  - Opera
  - Internet Explorer 9+
- On server side,MERN stack technology is used
  - MangoDb
  - Express
  - React
  - Nodejs

## 3.4 Communications Interfaces

- The Network communication server protocol used in this product is HTTPS.
- The maximum file size that you can upload as a proof is 100 Mb.
- You can upload the picture/video from your google drive.
- Email should always be in the format xxxxxxxxx@xxxxx.com.

## 4. System Features

### 4.1 User: Complaint Registration

#### 4.1.1 Description and Priority

- Every user can register complaints against other residents of society. The complaint will be forwarded to the admin. The complaint should be provided with proof (image or video). If a user registers a fake complaint, the admin can penalize him / her. The requirement has **high priority**.

#### 4.1.2 Stimulus/Response Sequences

- Users will login.
- System will give access to the user account.
- User will add the name, address of the complainant along with description and a proof (image/video clip).
- System will store complaints in the database.

#### 4.1.3 Functional Requirements

- **System should have the user's account in the database.**
- **System shall produce a session for the user while logging in.**
- **System shall receive complaint data from the user.**
- **System will process complaint data and store it into a database table.**

### 4.2 User: Complaint Tracking

#### 4.2.1 Description and Priority

- Users can track their complaints after registering it. They can check the status of their complaint; either complaint is pending, acknowledged, accepted or rejected. If a complaint is rejected, the user will get a penalty.

### 4.2.2 Stimulus/Response Sequences

- Users will login.
- System will give access to the user account.
- User will click on “View Complaint”
- System will retrieve data from the database and show all complaints.
- Users can view their complaints and the statuses.
- 

### 4.2.3 Functional Requirements

- **System should have the user's account in the database.**
- **System shall produce a session for the user while logging in.**
- **System will provide complaint data to the user by accessing the database.**
- **System will produce a view and inflate data.**

## 4.3 Admin: View and handle Complaints

### 4.3.1 Description and Priority

- Admin can see all complaints which are registered new or old. They will receive all information of complainer and complaine. They can change status of complaint from pending to ‘acknowledged’ or ‘under process’. They can then see proof (video or image) for verification. Based on proof or other factors (may require manual verification) he will decide fate of complaint. Admin will send notification to complainer and complaine about the complaint. If it is authentic complaint, he will change status of complaint to accepted, give penalty to complaine by sending email and notification. The identity of complainer will be kept confidential. If complaint is fake, the penalty will be given to complainer by sending notification and email to him. Moreover, the status of the complaint will be set to reject. The requirement has **high priority**.

### 4.3.2 Stimulus/Response Sequences

- Admin will login.
- System will give access to the admin account.

- Admin will click on “View Complaint”
- System will retrieve data from the database and show all complaints.
- Admin can view all complaints and can also filter to view pending complaints.
- Admin will change the status of the complaint from pending to “under process”.
- System will store a new state and send notification to the complainer about the change of status of their complaint.
- Once verification is done, admin will again change status of complaint to accepted or rejected.
- System will again save state and send notifications accordingly.
- Admin will penalize complainer or complaine.
- System will generate email and notification for it and send it to the complainer or complaine or both.

### 4.3.3 Functional Requirements

- **System should have an admin's account in the database.**
- **System shall produce a session for the user while logging in.**
- **System will provide complaint data to the admin by accessing the database.**
- **System will filter data on demand (i.e., new complaints or old).**
- **System will store changed states of complaint.**
- **System will notify the complainant or complaine about it.**
- **System will generate emails for complainer and complaine.**

## 4.4 Admin: Add User

### 4.4.1 Description and Priority

- Admin will add users when a new home is constructed in society. Residents will get their details registered to avail complaint facilities. Name, email, phone and address will be registered for every user. The requirement has **medium priority**.

### 4.4.2 Stimulus/Response Sequences

- Admin will login.
- System will give access to that account.
- Admin will click on “Add User”.
- System will redirect to the page.
- Admin will add all the details for user registration.
- System will validate data sent to the database.
- If data is invalid, System will generate an error.

### 4.4.3 Functional Requirements

- System should have an admin's account in the database.
- System shall produce a session for admin while logging in.
- System will authenticate data when the admin adds a user.
- System will produce error if email is invalid
- System will produce errors if the email is already registered.
- System will produce errors if passwords do not match or password length is less than 8 characters.
- System will provide errors if the phone is invalid.
- System will store data if a user is successfully created.
- System will send email to the user about account creation.



## **4.5 Admin: Update / Delete user**

### **4.5.1 Description and Priority**

- Admin may need to update information of residents if someone buys a house (residents changed). Admin may also need to delete the user's account if someone leaves the house (no resident anymore). Admin can update any information of that user from name, email, and phone to address. The requirement has medium priority.

