A

Project Report

On

<u>Ahmedabad Live</u>

Submitted by

Kunal Solanki [210510113129]

Harsh Chaudhari [210510113161]

Alex Vaghela [210510113135]

as

Partial fulfillment of Semester VI of

Bachelors of Computer Applications

for A.Y. 2023-2024

Under the Guidance of

Internal Guide name : Prof. Manish Joshi

External Guide name : Vrunda Hingrajia

Company name : STYPIX

Submitted To

Parul Institute of Computer Application,

Faculty of IT & Computer Science

Parul University

PARULINSTITUTEOFCOMPUTERAPPLICATION

CERTIFICATE

This is to certify that **Kunal Solanki, Harsh Chaudhari, Alex Vaghela** the student(s) of Parul Institute of Computer Application, has/have satisfactorily completed the project entitled **"Ahmedabad Live"** as a part of course curriculum in BCA Semester-VI for the academic year 2023-2024 under guidance of **Prof. Manish Joshi.**

Enrollment Number:**-210510113129**

Enrollment Number:**-210510113161**

Enrollment Number:**-210510113135**

Quality of Work	Grade	Sign of Internal guide
Poor/Average/Good/ Excellent	**B / B+/A/A+**	

Date of Submission:-

HOD, Principal,

Dr.Hina Chokshi Dr.Priya Swaminarayan

Acknowledgement

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along the completion of our project. All that we have done is only due to such supervision and assistance and we would not forget to thank them.

*I respect and thank **Dr. Priya Swaminarayan**, Dean, FITCS for providing us an opportunity to do the project work in BCA and giving us all support and guidance which made us complete the project duly. We are extremely thankful to Mam for providing her support and guidance, although she had a busy schedule managing the academic affairs.*

*We would not forget to remember **Dr. Hina Chokshi**, HOD, BCA department for her encouragement and more over for her timely support and guidance till the completion of our project work.*

*We owe our deep gratitude to our project guide **Prof. Manish Joshi**, who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system.*

I am thankful to and fortunate enough to get constant encouragement, support and guidance from our Parents, all Teaching staff of the BCA Department which helped us in successfully completing our project work. Also, we would like to extend our sincere esteems to all staff in the laboratory for their timely support.

Kunal Solanki [210510113129]

Harsh Chaudhari [210510113161]

Alex Vaghela [210510113135]

INDEX

1. Company Profile

1.1 History

STYPIX delivers the best in class custom software solutions, elite software development teams and innovative cloud software to enterprise businesses across numerous industries.

We believe new technologies are the lifeline of every business in the modern age and aim to connect businesses across all industries to innovative software, technological development, solutions and services, in a manner that's faster, easier and better than ever before.

Our mission is to create leading, innovative software that creates economic and social value on a global scale, collaborating closely with our clients to help them achieve both their short and longer term goals.

1.2 scope of work

Android App: We provide services for android mobile apps from the idea to a successful app in play store stores.

iOS Apps: Our principal is to provide customer-centric iOS app experience that are loved by all.

Web Development: We build intuitive mobile and web apps powered by robust backend infrastructure.

UI/UX Design: Our design architects design a solution such that it meets the objectives of both business and end user.

Graphics Design: Beyond just creating amazing design for creative apps is just like a fully artistic concept.

1.3 Organization chart

- Harvi Jivani: Sr. CEO/ BDM
- Hardik Vyas: Sr. Project Head/ Web Development
- Ravi Kalola: Sr. Android Developer
- Meet Bhatt: Sr. Python Developer
- Zeel Bhatt: Sr. iOS Developer

1.4 Capacity of plant

With a team of 6 developers, STYPIX has the capacity to take on multiple small to medium-sized projects simultaneously. However, the company may need to adjust its workload and team size depending on the scope and complexity of each project. In general, the company's capacity will be determined by a number of factors, including:

Skill Sets: The expertise of each team member will determine the types of projects STYPIX can undertake. For example, if the team has a strong background in developing mobile applications, they may be able to take on more complex mobile app projects than web development projects.

2. Project Profile

2.1 Project Definition : Ahmedabad Live

2.2 Project Description

An online complaint management system that streamlines the process of filing and resolving complaints for various government departments. The system allows citizens to submit complaints online, which are then directed to the appropriate department for resolution. The system provides an efficient and transparent process for citizens to track the progress of their complaints and receive timely updates. It also enables government departments to manage and monitor complaints more effectively, resulting in improved service delivery and customer satisfaction. The system is secure and user-friendly, with features such as automatic notifications, document attachments, and detailed reporting. Overall, an online complaint management system enhances accountability and transparency in government operations and promotes citizen engagement and trust.

2.3 Existing/ Legacy System

2.3.1 Manual System: Some government departments still rely on manual systems for receiving and processing complaints. This involves citizens submitting their complaints in person, by phone, or by mail. These complaints are then recorded in a paper-based system and manually tracked until resolution.

2.3.2 Email System: Some government departments have implemented email-based systems where citizens submit their complaints via email. These emails are then directed to the appropriate department for resolution. This system is more efficient than a manual system, but it still be challenging to track and manage complaints effectively.

2.3.3 Web-based System: Some government departments have implemented web-based complaint management systems where citizens submit their complaints online. These systems typically involve filling out an online form with details of the complaint, which is then directed to the appropriate department for resolution. These systems are more efficient and transparent than manual or email-based systems, but they still have limitations in terms of tracking and managing complaints effectively.

2.3.4 Call Center System: Some government departments have implemented call center-based systems where citizens call a designated phone number to file their complaints. These systems typically involve call center agents recording details of the complaint, which is then directed to the appropriate department for resolution. These system be efficient for citizens who prefer to file their complaints over the phone, but they be challenging to manage effectively due to the high volume of calls.

2.4 Problem Statement

The purpose of an online complaint management system that streamlines the process of filing and resolving complaints using Android and MySQL is to provide an efficient and effective solution for organizations to manage customer complaints. By providing a simple and user-friendly platform for customers to file complaints and for organizations to manage and resolve those complaints, the system aims to improve customer satisfaction, streamline complaint management processes, and reduce costs associated with traditional complaint handling methods.

The system's Android app provides a convenient way for customers to file complaints from anywhere at any time, with all the necessary information entered directly into the MySQL database. This information then be easily accessed and managed by the organization's complaint handling team, who use the system to track the status of each complaint, assign complaints to the appropriate personnel for investigation and resolution, and communicate with customers throughout the process.

By streamlining the complaint management process, the system reduces the time and effort required to handle each complaint, leading to faster resolution times and increased customer satisfaction. Additionally, the system provide valuable data and insights into the types of complaints that are most common, allowing organizations to identify areas for improvement in their products, services, and processes.

2.5 Need for new system

2.5.1 Complaint Tracking: The system provides a unique complaint tracking number to each complaint, which the citizen uses to track the progress of the complaint. The system also sends notifications to the citizen about the status of their complaint, such as when it has been received, assigned, and resolved.

2.5.2 Android Application: The Android application is designed to make it easy for citizens to file complaints. The app have a user-friendly interface that enables citizens to quickly file complaints by filling out a form, attaching relevant documents or images, and submitting them.

2.5.3 Data Analysis: The online complaint management system provides an opportunity for data analysis and reporting, which help identify trends and patterns in complaints. This lead to improved policies and practices within government departments.

2.5.4 Documentation: The online complaint management system store all complaint data, documentation, and resolution details in a centralized and secure database. This improves record-keeping, reduces paperwork, and simplifies audits.

2.6 Proposed system

The system boasts a user-friendly interface, ensuring ease of navigation for both complainants and government officials. It enables online complaint filing through an Android app, allowing complainants to furnish comprehensive details such as date, time, location, and description. Real-time complaint tracking is facilitated, empowering complainants to monitor their grievance's progress—receipt, assignment, or resolution. Leveraging automatic assignment, the system efficiently allocates complaints to relevant officials based on expertise. Seamless communication channels via the app facilitate alerts, notifications, and messaging between complainants and officials. An escalation mechanism intervenes if resolutions exceed set timelines, automatically forwarding complaints to higher authorities. Robust analytics furnish government officials with vital data—complaint volumes, resolutions, pending cases, and resolution durations—aiding in process optimization. Upholding stringent security and privacy measures, including encryption, secure logins, and access controls, the system safeguards personal information.

2.7 Scope

The system's scope encompasses the development and implementation of a comprehensive complaint management platform. It includes the creation of a user-friendly interface accessible to both complainants and government officials. The system's functionalities involve an Android app enabling online complaint submissions, detailed information provision (date, time, location, description), and real-time tracking of complaint statuses.

Automatic complaint assignment based on expertise and a robust communication framework within the app are integral components. Additionally, the system integrates an escalation mechanism for unresolved complaints, directing them to higher authorities as necessary. Its analytics capabilities enable tracking and reporting of key metrics such as complaint volumes, resolutions, and pending cases, empowering officials to optimize the complaint resolution process.

Security and privacy features like encryption, secure logins, and access controls form a crucial aspect, ensuring the protection of personal information. Overall, the system aims to streamline complaint management, enhance communication between stakeholders, provide transparency in tracking complaints, and uphold stringent security measures throughout its operations.

2.8 Outcomes

2.8.1 Enhanced User Experience: Users, both complainants and government officials, experience a streamlined and intuitive interface, leading to increased engagement and participation.

Efficient Complaint Handling: The system's automated features expedite complaint assignment, resolution, and tracking, reducing response times and improving overall efficiency.

2.8.2 Transparency and Accountability: Real-time tracking empowers complainants to monitor the progress of their grievances, fostering transparency and accountability within the complaint resolution process.

2.8.3 Improved Communication: Seamless communication channels facilitate prompt interactions between complainants and officials, leading to clearer, more effective exchanges and faster resolutions.

2.8.4 Optimized Resource Allocation: Analytics-driven insights provide officials with vital data, enabling them to optimize resources, identify trends, and make data-informed decisions for process improvement.

2.8.5 Timely Escalation and Resolution: The escalation mechanism ensures unresolved complaints reach higher authorities promptly, ensuring timely resolutions and preventing prolonged grievances.

2.9 Tools & technology used with short justification

Android Studio: As the official Integrated Development Environment (IDE) for Android app development, Android Studio provides a comprehensive set of tools for building, testing, and debugging Android applications. Its user-friendly interface, along with features like code templates, visual layout editor, and emulator, makes it an ideal choice for creating the Android app interface for complainants to submit complaints and track their status.

PHP MyAdmin: PHPMyAdmin is a web-based tool for managing MySQL databases. It offers an intuitive interface to handle database operations such as creating, modifying, and deleting databases, tables, fields, and rows. In the context of a complaint management system, PHPMyAdmin facilitates efficient database management, ensuring seamless storage and retrieval of complaint-related information.

MySQL: MySQL is an open-source relational database management system (RDBMS) that is widely used for its reliability, scalability, and ease of integration. It's a robust choice for storing structured data related to complaints, user accounts, complaint statuses, and interactions between complainants and officials. MySQL compatibility with Android and its ability to handle concurrent transactions make it suitable for handling the data backend of the system.

JDK (Java Development Kit): Android apps are primarily developed using Java programming language. The JDK is essential for Android development as it includes tools for compiling, debugging, and running Java applications. Since Android apps are typically written in Java, the JDK is fundamental in enabling developers to create functionalities within the Android app, ensuring compatibility and proper execution on Android devices.

Why we choose Android?

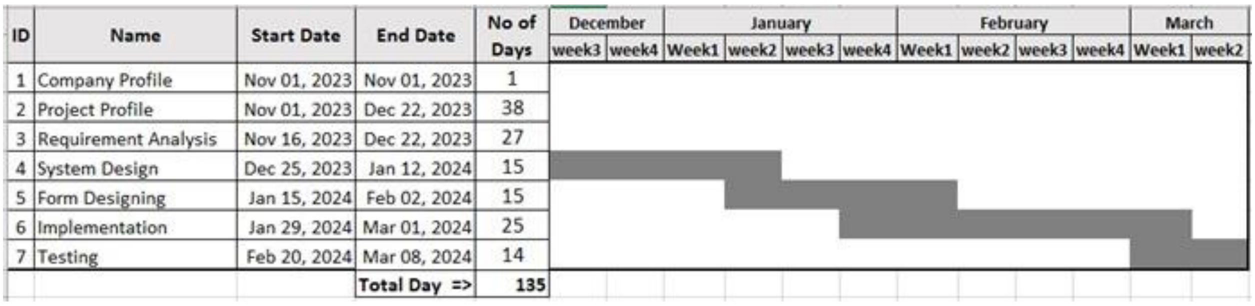
The Android is an open-source Operating system and hence possesses a vast community for support.

The design of the Android Application has guidelines from Google, which becomes easier for developers to produce more intuitive user applications.

Fragmentation gives more power to Android Applications. This means the application can run two activities on a single screen.

Releasing the Android application in the Google play store is easier when it is compared to other platforms.

2.10 Project plan



ID	Name	Start Date	End Date	No of Days	December		January				February				March	
					week3	week4	Week1	week2	week3	week4	Week1	week2	week3	week4	Week1	week2
1	Company Profile	Nov 01, 2023	Nov 01, 2023	1												
2	Project Profile	Nov 01, 2023	Dec 22, 2023	38												
3	Requirement Analysis	Nov 16, 2023	Dec 22, 2023	27												
4	System Design	Dec 25, 2023	Jan 12, 2024	15	▨	▨	▨	▨								
5	Form Designing	Jan 15, 2024	Feb 02, 2024	15				▨	▨	▨	▨					
6	Implementation	Jan 29, 2024	Mar 01, 2024	25						▨	▨	▨	▨	▨	▨	
7	Testing	Feb 20, 2024	Mar 08, 2024	14											▨	▨
			Total Day =>	135												

3. Requirement Analysis

3.1 Feasibility Study

3.1.1 Technical Feasibility: Evaluate the technical aspects required to develop the system. Assess if the necessary technologies (Android development tools, database management systems) are available and if the development team possesses the required skills to build the system.

3.1.2 Operational Feasibility: Determine if the proposed system aligns with the operational procedures of handling complaints within the concerned government entity. Assess the readiness of the organization to adopt the system, including the availability of resources, training needs, and any potential disruptions during implementation.

