E-Society

Final Year Project

Session 2019-2023

A project submitted in partial fulfillment of the degree of

BS in Information Technology

Department of Information Technology

Faculty of Computer Science& Information Technology

The Superior University, Lahore

Spring 2023

Type (Nature of project)	[✓] **D**evelopment [ ] **R**esearch [] **R&D**
Area of specialization	
FYP ID	**FYP-BITM-F22-007**

Project Group Members				
Sr.#	**Reg. #**	**Student Name**	**Email ID**	***Signature**
(i)	BITM-F19-080	Sheraz Hussain (**Leader**)	Bitm-f19-080@superior.edu.pk	
(ii)	BITM-F19-015	Muhammad Haseeb Ali	Bitm-f19-015@superior.edu.pk	
(iii)	BITM-F19-041	Abdullah Khaliq	Bitm-f19-041@superior.edu.pk	

*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to work of others

Plagiarism Free Certificate

This is to certify that, I **Sheraz Hussain** S/D of **Khadim Hussain**, group leader of FYP under registration no **BITM-F19-080** at Information Technology Department, The Superior University, Lahore. I declare that my FYP report is checked by my supervisor.

Date: Name of Group Leader: **Sheraz Hussain** Signature: _____

Name of Supervisor: Sir Ali Raza

Designation: Lecturer

Signature: _____

HOD: Dr. Asad Ali Naqvi

Signature:_____

Project Report

E-Society

Change Record

Author(s)	Version	Date	Notes	Supervisor's Signature
Sheraz , Haseeb , Abdullah	1.0		Template 1	
Sheraz , Haseeb , Abdullah	1.2		Documentation	
Sheraz , Haseeb , Abdullah	1.3		Review of layouts	
Sheraz , Haseeb , Abdullah	1.4		Review of Coding	
Sheraz , Haseeb , Abdullah	1.5		Review of database	
Sheraz , Haseeb , Abdullah	1.0		Template 1	
Sheraz , Haseeb , Abdullah	1.2		Documentation	

APPROVAL

PROJECT SUPERVISOR

Comments: _____

_____

Name:_____

Date:_____ Signature:_____

PROJECT MANAGER

Comments: _____

_____

Date:_____ Signature:_____

HEAD OF THE DEPARTMENT

Comments: _____

_____

Date:_____ Signature:_____

Dedication

First of all, we dedicate our project to Allah Almighty

And to whom the world owes its existence

Muhammad (Peace Be upon Him)

This humble effort is dedicated to

Our beloved parents who brought us

To the level of excellence where we

Are studying today looking for most

Promising and gleaming future ahead

For which they scarified most of the

Time of their life

&

To our respected and genius teachers

Who guided us throughout academic career!

And all those people

Who have remembered us in their prayers!

A lot of thanks for all my teachers!

Acknowledgements

Above all, I owe a debt of gratitude to the All-Powerful Allah who gave us a life that is worthwhile and whom I appreciate for providing us the stamina to complete this task. This project's accomplishment and success are the result of the hard work and dedication of numerous people who provided direct or indirect support. I give them all my gratitude for their devotion.

The author would like to convey their deep gratitude to **Sir Ali Raza** , their project adviser, for his careful oversight, intellectual direction, helpful counsel, and exceptionally nice demeanor throughout the project.

Thank you to all of our instructors whose guidance has led us to this point of academic stage. Last but not least, we would want to express our sincere gratitude and respect to our parents, whose love and care helped us advance academically and kept us motivated. The writers owe a huge debt of gratitude to their siblings, whose unceasing support gave us the motivation we needed to pursue our academic goals.

Executive Summary

The proposed system has functionality to facilitate the societies. Different societies need this purposed system to fulfill their needs. It is a unique idea in Pakistan and there is not existing system in Pakistan. This system is also known as E-Society system.

According to the survey this system is only using in few countries all over the world. This system is used by society admin, state builders and also the other common people. Where state builders can add different society and also each society admin can manage their society needs and requirements.

In this project we are made a platform where society members complain any problem regard to the society and there is a chat box where the society member chat about the society faults and society advantages.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Chapter 1: Introduction

First, we developed a mobile app for welfare of people in environment. This app is use for buying and selling house in society. Also, members of society can complain there problems according to the society online through this app.

In this there a map where different people see house. If any house is sale and for buy then there is be shown. People see house location and see anything they want.

Before this people face difficulties to buy and sell the houses, they go here and there to search the house according to their requirements. Also, they face difficulties to complain about sewerage, electricity and other problems regards to the society. Through this app problems of the people are solved and also saved their time.

This app has reviews and feedback according to society complaints of society members.

1.1. Background

- The term "society" came from the 12th century French society (meaning 'company').
- This was in turn from the Latin word societies, which in turn was derived from the noun socius ("comrade, friend, ally"; adjectival form socialist) used to describe a bond or interaction between parties that are friendly, or at least civil. Without an article, the term can refer to the entirety of humanity (also: "society in general", "society at large", etc.), although those who are unfriendly or uncivil to the remainder of society in this sense may be deemed to be "antisocial". In the 1630s it was used in reference to "people bound by neighborhood and intercourse aware of living together in an ordered community".
- However, in the 18th century the Scottish economist, Adam Smith taught that a society "may subsist among different men, as among different merchants, from a sense of its utility without any mutual love or affection, if only they refrain from doing injury to each other.
- In every society organization are working to manage the society. But they are not managing the proper according to the requirement of society. So, we are providing an application through which society admin and society members can solve their problems.
- Through this app in this there a map where different people see house. If any house is sale and for buy then there is be shown. People see house location and see anything they want.

Before this people face difficulties to buy and sell the houses, they go here and there to search the house according to their requirements. Also, they face difficulties to complain about sewerage, electricity and other problems regards to the society. Through this app problems of the people are solved and also saved their time.

1.2. Motivations and Challenges

Now a day people use more mobiles for internet for different purpose. It is very easy to use for everyone. It is app only for E-society. A great new idea to help people through mobile app.

1.3. Goals and Objectives

Our main objective is to create an application that can run **on both Android and iOS** devices, by the application creating tool called **android studio with dart language**. Through this application we can solve the problems of people in daily life like they face difficulties to buy and sale house in different societies. Also the problems related to the society like sewerage, cleaning and street light problems. They can online complain these problems online through this application.

This application can generate revenue as we can sale this application to the state builders and society head

1.4. Literature Review/Existing Solutions

In existing system when someone want to buy a house then he goes here and there to see and ask about the house and the plot then they tried to visit different places and different society to buy and sell house. In different society there is not a place or a complain box where society member complains about the society fault and any other things about the society. Some people want to say about society fault but they have no place to say.

We create an app which is easy in use for everyone. In this app different state builders and society through which their requirements fulfilled.

1.5. Gap Analysis

There is no online mobile app before this create for society online. This app creates for society welfare and we can generate profit through this application. Through this application we can generate business to sale this application to state builders and also to society head.