### **4.5.2 Stimulus/Response Sequences**

- Admin will login.
- System will give access to the admin account.
- Admin will click on “update / delete users”
- System will retrieve data from the database and show all users.
- Admin can filter users.
- System will retrieve filtered data from the database.
- Admin can update any information or delete users.
- System will execute queries on the database.

### **4.5.3 Functional Requirements**

- **System should have an admin's account in the database.**
- **System shall produce a session for admin while logging in.**
- **System will provide user data to the admin by accessing the database.**
- **System will produce a view and inflate data.**
- **System will validate data before updating.**
- **System will send an email to the user about the update.**
- **System will delete users from the database.**

## **4.6 Super Admin: View Complaints and monitor admin**

### **4.6.1 Description and Priority**

- Someone is required to monitor admin. Our system will provide an interface for super admin. Super admin can view all complaints and their status. If the admin fails to address a complaint in a given time, the super admin will penalize him. If the admin gets a penalty, a notification and email will be sent to him.

### **4.6.2 Stimulus/Response Sequences**

- Super Admin will login.
- System will give access to the super admin account.
- Super Admin will click on "View Complaints"
- System will retrieve data from the database and show all complaints.
- Super Admin can filter complaints to see unaddressed ones.
- System will retrieve filtered data from the database.
- Super Admin can generate a penalty for admin.
- System will send notification to the admin.
- System will generate email and send it to the admin.

### **4.6.3 Functional Requirements**

- **System should have a super admin's account in the database.**
- **System shall produce a session for super admin while logging in.**
- **System will provide complaints to the admin by accessing the database.**
- **System will produce a view and inflate data.**
- **System will generate a penalty.**
- **System will send an email to the admin.**
- **System will generate a notification on the admin side.**

## 5. Other Nonfunctional Requirements

### 5.1 Performance Requirements

Whenever the member logs in, He / she can save their login details so that our system remembers them and they don't have to login again and again. We displayed a notification bar where all the warnings, notifications of society and complaint results will be shown. We put a warning message to avoid fake complaints. We added facility of tracking the complaints in a better way by checking the status of complain. We make sure that no one outside the scope of society login the system

### 5.2 Safety Requirements

The personal information of the user will be secured. All the details added by the members about him/her are only visible to the Admin and Super Admin .There is a proper abstraction between the complainer and the complaine is recorded. Images and Videos which are supposed to be used as proofs are maintained secure.

### 5.3 Security Requirements

The user can only login into his account by his/her username and password which will provide security to the user's account so that all the information of the user's account will remain secure. User's login is secure; no other person can check his complaint history etc. It is a security requirement for both the software and the user that he will be notified about the warning that if the complaint is fake or scam, serious notice will be taken. Login ID and Password will be provided by the Admin to the verified members of society.

### 5.4 Software Quality Attributes

- Email and Password validation is implemented to prevent invalidity of users.
- **Remember Me** is implemented to save the user's password for onwards login
- **Any Trouble** feature is implemented to make login smooth, if a user faces any issue regarding login, we will help him out with the user manual and he can also contact us to resolve the issue.
- **How it works** is the feature which will open a new screen for a user having all the details of using our portal so that it becomes a convenience for anyone to use it

- While adding the complaint, validation is implemented about the details of the complaint and the maximum number of words allowed to write the complaint
- Unique ids are assigned to the users to prevent redundancy issues
- Sorting and searching members and complaints is made easy to make our flow better
- Status of Complaint is the best feature because it keeps the user updated about his/her complaint and helps in the complaint tracking process.

## 5.5 Business Rules

- While entering the complaint, the complainer has to make sure to provide the evidence in the form of video or image so that the complaint can be recorded otherwise it can't be picked up by the software.
- Fake complaints should be awarded with penalties and strict action will be taken against the complainer by the Admin.
- If Admin does not handle the complaint he will also have to pay a penalty.
- The complaint will only be solved if the admin approves it.
- Super admin will check the working efficiency of Admin.