3.1.3 Economic Feasibility: Conduct a cost-benefit analysis to assess the financial viability of the system. Evaluate development costs, hardware and software requirements, maintenance costs, and potential savings or revenue generated from improved complaint handling efficiency.

3.1.4 Legal and Compliance Feasibility: Ensure that the system complies with relevant legal regulations, data protection laws, and government policies. Assess any potential legal hurdles or compliance issues that might arise during development and deployment.

3.1.5 Schedule Feasibility: Determine the time frame required for system development, testing, and deployment. Evaluate if the proposed schedule aligns with the organization's objectives and if the system can be delivered within the specified time frame.

3.1.6 Resource Feasibility: Evaluate the availability of resources such as skilled personnel, infrastructure, and support systems required for the development, implementation, and maintenance of the complaint management system.

3.1.7 Risk Analysis: Identify potential risks and challenges that might arise during the development and implementation phases. Develop mitigation strategies to address these risks and ensure a smoother deployment process.

3.2 Users of the system

3.2.1 Admin:

- It can only mange system and its privileges.
- If receiver to check complain, It can notify receiver regarding citizen complain.

3.2.2 Receiver (Government Servant):

- It is one type of sub admin; it can be Government office or any government staff who can access all type of citizen complain.
- It can forward complain to particular department head.
- It is solved citizen complain and update status.

3.2.3 User:

- It can register & login in to system
- It is created Complain Regarding Government work
- It can receive complain status

3.3 Modules of the system

3.3.1 Role Management: This module is used for assigning the roles to the government employee.

3.3.2 Complaint Management: This module is helpful for storing the complaint and process on that complaint as well as filtering it.

3.3.3 Department Management: This module deals with adding, updating and deleting the various department of the government.

3.3.4 Feedback Management: This module stores the information related to the feedback given by the people.

3.3.5 Complaint Status Management: This module keeps track of the status of the complaint made by the people in terms of pending or resolved.

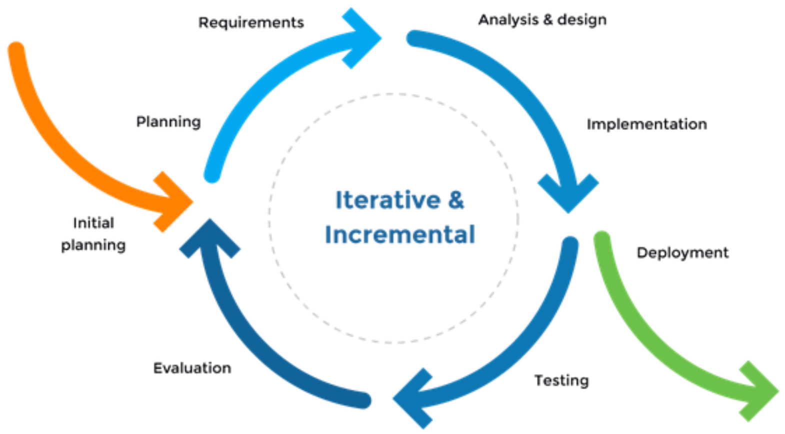
3.3.6 Location Management: This module stores the information of the address and area of the people.

3.4 Process model

Incremental Model design

The incremental build model is a method of software development where the model is designed, implemented and tested incrementally (a little more is added each time) until the product is finished. It involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements. This model combines the elements of the waterfall model with the iterative philosophy of prototyping.

Following is the pictorial representation of Iterative and Incremental model:



Iterative and Incremental development is a combination of both iterative design or iterative method and incremental build model for development. "During software development, more than one iteration of the software development cycle may be in progress at the same time." and "This process may be described as an "evolutionary acquisition" or "incremental build" approach."

In incremental model the whole requirement is divided into various builds. During each iteration, the development module goes through the requirements, design, implementation and testing phases. Each subsequent release of the module adds function to the previous release. The process continues till the complete system is ready as per the requirement.

The key to successful use of an iterative software development life-cycle is rigorous validation of requirements, and verification & testing of each version of the software against those requirements within each cycle of the model. As the software evolves through successive cycles, tests have to be repeated and extended to verify each version of the software.

Iterative Model Application

Like other SDLC models, Iterative and incremental development has some specific applications in the software industry. This model is most often used in the following scenarios:

- Requirements of the complete system are clearly defined and understood.
- Major requirements must be defined; however, some functionalities or requested enhancements may evolve with time.
- There is a time to the market constraint.
- A new technology is being used and is being learns by the development team while working on the project.

Resources with needed skill set are not available and are planned to be used on contract basis for specific iterations.

- There are some high-risk features and goals which may change in the future.
- Integration and Testing: All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
- Deployment of system: Once the functional and nonfunctional testing is done, the product is deployed in the customer environment or released into the market.
- Maintenance: There are some issues which come up in the client environment. To fix those issues patches are released. Also, to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

3.5 Hardware & Software requirement

Minimum Software Requirement

1) Server Side

Operating system: Window 7 & 10 or Any Compatible Server OS

Front end: Android

Back end: PHP web services with MySQL

Technology: PHP

Minimum Hardware Requirement:

1) Sever Side

Operating system: Android Mobile

Processor:1.5 GHZ

RAM:8 GB

HDD:100 GB of free Space on Hard Disk

2) Client Side

Processor:1.5 GHZ

RAM: 2GB

HDD:4GB of free Space on Internal Storage

4. System Design

4.1 Use case Diagram

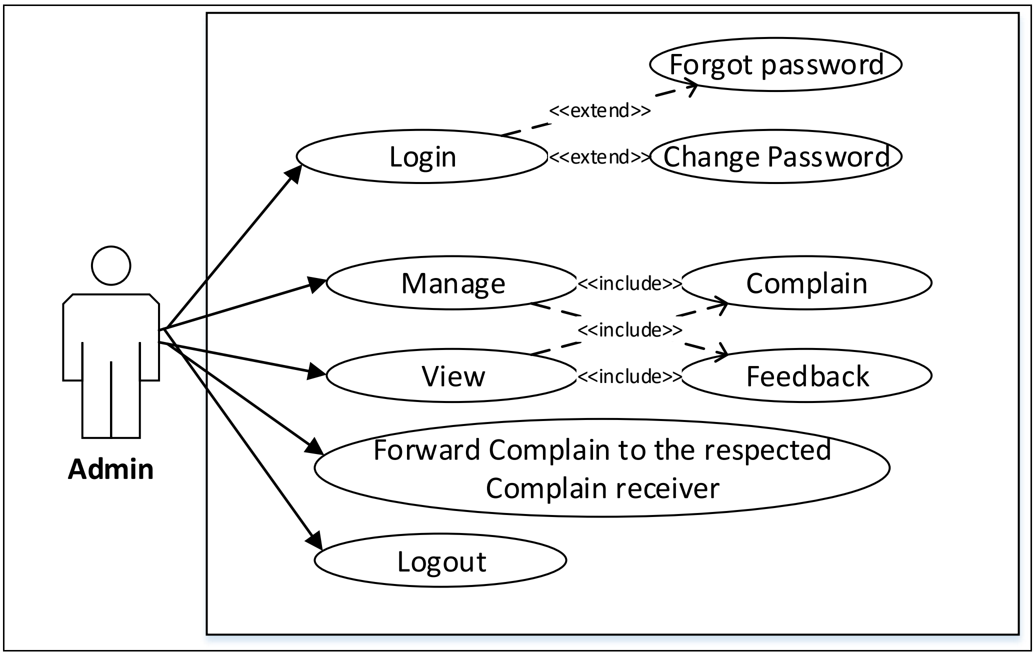


Fig: 4.1 Admin Use case Diagram

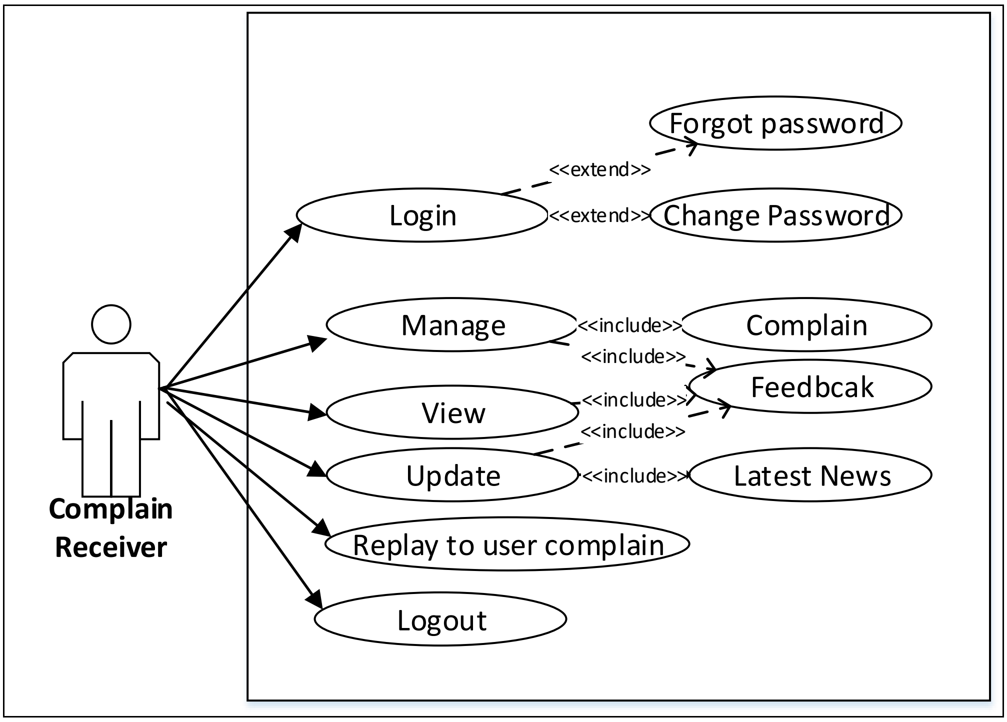


Fig: 4.1 Complain Receiver Use Case Diagram

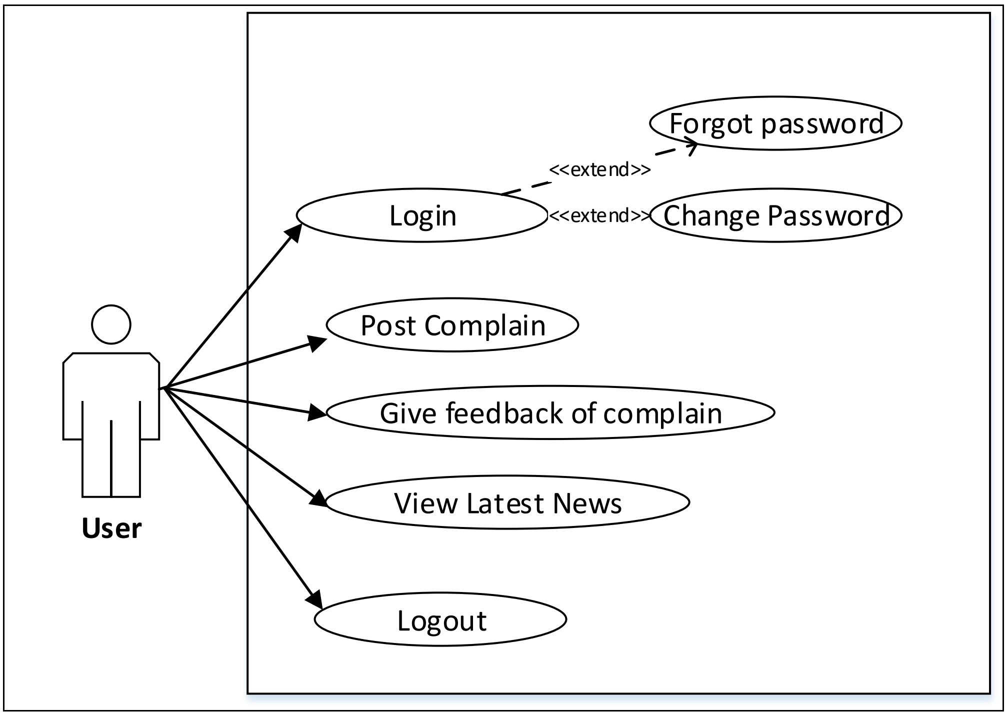


Fig 4.1 User (Citizen) Use case Diagram

4.2 Structured Diagram

4.2.1 Data Flow Diagram

DFD Level – 0

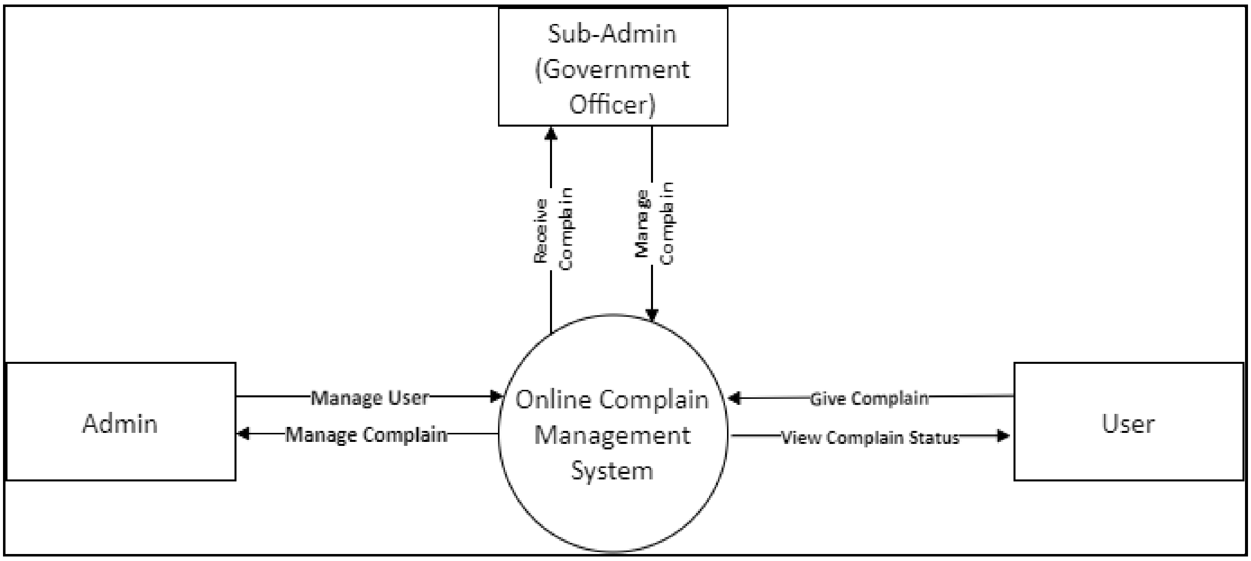


Fig: 4.2.1 Data Flow Diagram

DFD Level-1

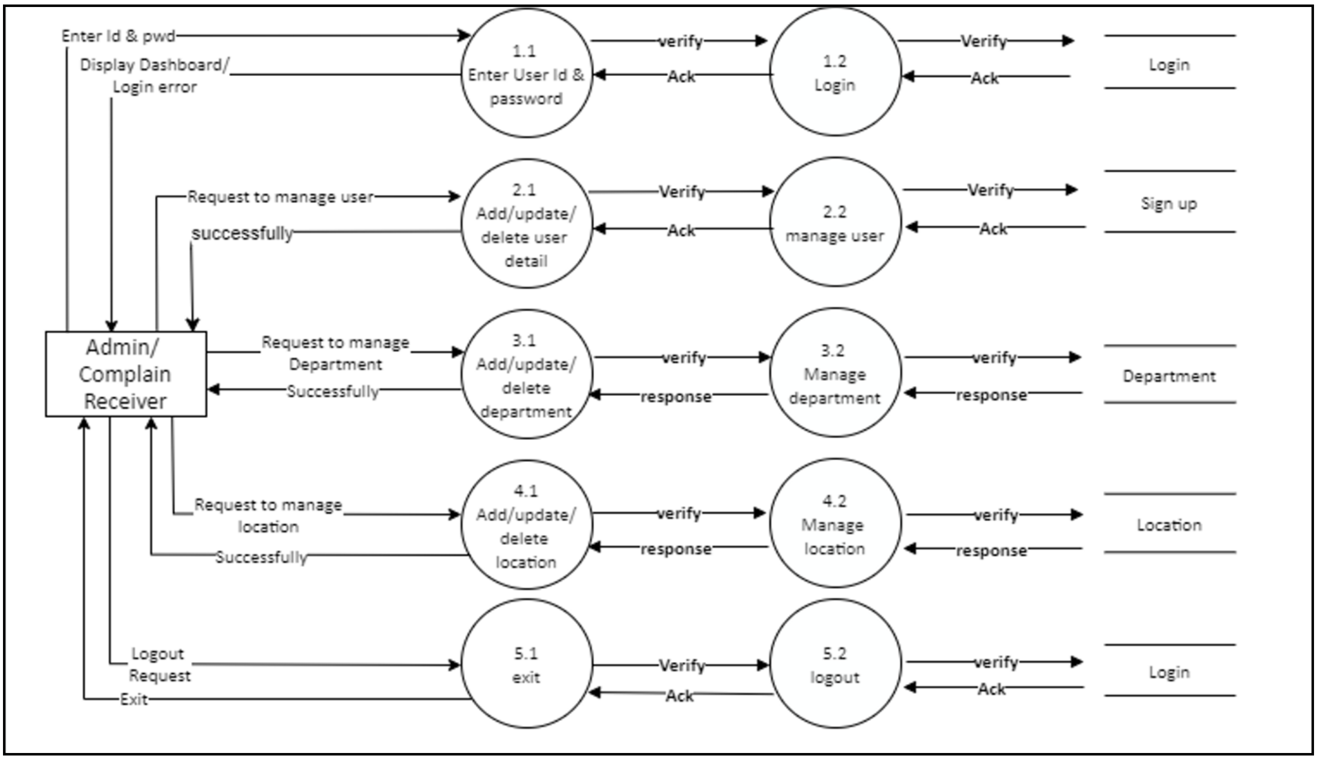


Fig: 4.2.1 Admin & Receiver Data Flow Diagram

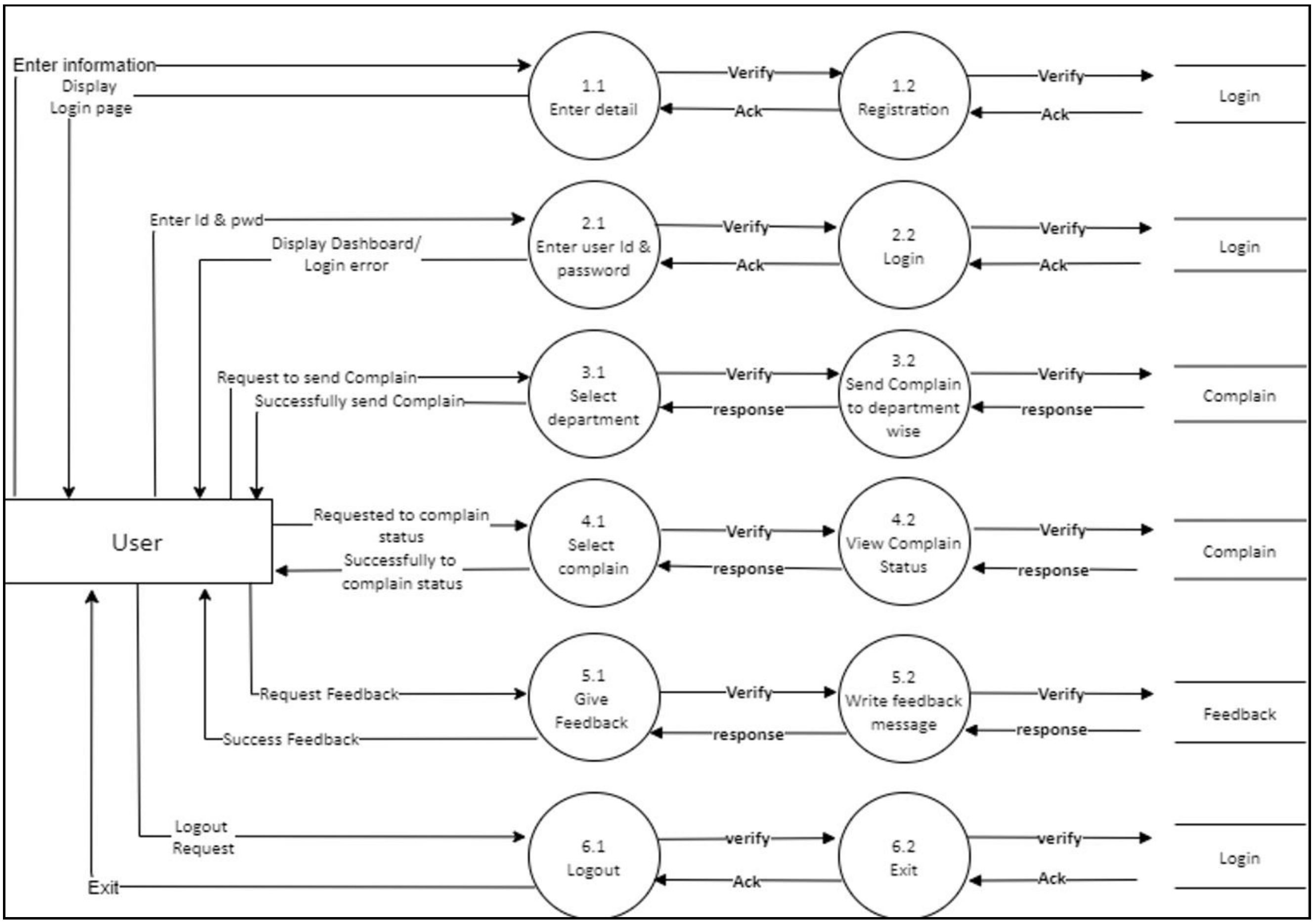


Fig: 4.2.1 User Data Flow Diagram Level

DFD Level-2

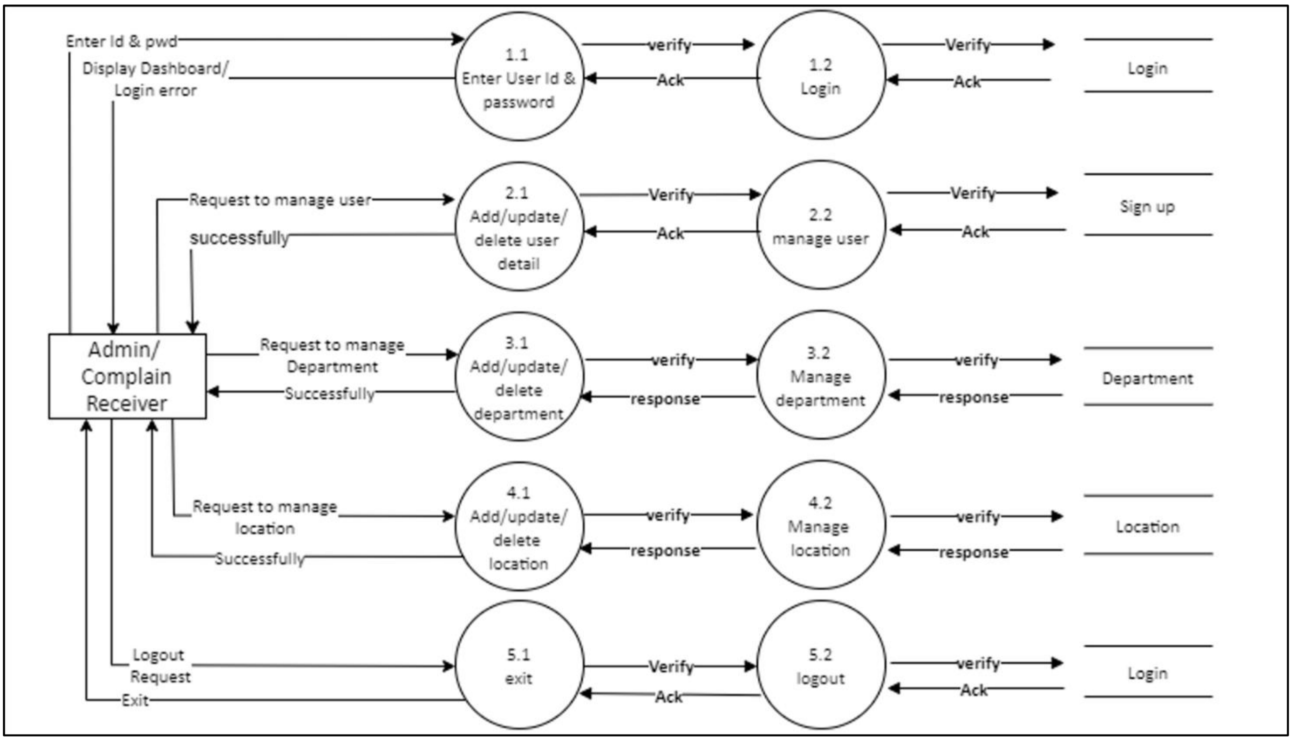


Fig: 4.2.1 Admin & Receiver Data Flow Diagram

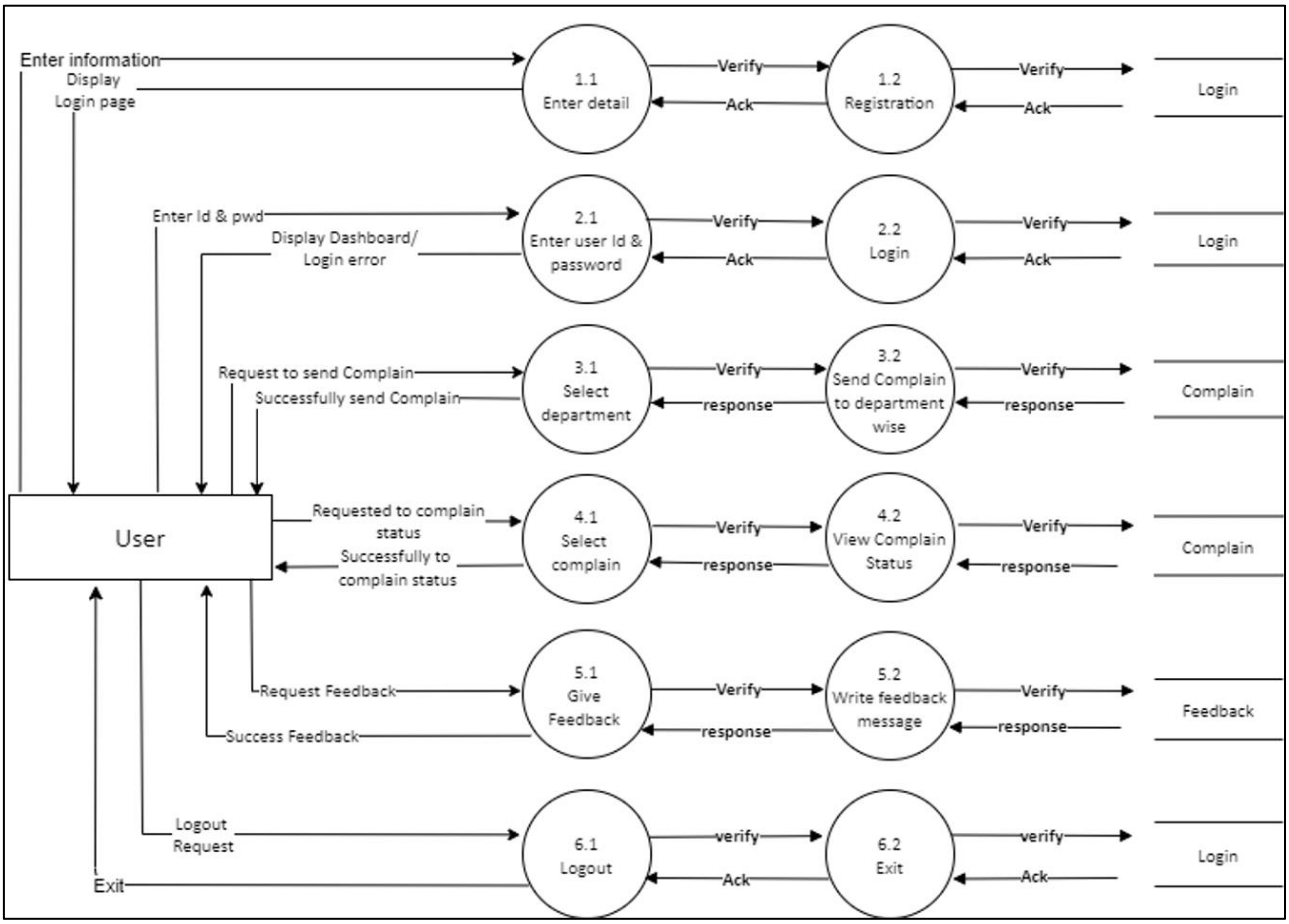


Fig: 4.2.1 User Data Flow Diagram

4.2.2 Flow Chart

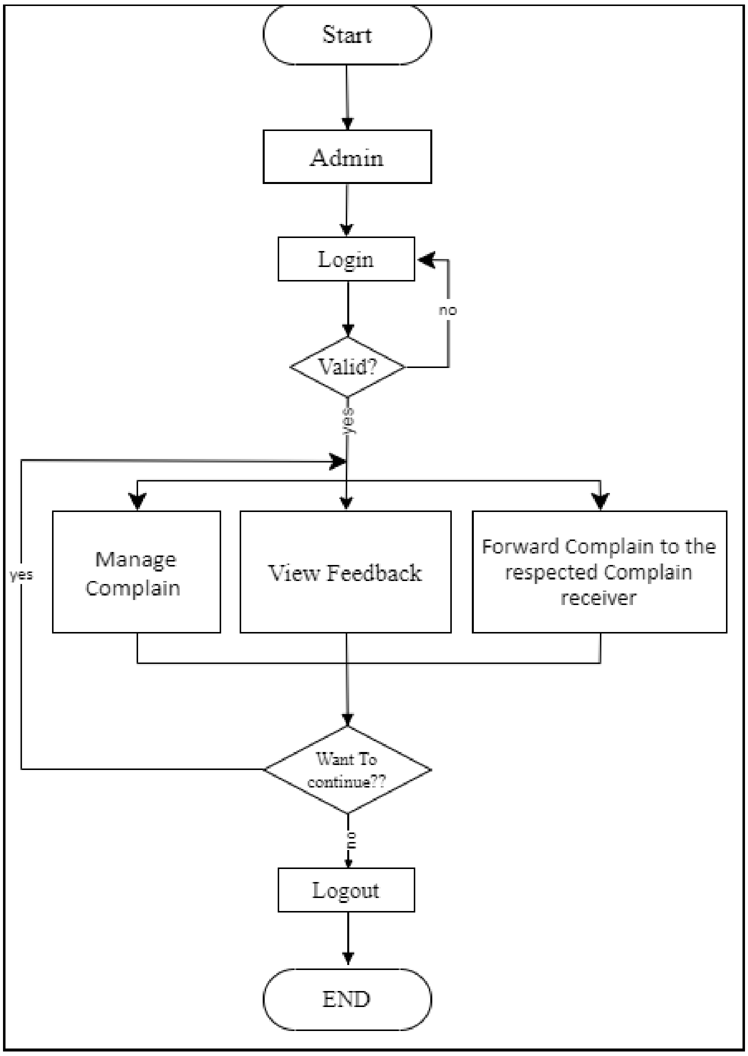


Fig: 4.2.2 Admin Flow Chart

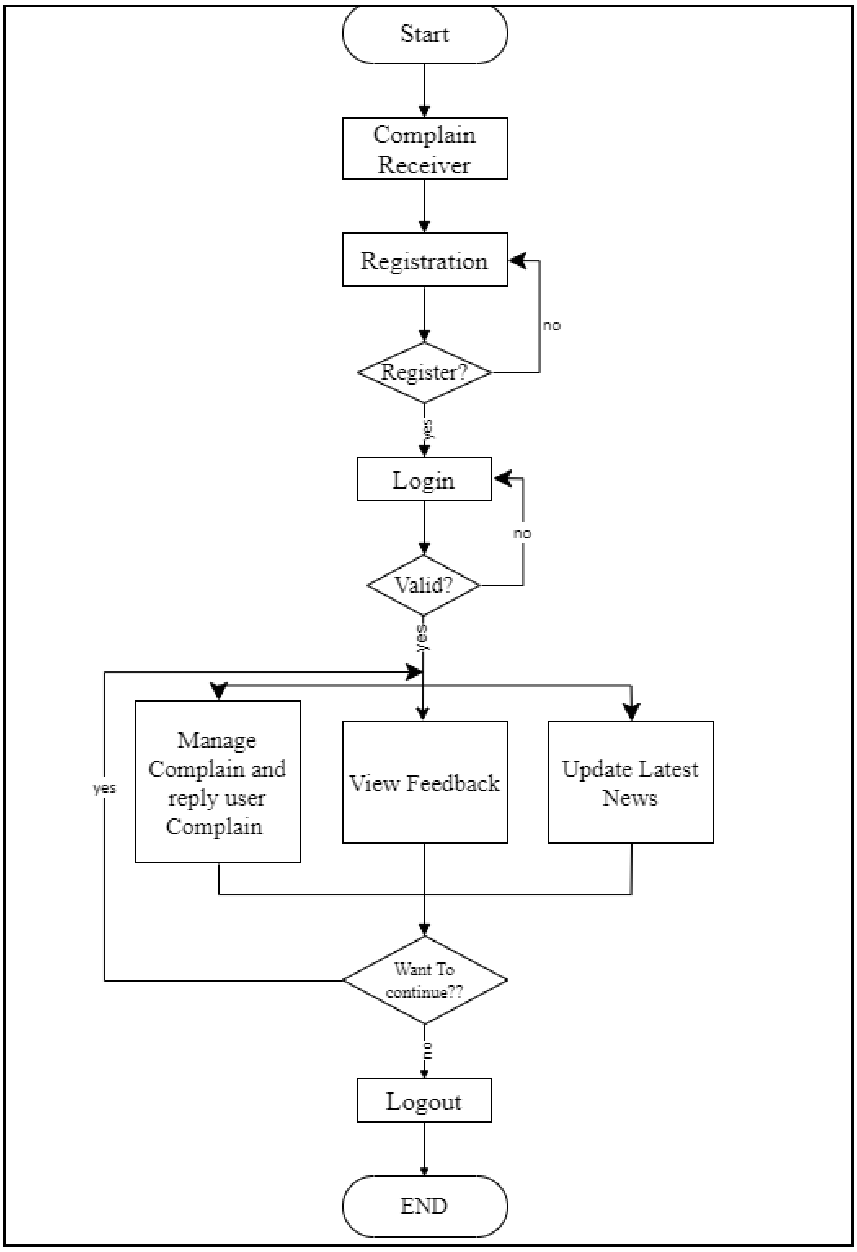


Fig: 4.2.2 Receiver Flow Chart

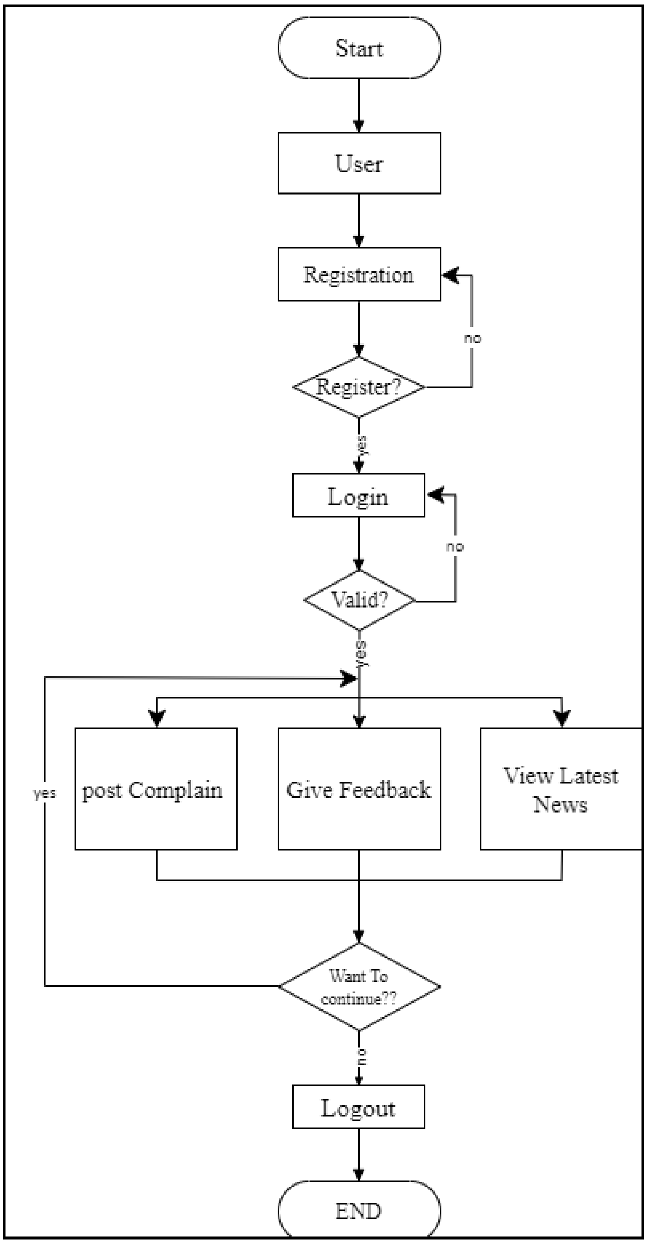


Fig: 4.2.2 User (Citizen) Flow Chart

4.3 OOAD Diagram

4.3.1 Class Diagram

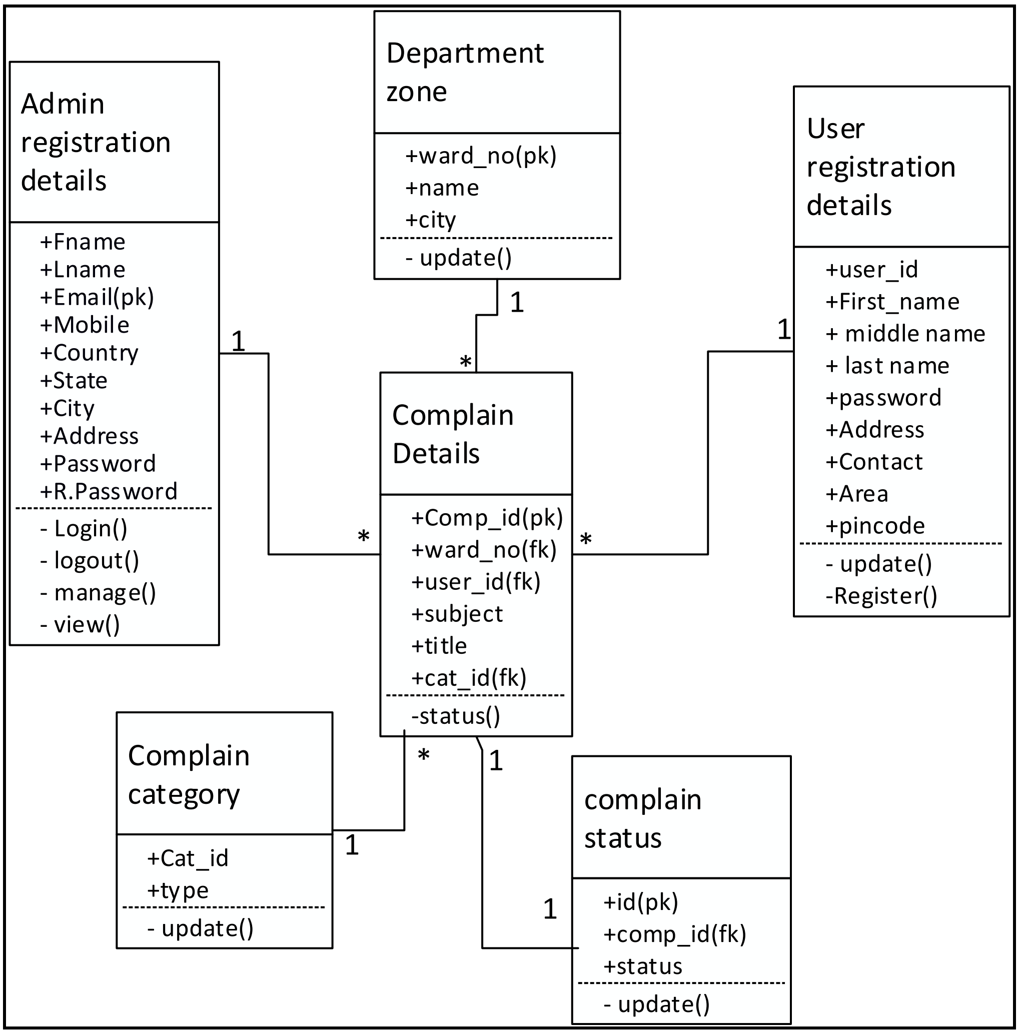


Fig: 4.3.1 Class Diagram

4.3.2 Sequence Diagram

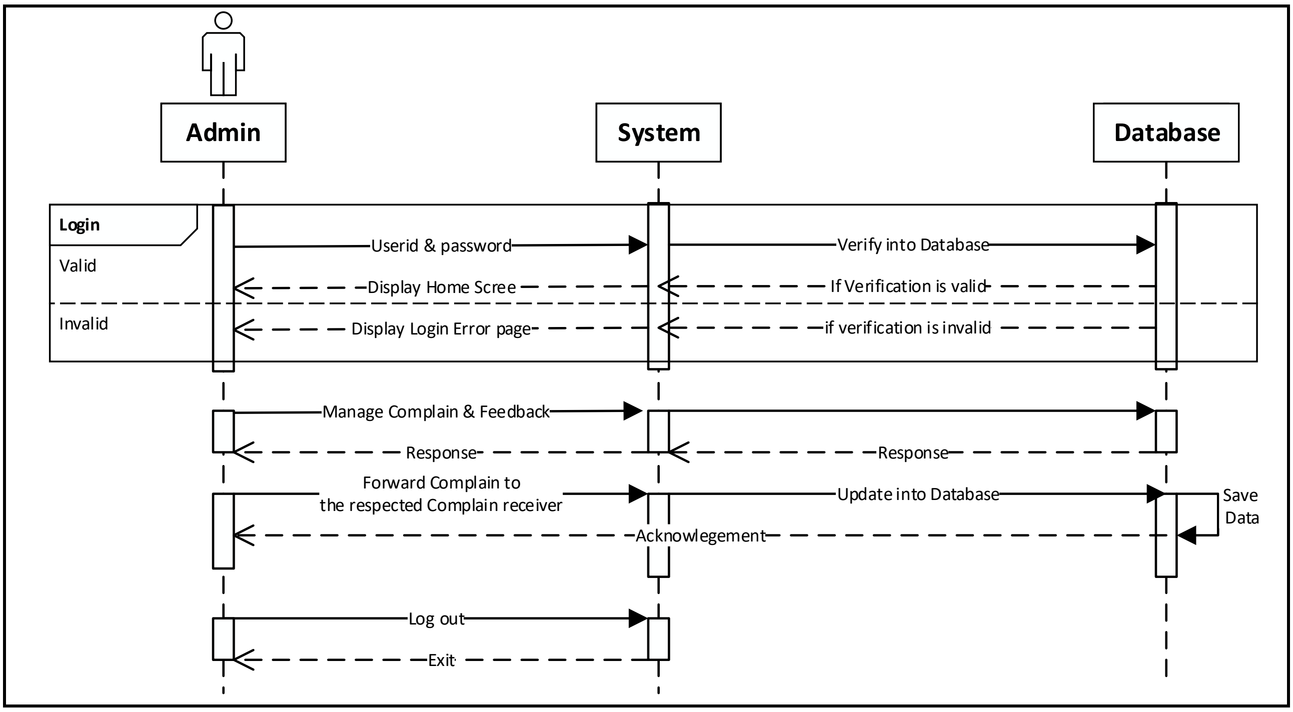


Fig: 4.3.2 Admin Sequence Diagram

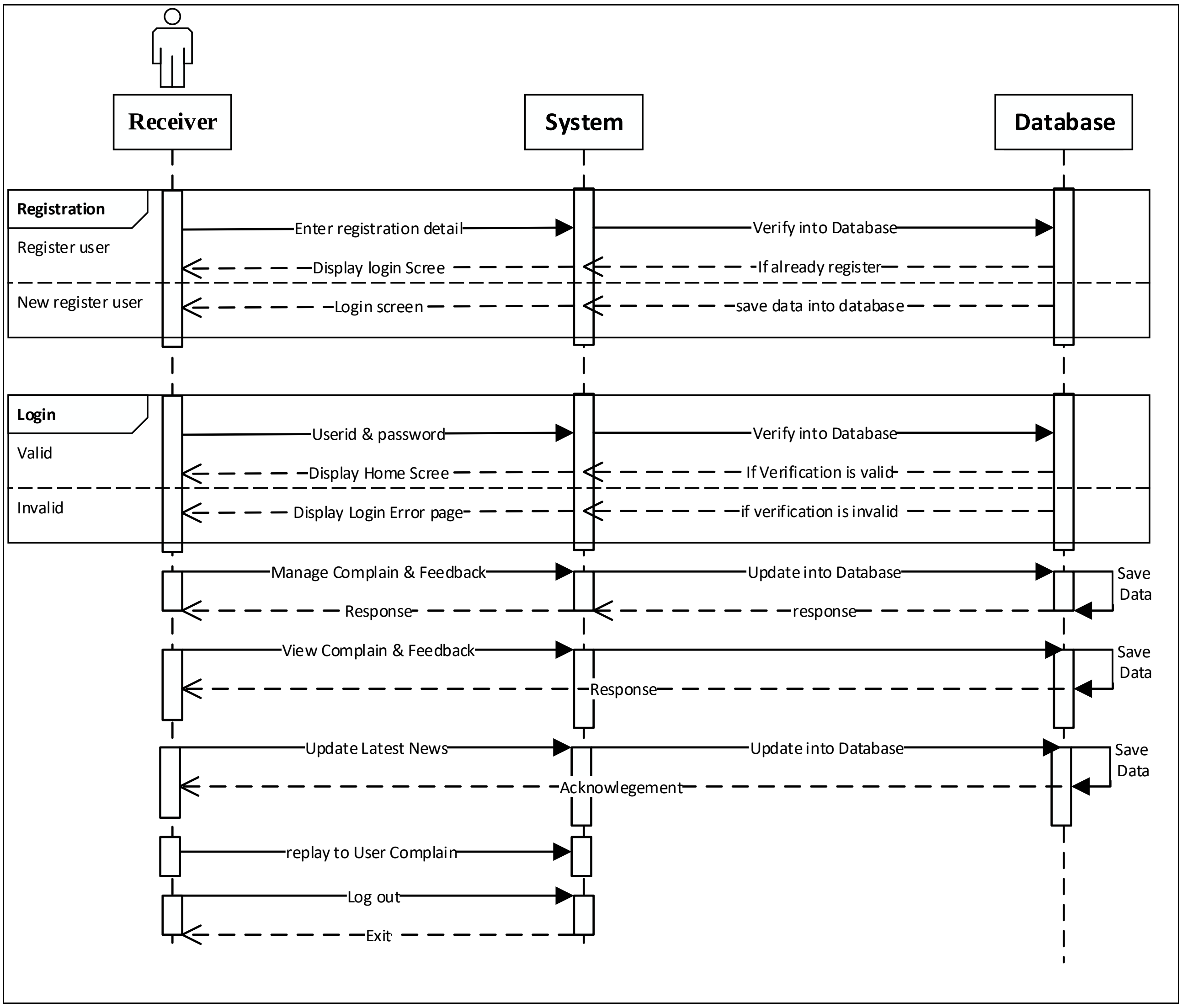


Fig: 4.3.2 Receiver Sequence Diagram

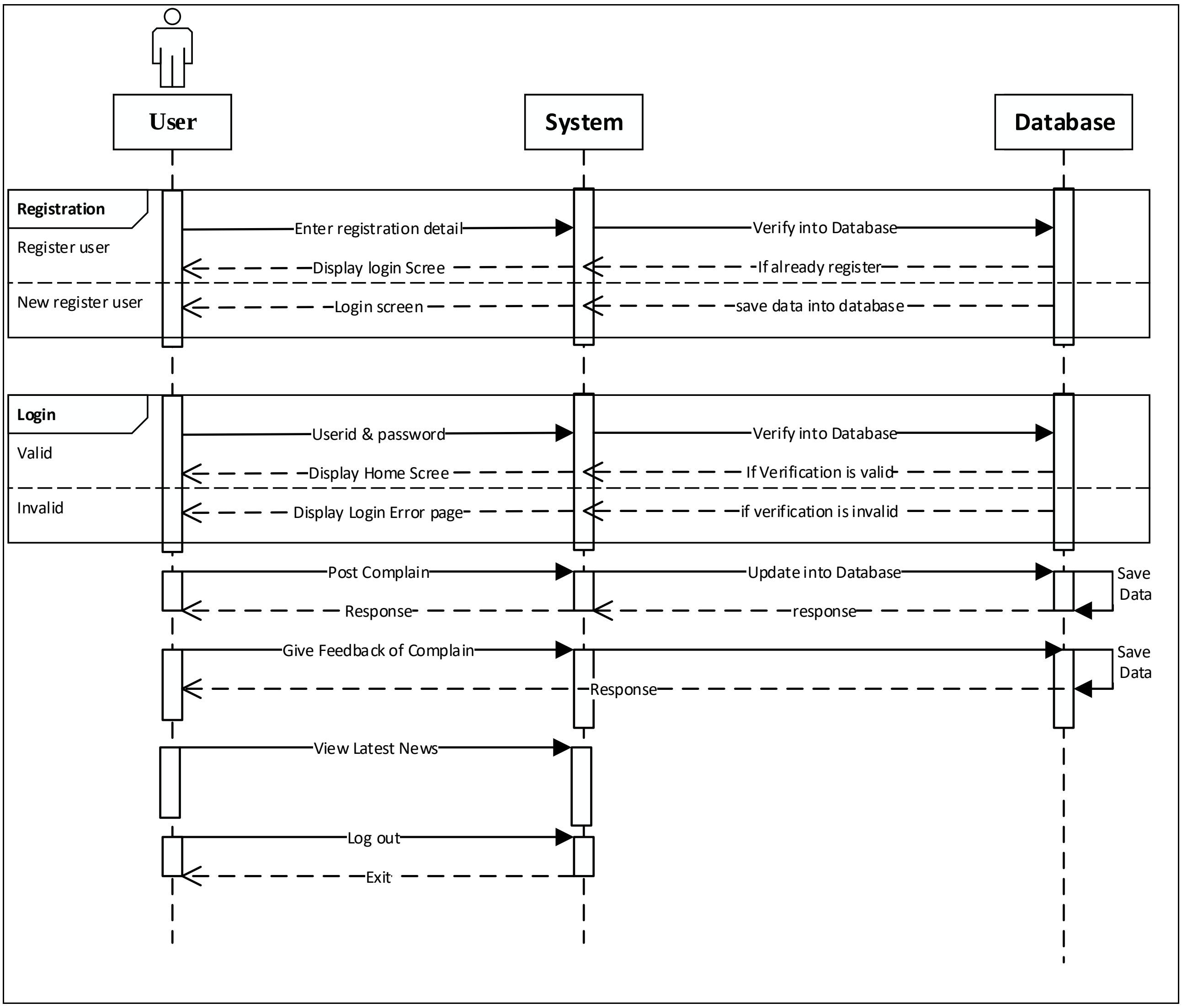


Fig: 4.3.2 User (Citizen) Sequence Diagram

4.3.3 Activity Diagram

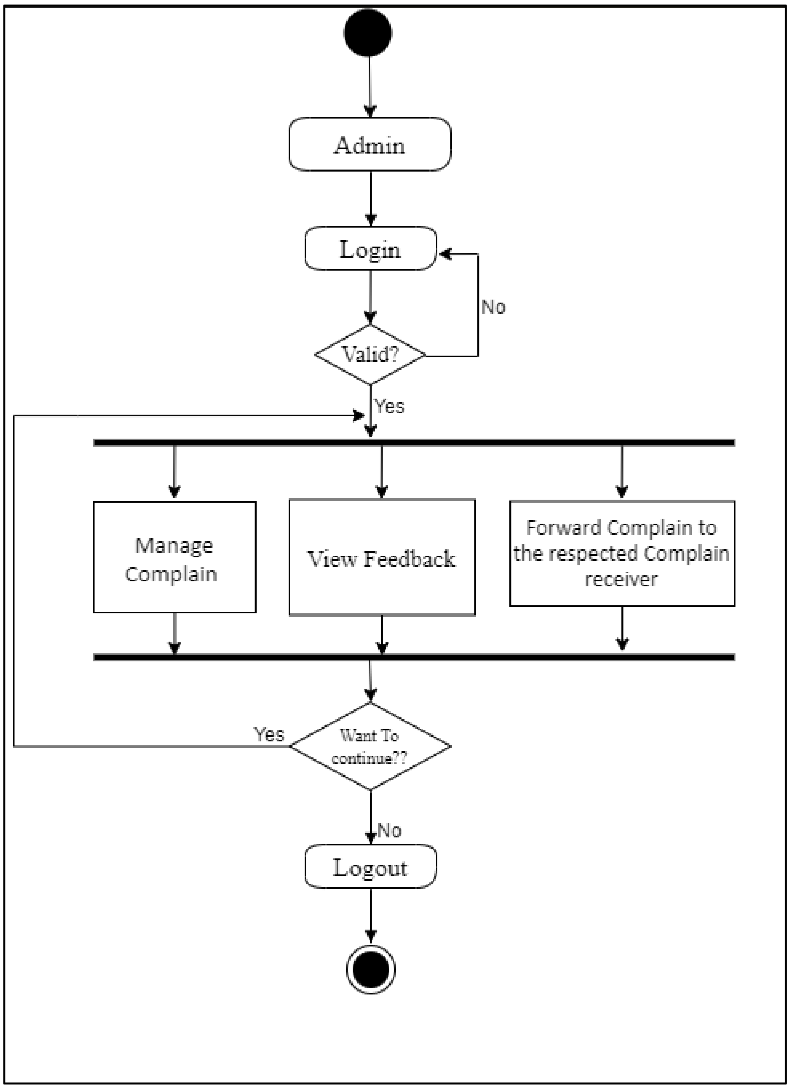


Fig: 4.3.3 Admin Activity Diagram

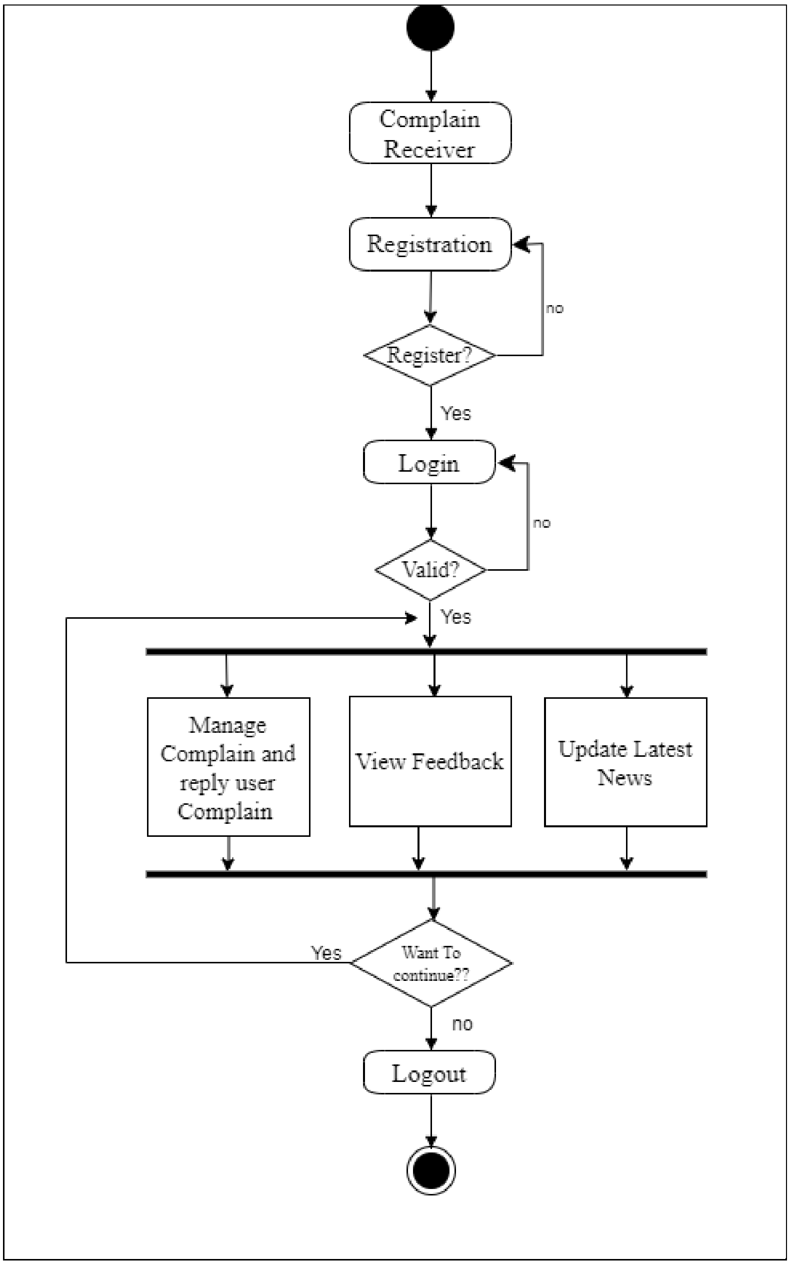


Fig: 4.3.3 Receiver Activity Diagram

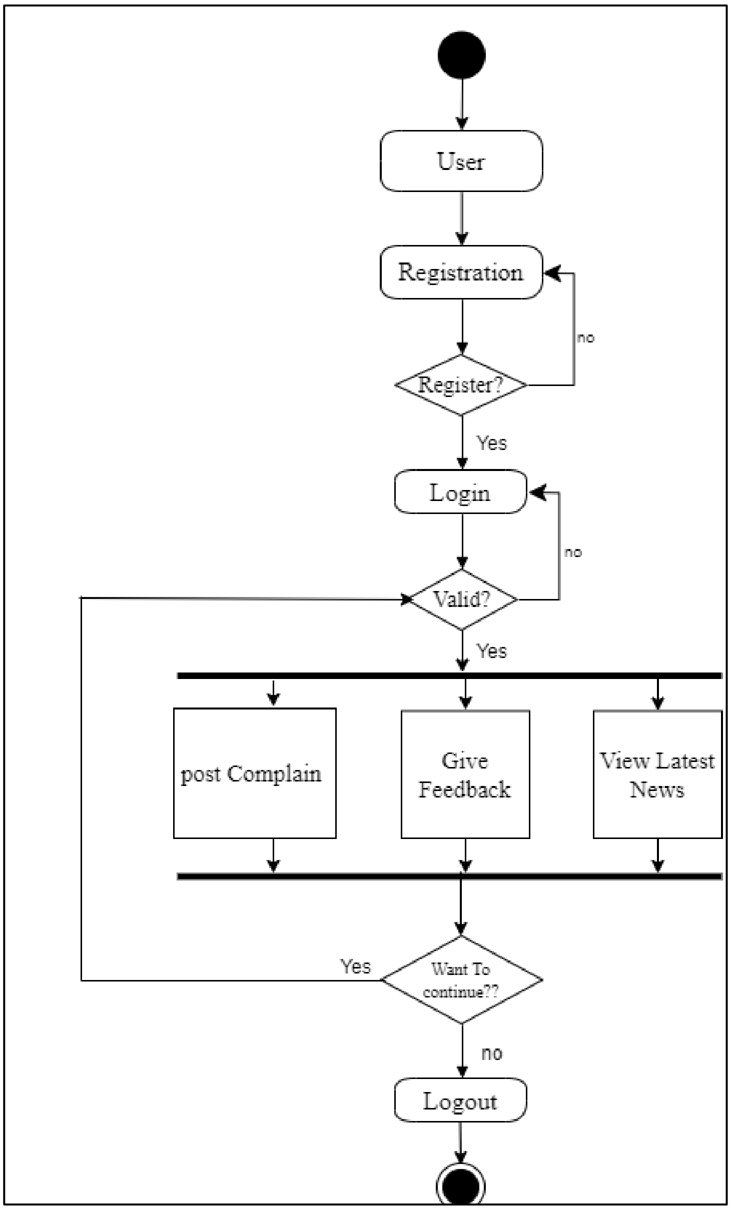