1.6. Proposed Solution

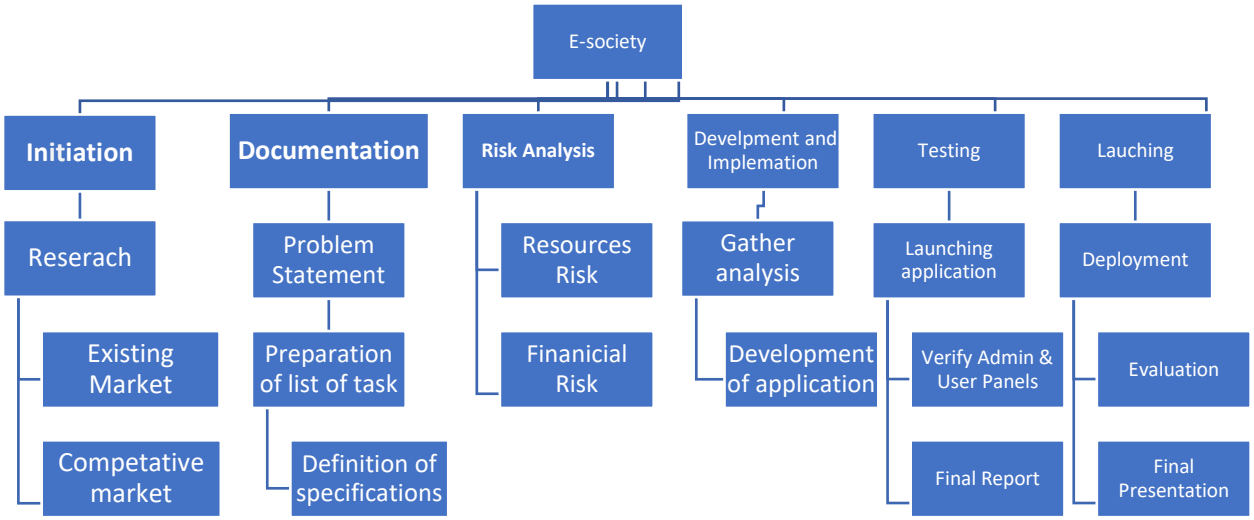
The main purpose of this app to provide easy and faster solution in new searching style in all the cities. In this era everything is moving so fast that we cannot afford any delay. Through this application looking at the problems of the people in the society, we will provide a platform to the people in the form of an app where people can go and register their complaints and through such an app, people will be able to buy and sell houses and also the location of the house. On the same way, they will be able to see both buyers and sellers from this app and can communicate with each other.

1. Registration of the user
2. Admin manage all resources

1.7. Project Plan

This is our project plan that tells about how our Android Application operate by and with following responsibility matrix, WBS and Gantt chart.

1.7.1. Work Breakdown Structure



1.7.2. Roles & Responsibility Matrix

The purpose of roles & responsibility matrix is to identify who will do what.

WBS #	WBS Deliverable	Activity #	Activity to Complete the Deliverable	Duration (# of Days)	Responsible Team Member(s) & Role(s)
1	Planning	1		5 days	Sheraz,Haseeb,Abdullah
2	Research	2		10 days	Sheraz,Haseeb,Abdullah
3	Requirements	3		5 days	Sheraz,Haseeb,Abdullah
4	Analyze systems	4		20 days	Sheraz,Haseeb,Abdullah
5	Design	5		15 days	Sheraz,Haseeb,Abdullah
6	Front end	6		30 days	Sheraz,Haseeb,Abdullah
7	Back End	7		60 days	Sheraz,Haseeb,Abdullah
8	Implementation	8		30 days	Sheraz,Haseeb,Abdullah
9	Testing	9		10 days	Sheraz,Haseeb,Abdullah
10	Documentation	10		30 days	Sheraz,Haseeb,Abdullah

1.7.3. Gantt Chart

ID	Name	Start Date	End Date	Duration
1	Requirements	Sep 08, 2022	Sep 21, 2022	10 days
2	Interviews	Sep 08, 2022	Sep 12, 2022	3 days
3	Survey	Sep 12, 2022	Sep 21, 2022	8 days
4	Design	Sep 22, 2022	Oct 11, 2022	14 days
5	Prototype	Oct 12, 2022	Nov 07, 2022	19 days
6	Implementation	Nov 08, 2022	Jan 16, 2023	50 days
7	Front End	Nov 08, 2022	Nov 25, 2022	14 days
8	Back End	Nov 25, 2022	Dec 15, 2022	15 days
9	Implementing Blockchain	Dec 16, 2022	Jan 16, 2023	22 days
10	Testing	Jan 16, 2023	Feb 03, 2023	15 days
11	Maintenance	Feb 03, 2023	Feb 16, 2023	10 days
12	Documentation	Sep 08, 2022	Feb 28, 2023	124 days

1.8. Report Outline

- The Requirement was gathered in about 15 Days.
- The Documentation was made in a whole year.
- The coding and data were implanted in almost 4 months.
- The Test was made, and errors were debugged after and through the coding period.
- Validation and finalization in added golden days.

Chapter 2

Software Requirement Specifications

Chapter 2: Software Requirement Specifications

2.1. Introduction

2.1.1. Purpose

The purpose of this Smartphone application is to address social issues. Response time should be reduced and application performance should be optimized. And spares individuals from having to wait in line, saving them precious time.

2.1.2. Document Conventions

This record is. Its degree is to portray the necessities of the Advertisement Management System and its affiliations. This report has been composed after the AMS and consider that future changes to this program ought to be incorporated into this prerequisites determination record for keeping up its helpful part. To the extent the program usage is concerned while it's not been actualized, this record can be utilized as manual for improvement of framework. CSS, VCSS property, descriptor, and pseudo-class names are specify by single quotes. Values are specifying by single quotes. Document language element names are in upper case Letters. Document language attribute names are in lower case letters and specify by Double quotes.

2.1.3. Intended Audience and Reading Suggestions

This mobile application is very scalable. This mobile application to address issues that people have. This application's goal is to ensure that people's concerns are solved by listening to them, responding to them quickly, and assuring them. We can assist clients in purchasing and selling real estate with this application. The users of this application can view the locations of residences on map that are for sale and those that are not for sale.

2.1.4. Product Scope

The scope of this project is to eliminate the necessity to visit any civilization or location. We offer a platform so that potential buyers can quickly find the home of their dreams. We give the purchaser a map that shows the location of the house. The customer can chat about the house and its price in the chat box. It is simple for a member of the community to file a complaint in order to address social issues.

2.1.5. References

Software requirement specification standard document published by IEEE.