## 6. Other Requirements

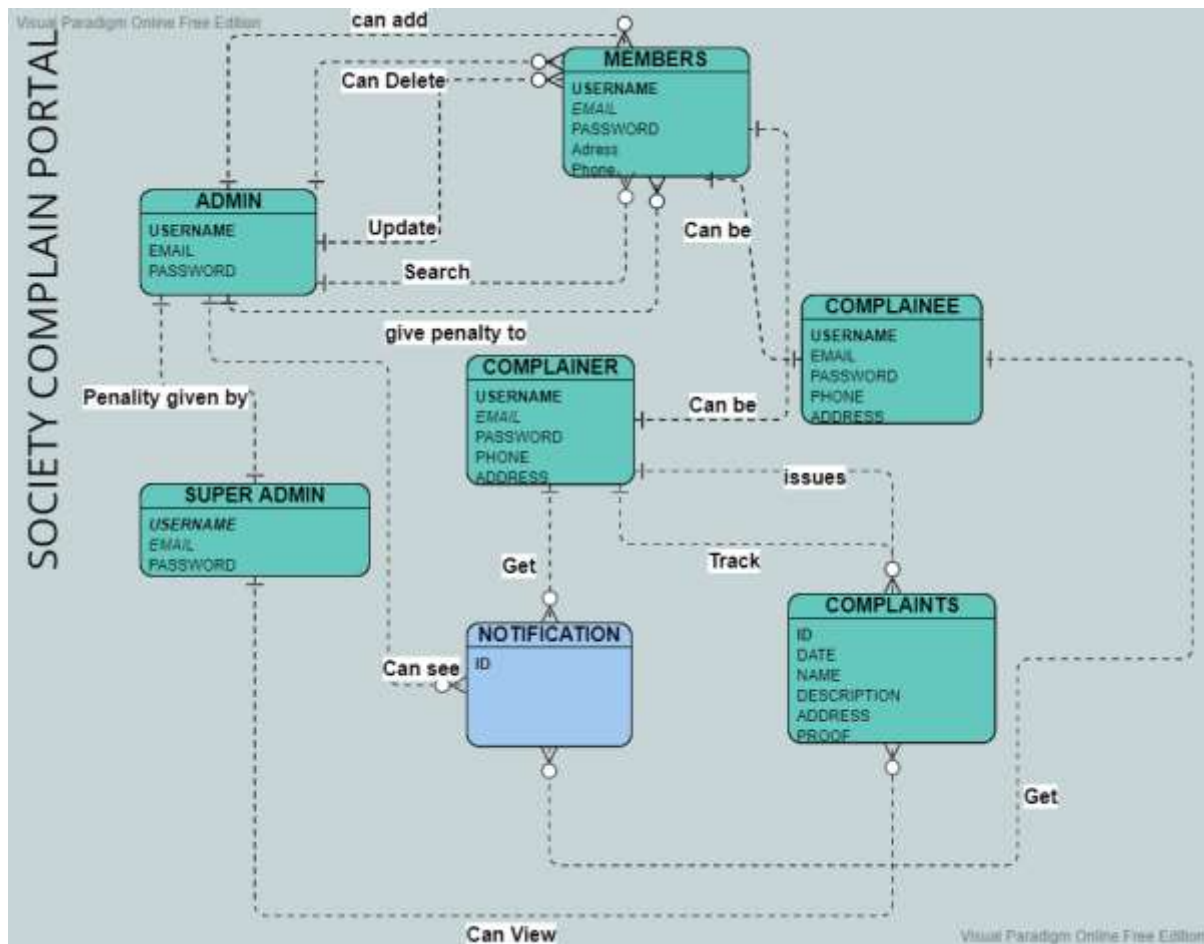
A general management backend with inventory and general system requirements is also required. A defined privacy policy, SSL certification and two-step verification through external mail or phone number is also necessary to ensure no breach in both user data and system data.

## Appendix A: Glossary

- **Business requirement:** A high-level business objective of the organization that builds a product or of a client who brings it out.
- **Business rule:** A policy, guideline, standard, or regulation that defines or constrains some facts of the business.
- **Constraint:** A restriction that's obligatory on the alternatives of the market to the developer for the planning and construction of a product.
- **Customer:** A project stakeholder, who requests the product, pays for it, selects and specifies, uses the product, or receives the output obtained by a product.
- **Developers:** One who programs and computers the program or styles the system to match the requirements of a system analyst.
- **Feature:** A group of logically connected functional requirements that gives a capability to the user and permits the satisfaction of a business objective.
- **Flowchart:** A model that shows the process steps and calls points within the logic of a process or in the logic of a program, like that of an activity diagram.
- **Functional requirements:** A statement of a chunk of needed functionality or behavior that a system can exhibit under certain conditions.
- **IEEE:** The Institute of Electrical and Electronics Engineers.
- **Nonfunctional requirements:** An outline of a property or characteristic that the system ought to exhibit.
- **SuperAdmin:** Super Admin is somebody who has the privilege to manage all the complaints cell directly.
- **Admin:** Admin is someone who is responsible for managing the portal, receive complaints, update them and resolve within the time constraint. He is also in charge of adding, updating, and deleting members of society.
- **Fake Complaint:** Any complaint without legit proof and evidence is called fake complain
- **ERD:** Entity Relationship Diagram

## Appendix B: Analysis Models

ERD for our “Society Complaint Portal” is:



## **Appendix C: To Be Determined List**

There is nothing to mention specifically in our “Society Complaint Portal” TBD. We have done a very specific part having specific things and have nothing to be determined.