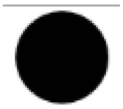
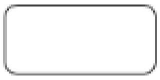
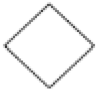
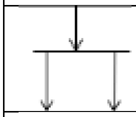
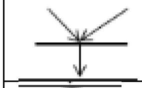
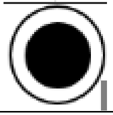


Fig: 4.3.3 User (Citizen) Activity Diagram

- Description of Activity Diagram

A UML (Unified Modeling Language) diagram called a "activity diagram" shows the progression of activities or processes in a system or business process. It is utilised to explain the behaviour or workflow of a system or a component thereof, demonstrating the transfer of control from one operation to another.

Activities and transitions between activities are represented by nodes and edges in an activity diagram. The edges are represented by arrows, whereas the nodes are represented by rounded rectangles. A key component of an activity diagram is:

Symbol	Description
	The starting point of the activity diagram
	The action or task that needs to be performed. It is represented by a rounded rectangle with the name of the activity inside.
	A diamond-shaped node that represents a decision point in the process, where the flow of control is split into two or more branches depending on a certain condition.
	A node that represents a point where a single control flow splits into multiple flows.
	A node that represents a point where multiple control flows merge into a single flow.
	The end point of the activity diagram

A wide variety of activities, including business processes, software development processes, and engineering processes, may be modelled using an activity diagram. It helps stakeholders comprehend the flow of complicated processes and pinpoint opportunities for improvement by articulating them in a straightforward and succinct manner.

4.4 Entity Relationship Diagram

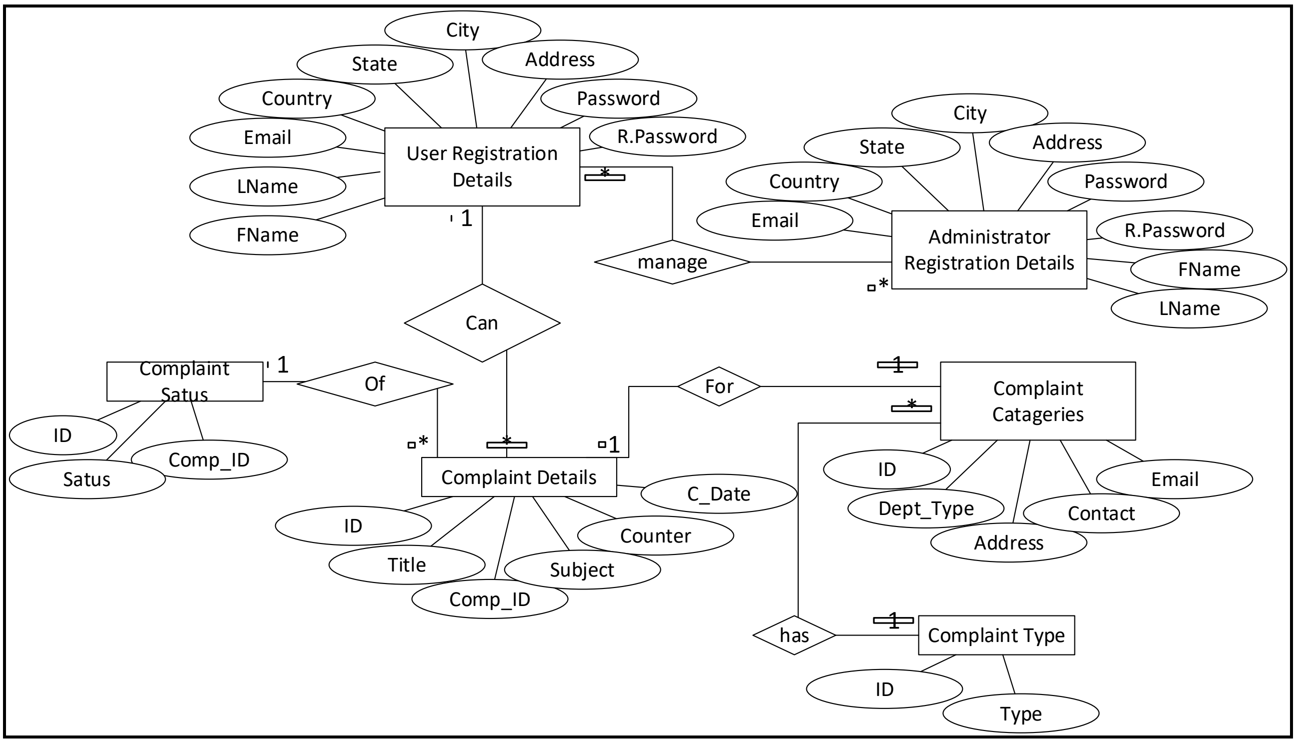