2.2. Overall Description

2.2.1. Product Perspective

After properly registering onto the system, E-Society app will offer a platform for society members to record complaint. A user can submit a request via the application to view the residence on a map or to file a complaint. Our main goal is to save people's valuable time because it is a problem in the coffee society that people have to wait in lengthy lines to file complaints and the office's time is running out. Our second objective is to make it easier for individuals to purchase homes on the map. Additionally, we will add more features to the application in consideration of user demand and timing in order to better serve users.

2.2.2. Product Functions

Our app's functionality also includes responding to user requests as quickly as we can and assisting users in buying and selling homes using a map. Both admin and users can access this capability. Users and administrators can both download app upgrades to their mobile devices and access the app's evolving features. Through the chat box, both users and administrators can connect with one another.

2.2.3. User Classes and Characteristics

The following user of our system is present.

Administrators:

- All actions that can be taken are completely under the administrators' power.
- After registering, they check the user.
- They must stipulate the Login's rules.
- They must update and maintain the application, making changes as needed from time to time.
- They have to take care of the login's security concerns.
- Through the email, they must update the users on the status of their login and notify them of any changes.

- We can also sell it to a developer of real estate.
- We could also market it to society.

Users:

- They are the system's registered users. They have access to every feature.
- The terms login, register, etc., should be understood by the user.
- There are two categories of users: those who must register an application and those who must buy homes.
- While admin and registered users can see, alter, and add features, general users only have access to view certain features.
- They are the system's registered users. They have access to every feature.
- They can ask to see a house in a desirable area.
- They can apply to have issues with water, gas, electricity, and the internet fixed.
- For updates on the status of their application, applicants should continue to check their complaint status.

2.2.4. Operating Environment

Flutter dart is the best platform for creating apps, and it can create apps for both iOS and Android devices.

- First, for the user's development.

  -A laptop running Windows 10 is needed.

  -Having **Android Studio** already installed.

  -Containing the **flutter plug-in** installed.

  -Containing a **dart plug-in** installed.

  -Using the **Memo Play** emulator installed, it allows you to run applications.
- After the app's development, for making the e-society app compatible with Android Software (creating an app for Android phones and IOS):

  -A Smartphone running the most recent Android OS version.
- After the e-society app was created, the following steps were taken to build an **iOS-compatible** version of the app (building an app for iPhone and iPad):

  -The latest version of android studio installed in PC.

-An iOS device with the latest version installed.

2.2.5. Design and Implementation Constraints

The following limitations pertain to the execution and design of our senior project:

- This e-society app cannot be used on an Android Smartphone with an OS version lower than 7.0 installed.
- This e-society app cannot be used on an iOS device with an iOS version lower than 8.
- The flutter dart software must be at least version 3.0 or above and installed on a laptop running Windows 10.

The e-society app must complete its development, testing and documentation till the end of 8th Semester.

2.2.6. User Documentation

Only the Apple Store or Google Play Store will support this system.

- Both iPhone and Android devices can access it.
- This application requires a powerful processor to run.
- The processor should operate at 2.8 GHz or higher.
- The Smartphone RAM should be 64 GB or more.

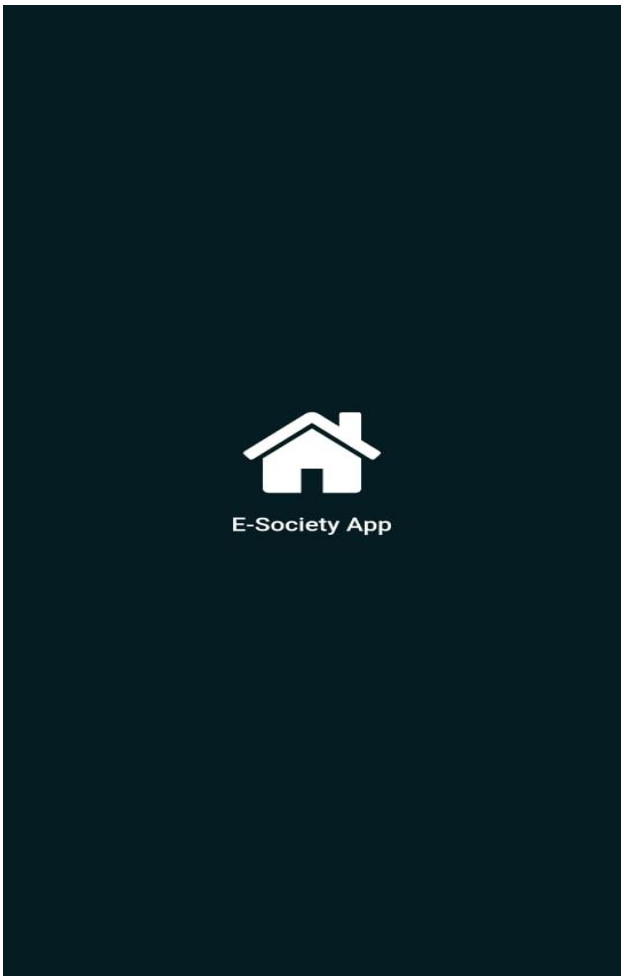
2.2.7. Assumptions and Dependencies

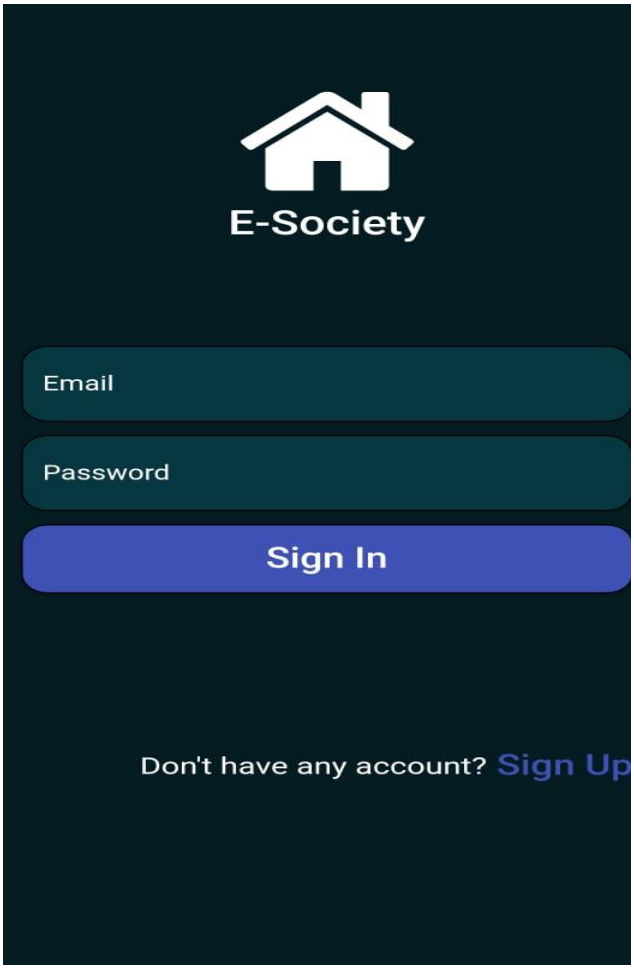
- Performance of the project and project member availability: If one of our team members becomes unavailable due to an essential or urgent matter, the performance of the development may be impacted.
- The user should be familiar with the fundamentals of using a computer and a Smartphone.
- Application usage is being watched by the administrator.

User has a fundamental understanding of how to use a computer and software.

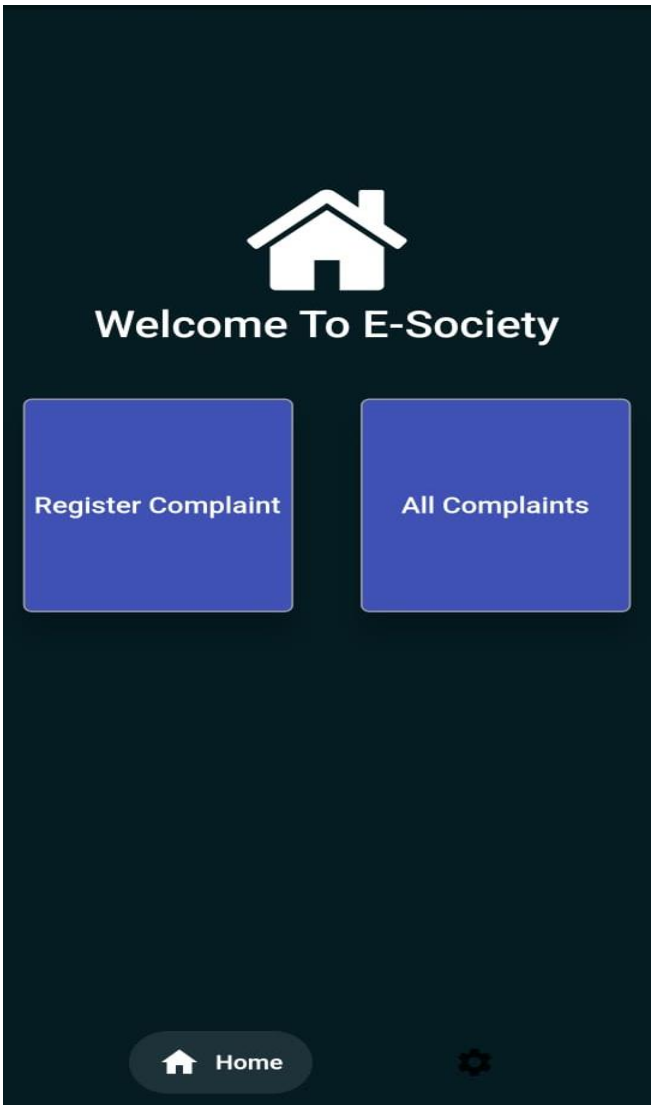
2.3. External Interface Requirements

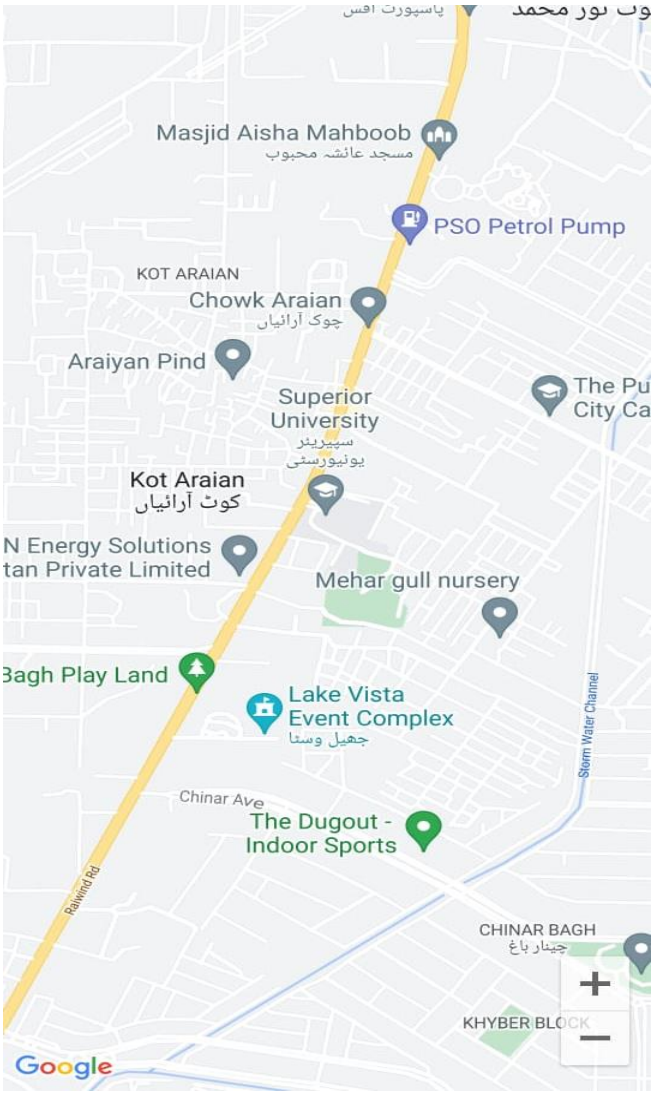
2.3.1. User Interfaces




E-Society
Email
Password
Sign In
Don't have any account? Sign Up


E-Society
Name
Email
Phone
Password
Is Admin
Sign Up


Welcome To E-Society
Register Complaint
All Complaints
Home



2.3.2. Hardware Interfaces

Since this is an online system, the hardware components needed for internet connectivity will be used as the hardware interface. In the table below, these have been listed.

WLAN	WAN	LAN	Ethernet cross-cable	Modem	Router

2.3.3. Software Interfaces

- **External interfaces for machines**.

  -Since calculation would be done on Windows OS and Android OS, an external machine interface is not necessary.

- **Interfaces to external systems**

  -This will just cover the products' Firebase database.

- **Human interface**

  -A dart-coded graphical user interface will be used by the app.

2.3.4. Communications Interfaces

The channels of communication with our users are as follows:

- Through the Apple Store or Play Store, customers will be able to rate our app and provide textual feedback.
- Users will be able to submit complaints using our e-society app and provide feedback or issue reports on the address that will be supplied both in the app and on the applications store when installing our e-society app.

2.4. System Features

2.4.1. System Feature

The system will be made up of various modules, each of which has been independently designed and has its own features.

Register Account

The user must be registered on the system if they want to purchase homes or file complaints.

Response Sequences

- In order to begin the registration procedure, the user must first click the button.
- The user is prompted by the system to enter his or her first and last names, address, and email address. both their confirm password and password.
- User fills out fields.
- The user's new account is created by the system.

Login/Logout Account

Description and Priority

The user login to the system using this feature. After registering, a user must login into the system using his user's name and password. If they are fake, the user cannot access the system.