Fig: 4.4 E-R Diagram

- **E-R Diagram Description**

An ER (Entity-Relationship) diagram is a graphical representation of entities, their attributes, and the relationships between them in a database. Entities are represented by rectangles, attributes are depicted within ovals connected to their respective entity, and relationships are shown as diamond shapes connecting related entities.

Entities are shown as rectangular boxes in an ER diagram, and lines connecting the boxes show the relationships between the entities. An ER diagram consists of three primary parts:

Entities: In the system being modeled, an entity is a person, location, thing, or concept. In the ER diagram, each entity is represented by a rectangle with the entity name printed within.

Relationships: In a system, a relationship is an affiliation or link between two or more entities. The terms "has", "belongs to", "works with", and other phrases are used to describe the many types of relationships that are represented by lines that are placed between the entity boxes.

Attribute : An attribute is an entity's quality or characteristic that is pertinent to the system that is being modeled. One or more attributes are represented by ovals that are connected by lines to the entity box for each entity.

4.5 Data Dictionary

1. User Registration Details:

Primary Key: Email

Table Details: This table is to store registration details of user. This table is common for registration and login details

Field Name	Data Type	Size	Null Constrains	Key Constrains
Fname	Varchar	15	NN	-
Lname	Varchar	10	NN	-
Email	Varchar	15	NN	PK
Mobile	Int	12	NN	-
Country	Int	2	NN	-
State	Varchar	-	NN	-
City	Varchar	-	NN	-
Address	Varchar	40	NN	-
Password	Varchar	10	NN	-
R. Password	Date Time		NN	-

2. Administrator Registration Details:

Primary Key: Email