Response Sequences

- In order to begin the login procedure, the user clicks the button.
- User is prompted by system to enter password and email.
- Information is verified by the system.
- The user sees the account home page on the system.
- The process of logging out is started when the user clicks the button.

2.5. Other Nonfunctional Requirements

2.5.1. Performance Requirements

- **Response time:** In 4-5 seconds, the home page had loaded.
- **Availability:** 99.96% of the time, the website will be accessible.
- **Throughput:** Orders per hour or page views per second.
- **Capacity:** Maximum throughput that may be attained with respect to availability and reaction time.

2.5.2. Safety Requirements

- The data of a user cannot be recovered if an administrator deletes a record or user account. The system issues a warning before deleting the record to address this issue.
- There will be safety disclaimers for both students and teachers.
- If a user encounters any issues while using the app, he or she should email the website's administrator.

2.5.3. Security Requirements

One of the main issues of today is security. The security criteria on which our system is built are as follows:

- The data of kids and teachers is safe thanks to this software.
- After user info has been verified, the user will be registered.

- Authenticated login and exit. Renter must request a new credential through the sign-up email if they forget their account credentials.
- It will be secure in the admin area.

2.5.4. Software Quality Attributes

- **Usability**: It should be simple to use and understand how to utilize this e-society app.
- **Correctness**: When a user searches for something but misspells it, the system suggests auto complete searches or searches that are related to that spelling.
- **Portability**: This app is compatible with a variety of gadgets, including tablets and smart phones.
- **Robustness**: The system offers the app the ability to rewrite the user's name and password if the user forgets their password, for example. This app can handle numerous issues on its own.
- **Flexibility**: After development, this app's functionality can be expanded because the documentation is already available.
- **Performance:** The speed at which app pages load onto a client's Smartphone is quite fast.
- **Reliability:** This app is trustworthy**.**
- **Maintainability:** The operation of this e-society application will be maintained by the app's administrator.
- **Security:** Use of this app is safe enough. Users' information is secure with this app.

2.5.5. Business Rules

- User must first join up in order to execute any crude operations.
- The user must abide by the app terms and conditions.

2.6. Other Requirements

- User must first join up in order to execute any crude operations.
- The user must abide by the app terms and conditions.

Chapter 3

Use Case Analysis

Chapter 3: System Analysis

This chapter discusses system use scenarios and how various actors engage with the system. This system is all about the use cases we create for it, the roles we give users, and how we classify them. For every role, we define every phrase related to the use case model and use case diagram. Finally, we dress up a use case model for our idea.

3.1. Use Case Model

At its most basic level, a use case diagram is an illustration of a user's engagement with the system that demonstrates the connection between the user and the many use cases in which the user is engaged. A use case diagram, which is frequently supplemented by other types of diagrams, can be used to identify the various system users and use cases. Either circles or ellipses are used to depict the use cases.

Use case model for Admin:

Figure 7: use case of admin

Fully dressed use case

Here is the filly dressed use case where admin, user and state builder can interact each other.

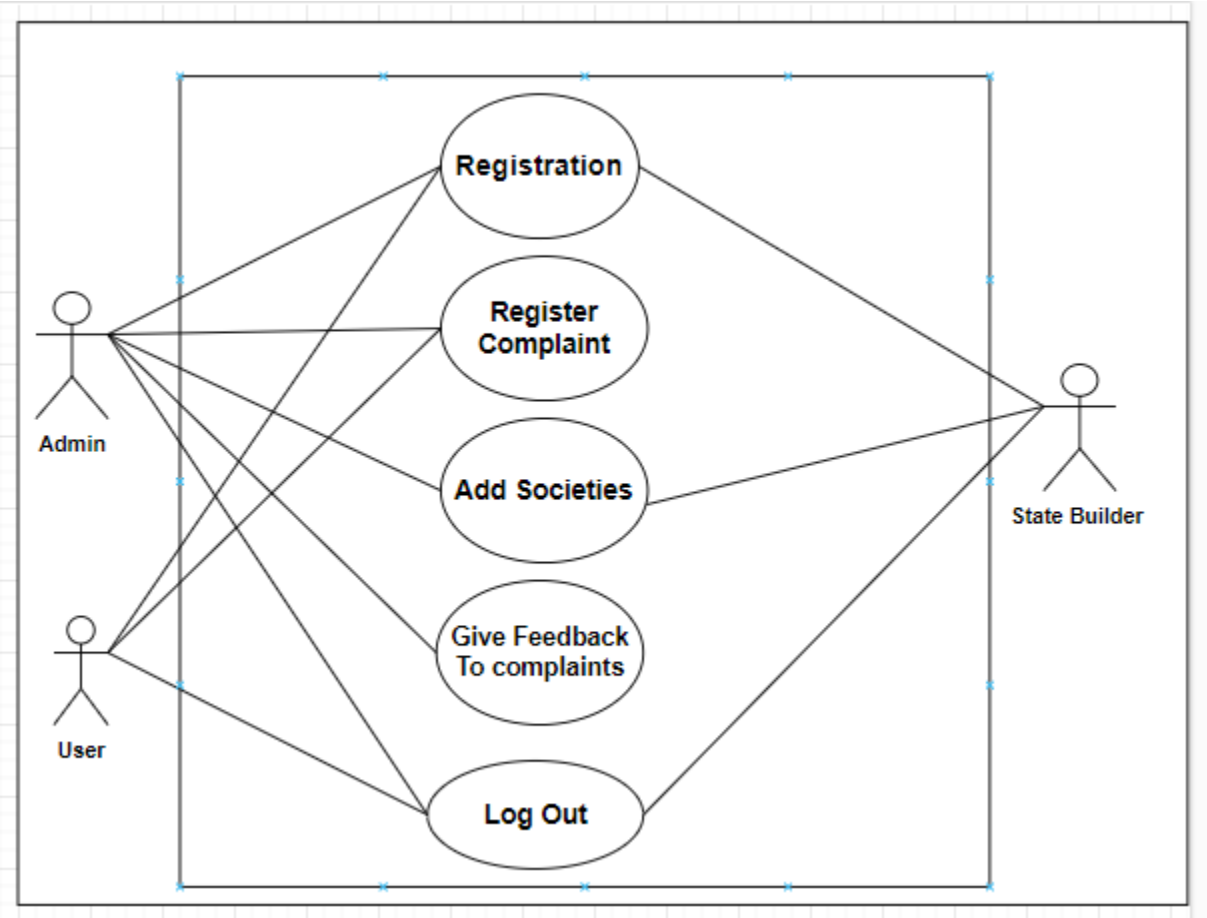
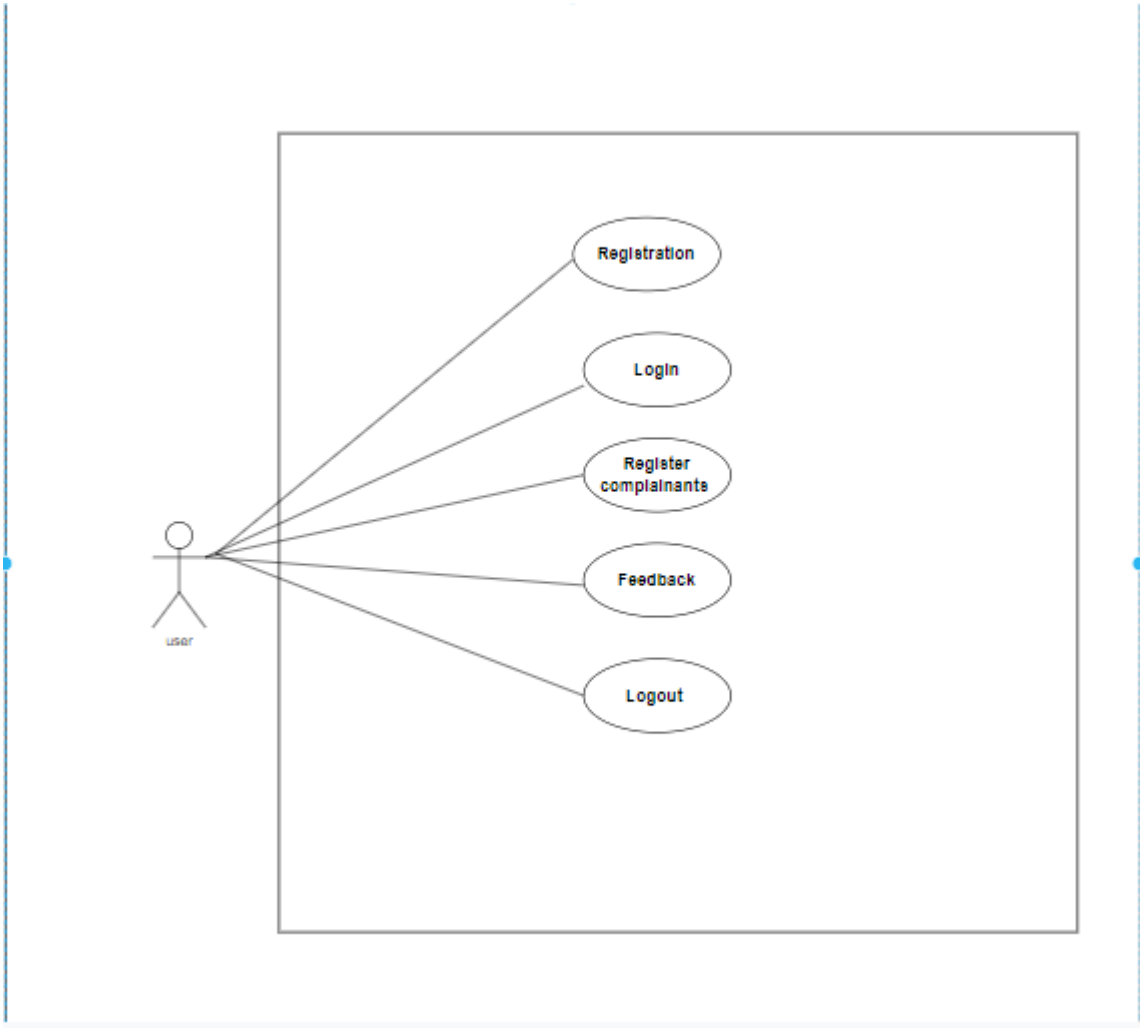