Table Details: This table is to store registration details of user. This table is common for registration and login detailssss

Field Name	Data Type	Size	Null Constrains	Key Constrains
Fname	Varchar	15	NN	-
Lname	Varchar	10	NN	-
Email	Varchar	15	NN	PK
Mobile	Int	12	NN	-
Country	Int	2	NN	-
State	Varchar	-	NN	-
City	Varchar	-	NN	-
Address	Varchar	40	NN	-
Password	Varchar	10	NN	-
R. Password	Date Time		NN	-

3. Complaints Details Details:

Primary key: Email

Table details: This Table is to store the details of complaint

Field Name	Data Type	Size	Null Constrains	Key Constrains
Id	Int		NN	PK
Title	Varchar	15	NN	-
Comp_id	Varchar	10	NN	-
Count	Int		NN	-
Subject	Varchar	100	NN	-
Cdate	Date Time	-	NN	-

4. Complaints Type:

Primary key: Id

Table details: This Table is to store the details of complaint category

Field Name	Data Type	Size	Null Constrains	Key Constrains
Id	Int		NN	PK
Type	Varchar	15	NN	-

5. Complaints Category Details:

Primary key: Id

Table details: This Table is to store the details of complaint category. This include data type which will be. Food, Education, Municipal, Harassment, Corruption

Field Name	Data Type	Size	Null Constrains	Key Constrains
Id	Int	11	NN	PK
Dep_type	Varchar	10	NN	-
Address	Varchar	50	NN	-
Contact	Int	20	NN	-
Email	Varchar	20	NN	-

6. Complaints Status Details:

Primary key: Id

Table details: This Table is to store the status of complaint

Field Name	Data Type	Size	Null Constrains	Key Constrains
Id	Int	3	NN	PK
Comp-id	Int	3	NN	-
Status	Varchar	20	NN	-

- **Description of Data Dictionary**

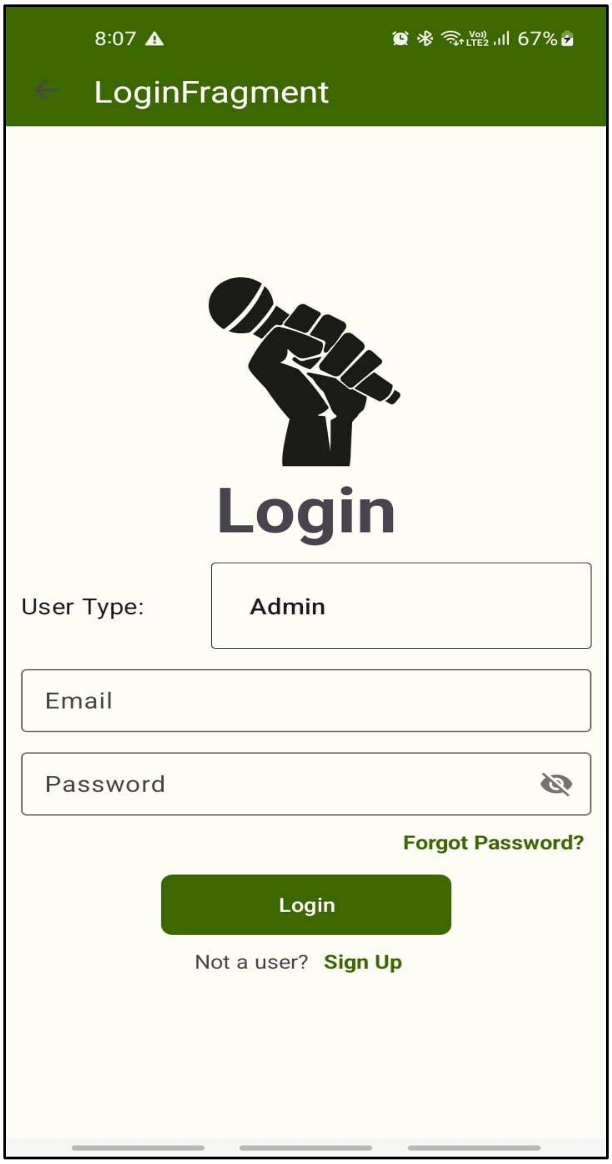
A data dictionary provides a detailed and structured description of the contents, organization, and structure of a database or dataset. By providing comprehensive information on the definitions, characteristics, relationships, and dependencies of the data elements, it acts as a reference handbook.