Figure 8: fully dressed use case

Use Case of user



3.2. Use Case Descriptions

Name	Register
Brief Description	A user of the System creates an account
Actor(s)	Admin
Flow of Events	
Basic Flow	
1. When a system user visits the login page while not logged in to the system, this use case begins. 2. The system requests the user's username and password or fresh account registration.	

3. The user opts for registration. 4. The system requests the user's username, password, and other information during registration. 5. The user fills out their data. The system validates the data and establishes the account. The use case ends.	
Alternate Flows	
Title	**Description**
Cancel Registration	1. The option to cancel is chosen by the user. 2. Without logging in, the system sends the user back to the home page, and all data input is deleted.
Invalid Information Entered	1. After inputting the information, the system requested, the user clicks submit. 2. Information is displayed by the system along with the proper notice to correct inaccurate information. 3. User enters data once more.
Pre-Conditions	
Title	**Description**
System	The user must possess the System Key in order to log in.
Internet	The user must have access to the system's network in order to log in.
Post-Conditions	
Title	**Description**
Success	After successfully entering their information, the user is taken back to the homepage as a logged-in user.
Failure	User gets sent back to the home page as a Guest if they are unable to log in for one or more reasons.
Extension Points The user must once check the internet connection if they are unable to log in for one or more reasons.	

Name	Log In
Brief Description	The System user logs in to the System.
Actor(s)	Logged In User
Flow of Events	
Basic Flow	

1. When a system user visits the login page while not logged in to the system, this use case begins. 2. The system requests the user's username and password or fresh account registration. *3.* Input is made by the user of their username and password. *4.* The system verifies that the username and password entered are correct and that the appropriate password is being typed for the entered username, ensuring that the username entered is a legitimate username in the system. *5.* As soon as the user logs in, they are taken back to the main page as a Logged in User. *6.* Use case is completed.	
Alternate Flows	
Title	**Description**
User Fails Authentication	1. The following happens if the user enters an invalid username or password: 2. The system provides an explanation of the User's unsuccessful authentication. 3. The system offers the User suggestions for adjustments that must be made in order for the User to successfully complete authentication. 4. The computer asks the user to enter the correct data again. 5. Where the user enters fresh data, the Basic Flow continues.
Pre-Conditions	
Title	**Description**
Register	A System user registers for an account.
Post-Conditions	
Title	**Description**
Success	The system displays a home page based on the user type after the user has been authorized.
Failure **Extension Points**	Because of one or more issues, the user cannot log in. The user is responsible for verifying that the username and password they have entered are both valid in the system.
Name	Consumer Request for Buying House.
Brief Description	The customer, who requests to purchase a home in their closest market region, may be a whole family or a single person.
Actor(s)	Consumer

Flow of Events	
Basic Flow	
1. This use case begins when a student uses the system's "Application Form" feature. *2.* The system asks the user to enter the appropriate information to get registered in the system (e.g., Student Name, Father Name, Gender, CNIC, DOB, etc...). *3.* User enters appropriate information and clicks submit. *4.* System validates student information. *5.* User is returned to User panel. *6.* This Use Case ends.	
Alternate Flows	
Title	**Description**
Invalid Information Entered	1. The system displays an “Invalid Information” error after the user submits the information and asks the User to re-enter the information. 2. User re-enters information and clicks submit.
Cancel Submitted Application Form	1. Consumer clicks cancel after selecting the “Application Form” feature. 2. System returns consumer to the consumer panel.
Pre-Conditions	
Title	**Description**
Register	A user of the System creates an account
Login	The System user logs in to the System.
Post-Conditions	
Title	**Description**
Success	The new user information is added into the database.
Failure	User information is not added for one or more reasons.
Extension Points	
If the consumer information form is not submitted for one or more reasons the User must re-entered or fill valid information in all the requisite information and re-submit the application form.	