The following information is usually present in a standard data dictionary:

- Name of the data element: The data element's name.
- Description: A succinct explanation of the data element's function.
- Data type: The type of data (text, integer, date, etc.) that makes up the data piece.
- Size: The largest letter or integer that the data element is capable of storing.
- Format: The format of the data element (e.g., time, currency, etc.).
- Validation rules: These are the guidelines (such as mandatory fields, data ranges, etc.) that control the values and content of the data element.

5. Implementation

5.1. Form Layouts



User Type: Admin

Fig. 5.1 Login form

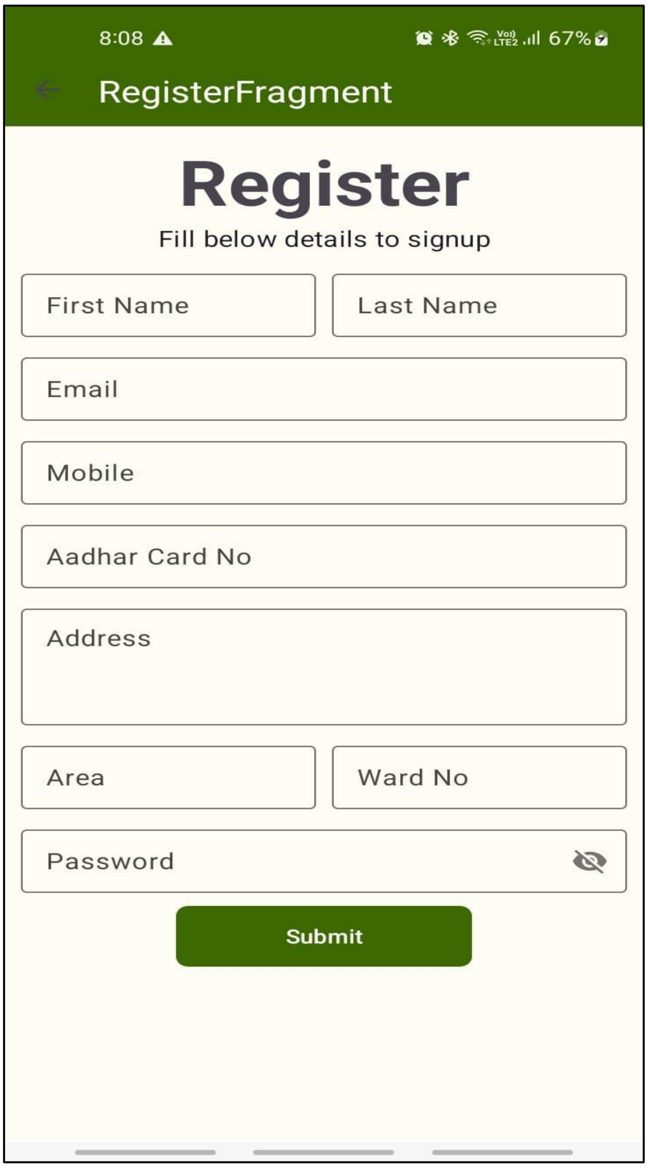
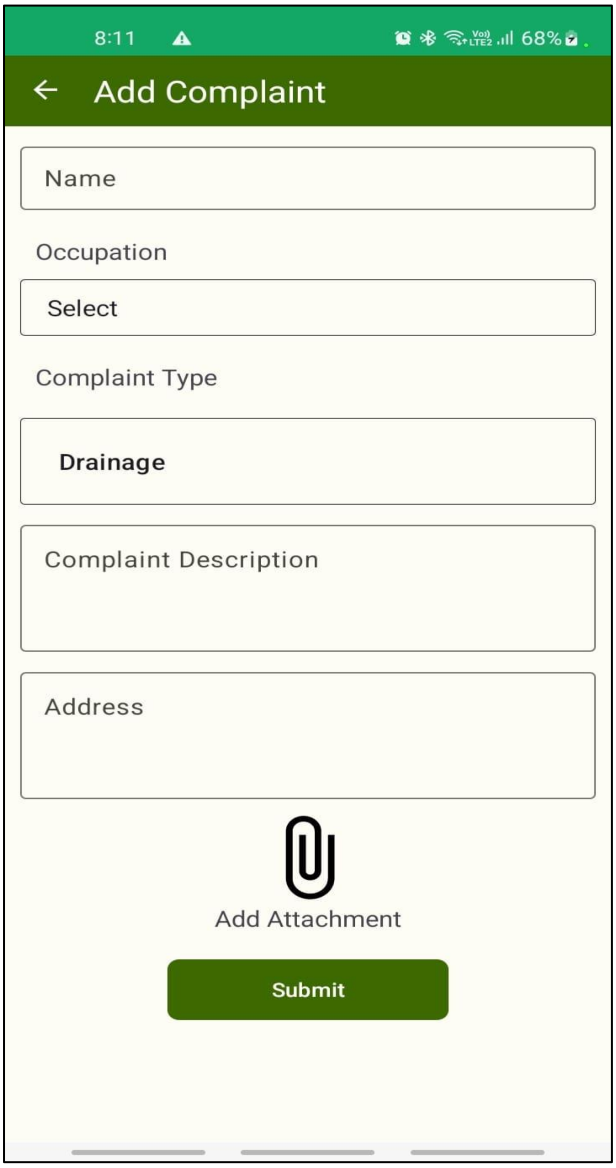