Table 5 : use case of consumer

Chapter 4

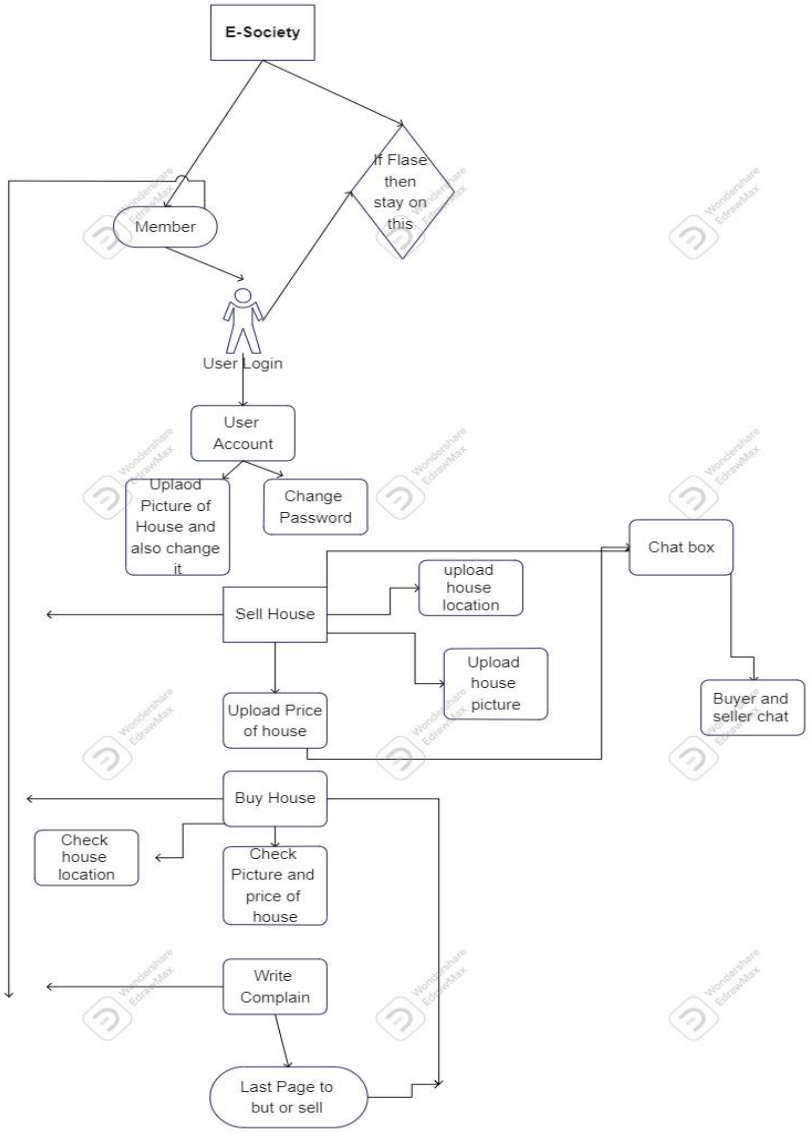
System Design

Chapter 4: System Design

In this chapter we describe our system design that how we design our system for people ease. The design is so easy that everyone can use it without any worry. The design is so attractive that there is very simple way. We connect database with our system that we can store data of our users. Systems design is the process of defining a system's components, including modules, architecture, components, their interfaces, and data, depending on the requirements that have been given. Interfaces and data for an electronic control system that are designed to meet specific criteria. It is possible to think of system design as the application of system theory to the creation of products. The fields of system analysis, system architecture, and system engineering have certain areas of overlap. A methodical approach to developing and engineering systems is necessary for system design. You must consider every aspect of an infrastructure when creating a good system design, from the hardware and software to the data and the way it is stored.