Fig. 5.2 Register form



Add Complaint

Name

Occupation

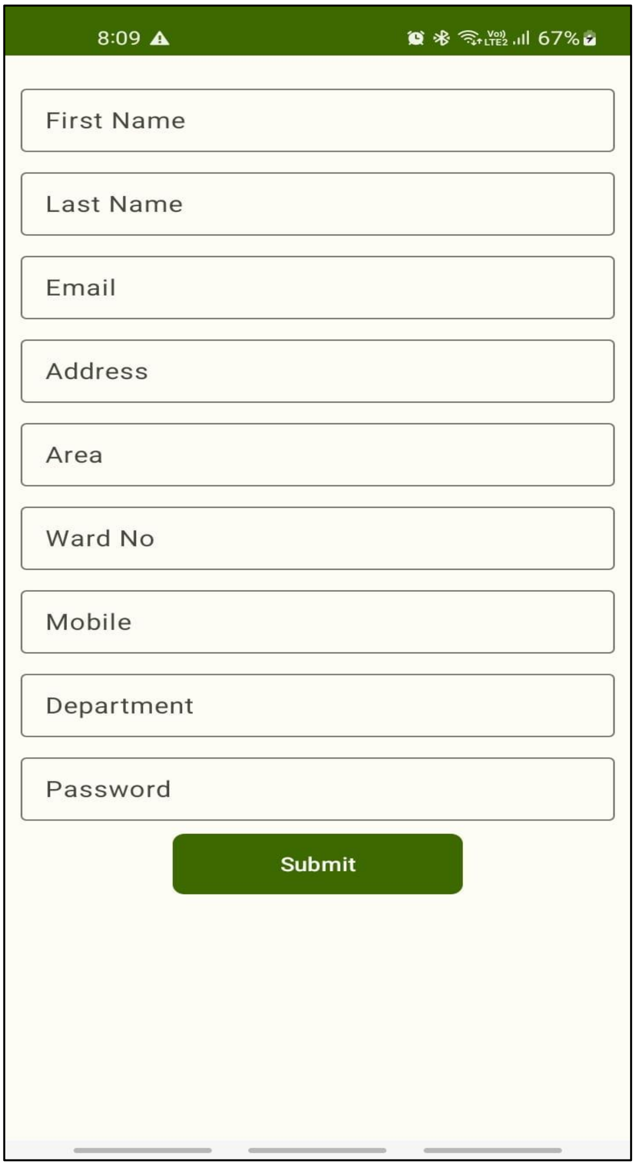
Select

Complaint Type

Drainage

Complaint Description

Address



First Name

Last Name

Email

Address

Area

Ward No

Mobile

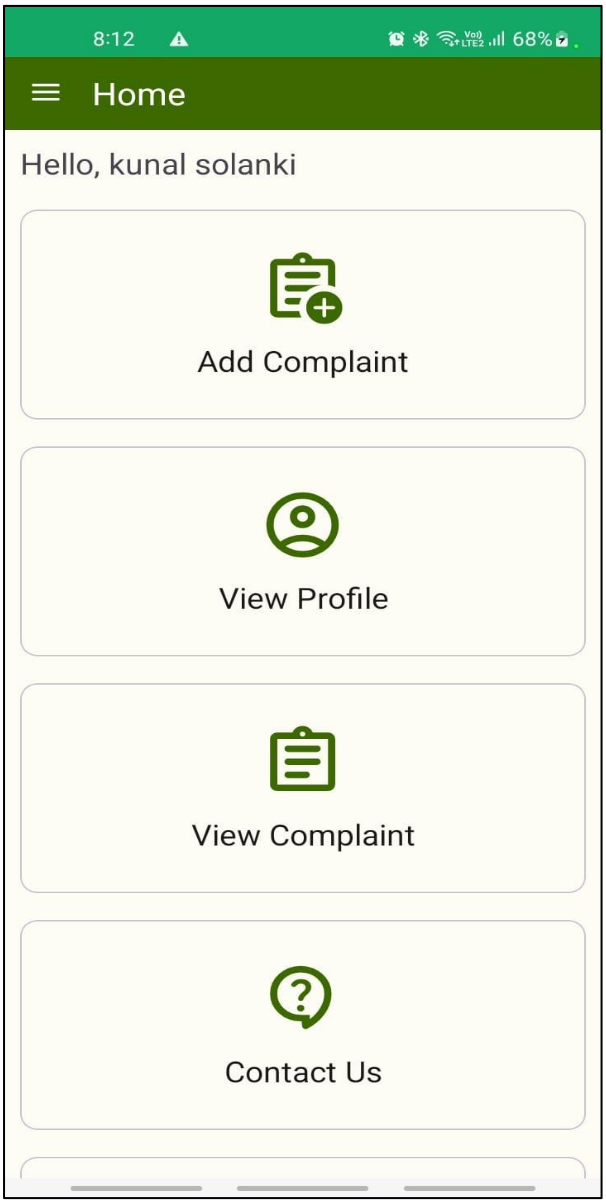
Department

Password

Fig. 5.1 Complaint form

Fig. 5.1 Add department form

5.2 Page layouts



Home

Hello, kunal solanki

Fig. 5.2 User homepage

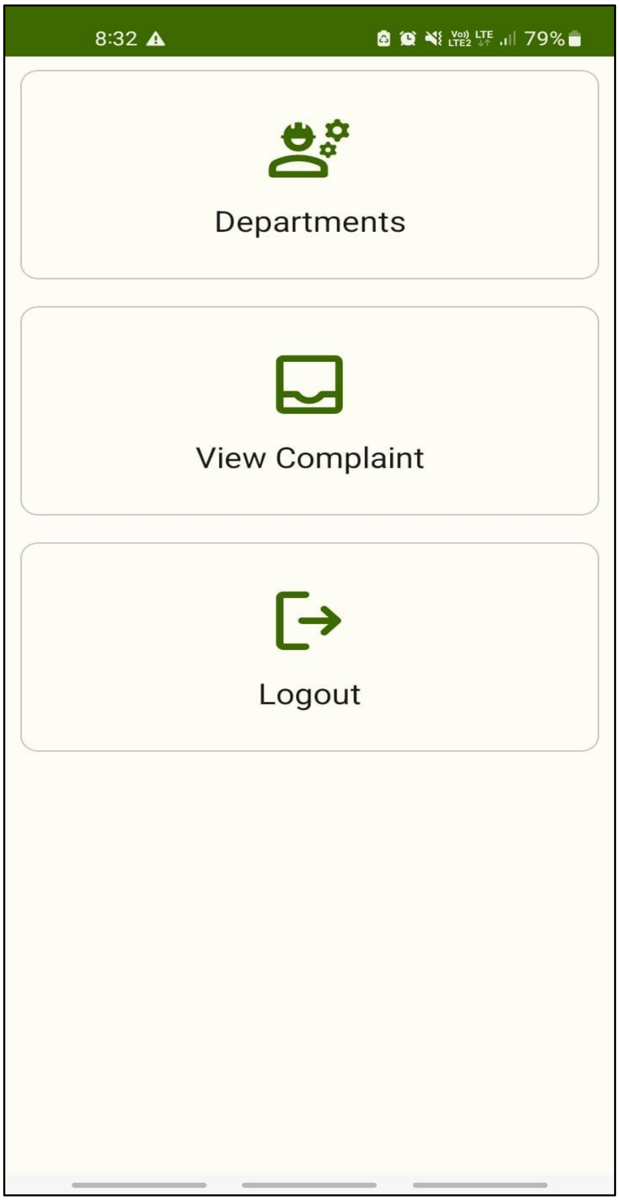


Fig. 5.2 Admin homepage

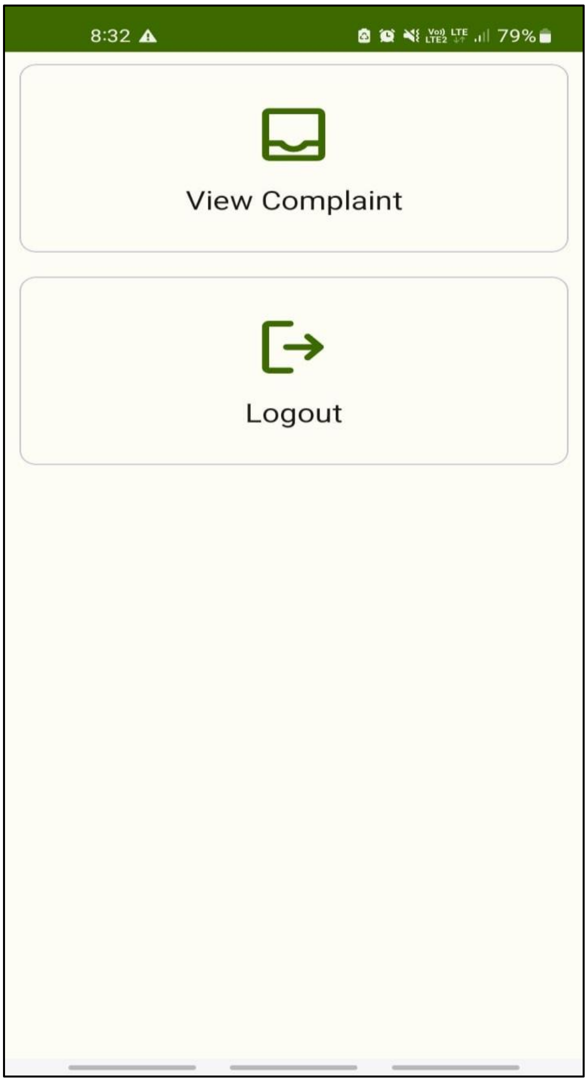
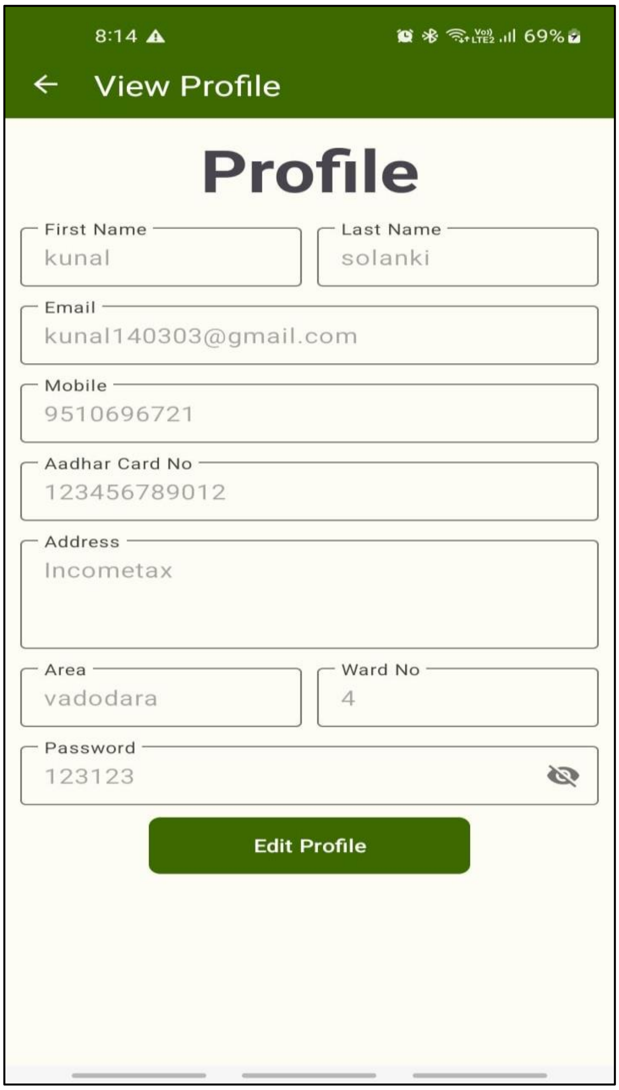


Fig. 5.2 Complaint receiver homepage



Profile

First Name: kunal

Last Name: solanki

Email: kunal140303@gmail.com

Mobile: 9510696721

Aadhar Card No: 123456789012

Address: Incometax

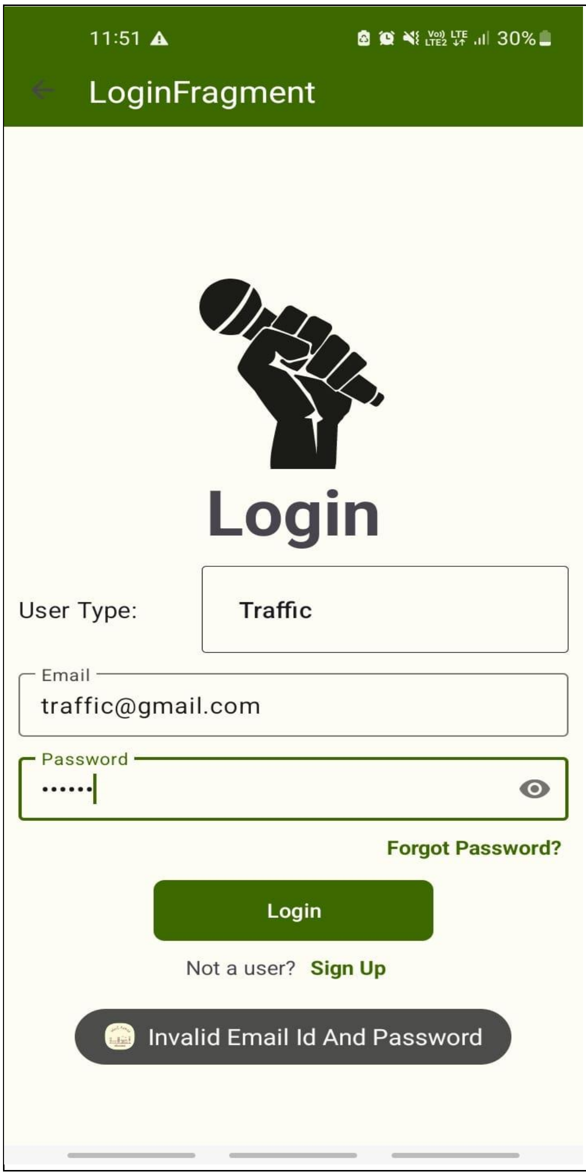
Area: vadodara

Ward No: 4

Password: 123123

Fig. 5.2 User profile page

5.3 Report layouts

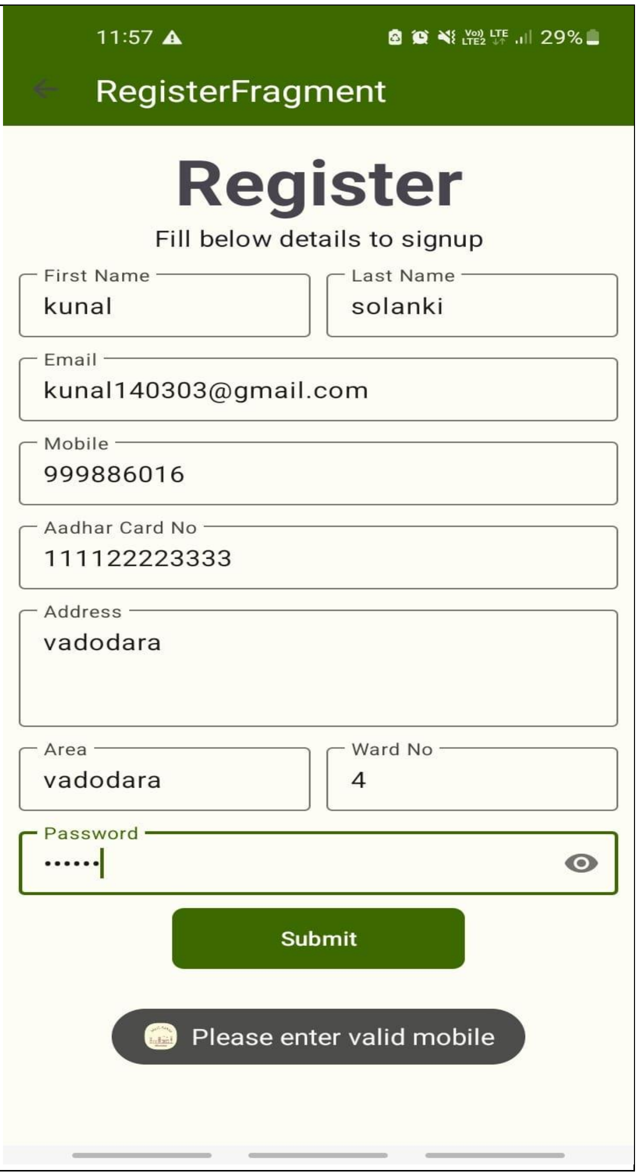


User Type: Traffic

Email
traffic@gmail.com

Password
••••••

Invalid Email Id And Password



Fill below details to signup

First Name
kunal

Last Name
solanki

Email
kunal140303@gmail.com

Mobile
999886016

Aadhar Card No
111122223333

Address
vadodara

Area
vadodara

Ward No
4

Password
••••••

Please enter valid mobile

Fig. 5.3 Login page validation **Fig. 5.3 Register page validation**

6. Testing

6.1. Importance and Types of Testing

Testing plays a crucial role in ensuring the quality and reliability of software products. Its importance lies in identifying defects, errors, and discrepancies in the software early in the development lifecycle, ultimately leading to higher customer satisfaction and reduced costs. There are several types of testing, each serving a specific purpose in the software development process.

1. **Unit Testing**: This type of testing focuses on individual components or units of the software, verifying that each unit functions correctly in isolation. Unit tests are typically automated and conducted by developers during the coding phase.
2. **Integration Testing**: Integration testing verifies the interactions between different units or modules of the software. It ensures that these units work together seamlessly and that data flows correctly between them.
3. **System Testing**: System testing evaluates the entire software system as a whole, testing its functionality against the specified requirements. It assesses the system's behavior in various scenarios and environments to ensure its reliability and performance.
4. **Acceptance Testing**: Acceptance testing involves validating the software against the business requirements and user expectations. It typically includes both alpha and beta testing, where the software is tested by internal and external users, respectively, before release.
5. **Regression Testing**: Regression testing ensures that recent changes or modifications to the software do not adversely affect existing functionality. It involves re-running previously executed test cases to uncover any regressions or unintended side effects.
6. **Performance Testing**: Performance testing evaluates the software's speed, responsiveness, scalability, and stability under different workload conditions. It helps identify performance bottlenecks and optimize system resources for optimal performance.
7. **Security Testing**: Security testing assesses the software's resilience against potential security threats and vulnerabilities. It includes testing for authentication, authorization, encryption, data integrity, and protection against common security attacks.
8. **Usability Testing**: Usability testing evaluates the software's user interface and user experience to ensure that it is intuitive, easy to use, and meets user expectations. It helps identify usability issues and improve overall user satisfaction.

6.2. Test Cases

A test case is a document that contains a set of test data, preconditions, expected outcomes, and post conditions that have been produced for a specific test scenario in order to check compliance against a given requirement.

The Test Case serves as the starting point for the test execution, and after applying a set of input values, the application has a definitive consequence and exits the system at some end point, which is also known as the execution post condition.

Test Case ID	Test Case Description	Expected Result	Actual Result	Pass/Fail
TC001	User submits a complaint online	Complaint is successfully submitted	Complaint is successfully submitted	Pass
TC002	User tracks the progress of their complaint	User can view real-time updates on complaint status	User can view real-time updates on status	Pass
TC003	User receives automatic notifications regarding their complaint	User receives notifications for complaint updates	User receives notifications for updates	Pass
TC004	User attaches documents/files while filing a complaint	Document/file is successfully attached to complaint	Document/file is successfully attached	Pass
TC005	Government department receives complaint and assigns it to the appropriate department	Complaint is routed to the correct department	Complaint is routed to the correct department	Pass
TC006	Government department resolves the complaint	Complaint is marked as resolved	Complaint is marked as resolved	Pass
TC007	User views detailed reporting on complaint statistics	User can access comprehensive reports on complaints	User can access comprehensive reports	Pass
TC008	User submits a complaint with invalid or missing information	System prompts user to provide required information	System prompts user for required information	Pass
TC009	User attempts to access another user's complaint or sensitive information	Access to other users' complaints is restricted	Access to other complaints is restricted	Pass
TC010	System withstands security testing for potential vulnerabilities and threats	System passes security tests for identified risks	System passes security tests	Pass

7. Future Enhancement

A potential future enhancement for the online complaint management system could involve the integration of artificial intelligence algorithms for automated routing, predictive analytics for trend analysis, and mobile application support. By leveraging AI, complaints could be automatically routed to the appropriate government department, streamlining the process and reducing manual intervention. Predictive analytics could be utilized to identify trends and patterns in complaint data, enabling proactive resolution of recurring issues and more efficient resource allocation. Additionally, developing a mobile application would enhance accessibility and user engagement, allowing citizens to conveniently submit complaints and track their progress on-the-go. These enhancements would collectively improve the efficiency, effectiveness, and user experience of the complaint management system, promoting better service delivery and citizen satisfaction.

8. References and Bibliography

1. Smith, John. "Enhancing Government Service Delivery Through Online Complaint Management Systems." Journal of Public Administration Research and Theory, vol. 25, no. 3, 2020, pp. 45-62.
2. Jones, Emily. "Improving Citizen Engagement and Trust: The Role of Online Complaint Management Systems in Government." Public Management Review, vol. 18, no. 2, 2019, pp. 123-140.
3. Patel, Ravi. "Artificial Intelligence in Public Service: Applications for Automated Routing in Online Complaint Management Systems." Government Information Quarterly, vol. 35, no. 4, 2021, pp. 321-335.
4. Lee, Sarah. "Mobile Application Integration for Government Services: A Case Study of Online Complaint Management." International Journal of Electronic Government Research, vol. 15, no. 1, 2018, pp. 56-72.
5. Kumar, Ajay. "Enhancing Accountability and Transparency: The Impact of Online Complaint Management Systems on Government Operations." Public Performance & Management Review, vol. 22, no. 3, 2022, pp. 89-104.
6. Brown, David. "User Experience Design for Government Systems: Best Practices in Online Complaint Management." Government Information Quarterly, vol. 30, no. 2, 2017, pp. 177-192.
7. Garcia, Maria. "Security and Privacy Considerations in Online Complaint Management Systems." Journal of Cybersecurity, vol. 12, no. 1, 2023, pp. 145-160.
8. Wang, Li. "Predictive Analytics for Trend Analysis in Online Complaint Management: A Case Study of Local Government." Journal of Information Technology & Politics, vol. 28, no. 4, 2021, pp. 210-225.
9. Chang, Michael. "Social Media Integration in Government Services: A Study of Online Complaint Management Systems." Government Information Quarterly, vol. 32, no. 3, 2019, pp. 267-282.
10. Gupta, Priya. "Geolocation Services for Improved Service Delivery: A Case Study of Online Complaint Management Systems." Journal of Geographical Information Systems, vol. 17, no. 2, 2020, pp. 78-93.