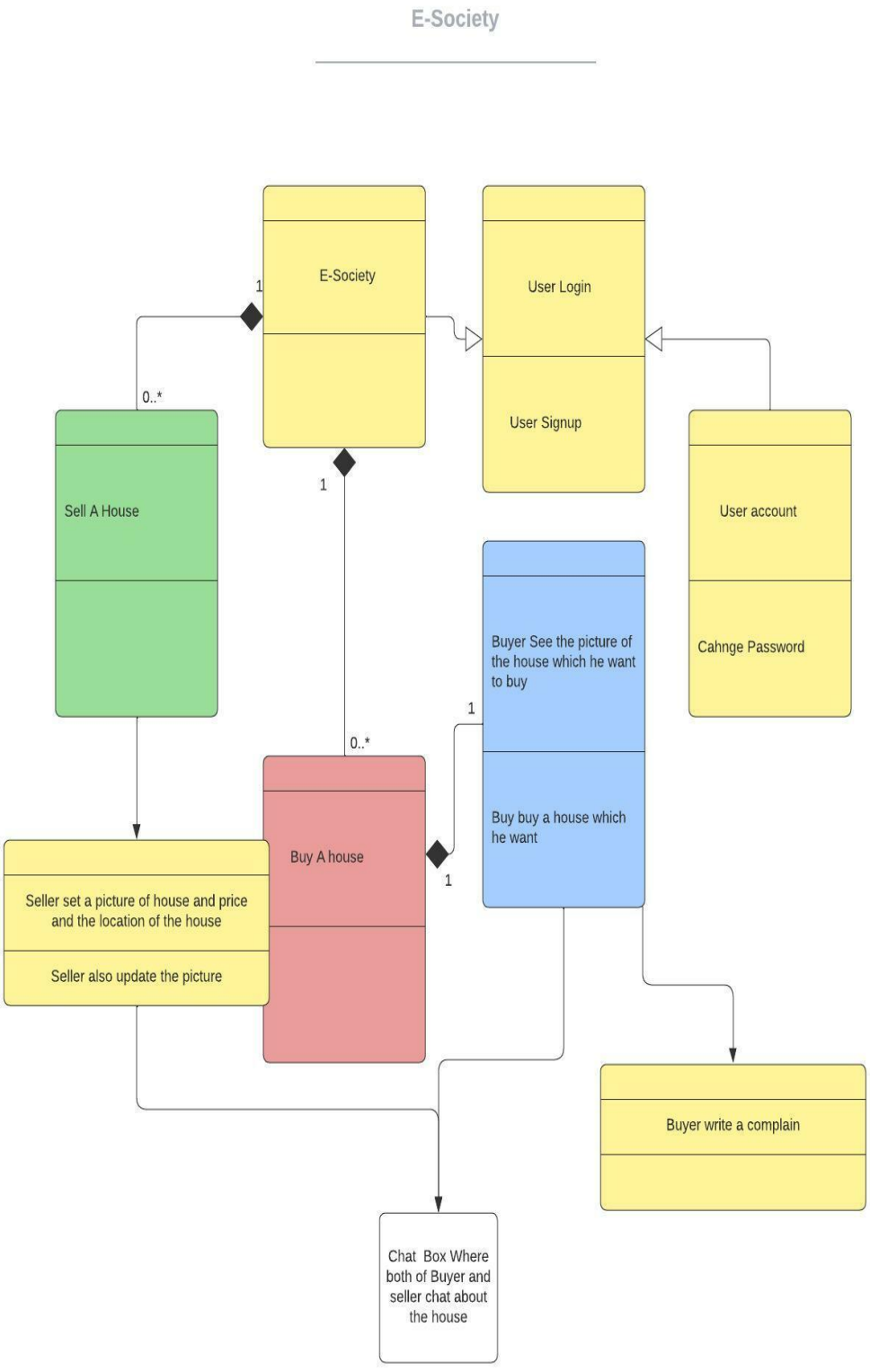
4.1. Architecture Diagram

In this we describe our architecture diagram of our system. A visual depiction that depicts the actual physical implementation of a software system's components is called an architecture diagram. It displays the relationships, constraints, and boundaries between each piece as well as the overall structure of the software system. First there is a title page after that there is a login page or signup page. After that there is user account where the user updates the house picture and user also changes it account password. If the login is false then it stays on the same page it doesn't authorize the user. When the user successful login then it goes on user account. After that there is a page of selling house where the user adds the house information and the price on which the user wants to sale. On this user can also upload picture of the house and location of house. After that there is a page where user can buy house also. You can check pictures and prices of the houses. A user can check all the pictures of related houses they want to buy and can also check the price of the house they are interested and are in their budget. This will make ease for the buyers to buy house according to his interest and location. After that there is a page of complain where user can complain about the society and their service.


E-Society
If Flase then stay on this
Member
User Login
User Account
Uplaod Picture of House and also change it
Change Password
Chat box
upload house location
Sell House
Upload house picture
Buyer and seller chat
Upload Price of house
Buy House
Check house location
Check Picture and price of house
Write Complain
Last Page to but or sell

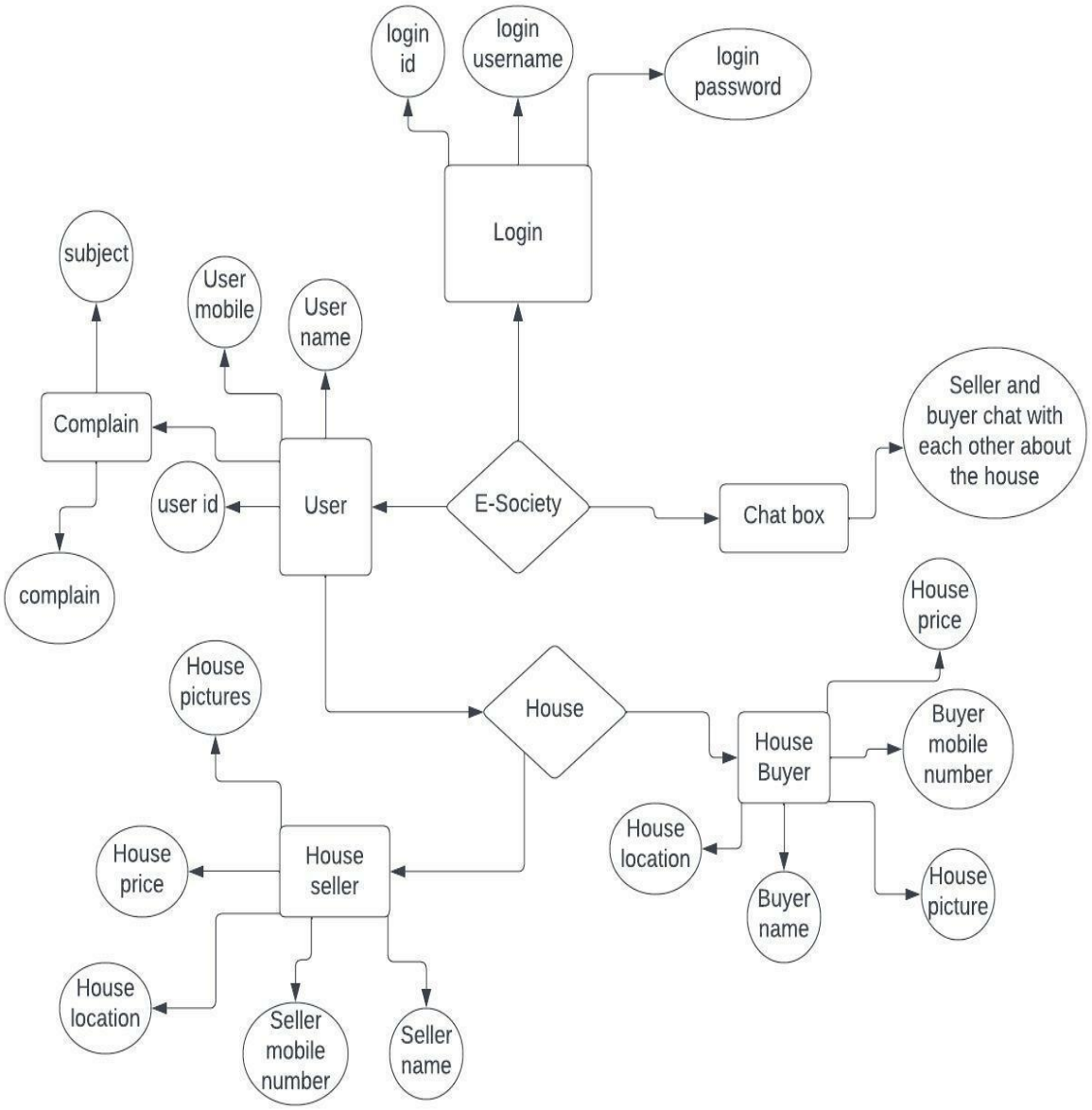
4.2. Domain Model

A domain model, as opposed to software or databases, is a graphical depiction of real-world concepts or information flow. Domain models display entities or things together with their relationships. Different domains can be modeled using a variety of diagrams and methods. To illustrate how entities, interact, it's crucial to create effective diagrams. A domain model is a conceptual representation of the domain that includes both data and behavior. A domain model is a set of abstractions that highlights particular features of a field of expertise, power, or activity. Then, issues pertaining to our e-society can be resolved using the model. The meaningful real-world concepts relevant to the domain that need to be modeled in software are represented by the domain model. The concepts cover the data used in the business and the rules that are applied to the data by the business. A domain model makes use of the domain's natural language. This model describes the working and relation of each process with others. There is different model which are connected and work together.

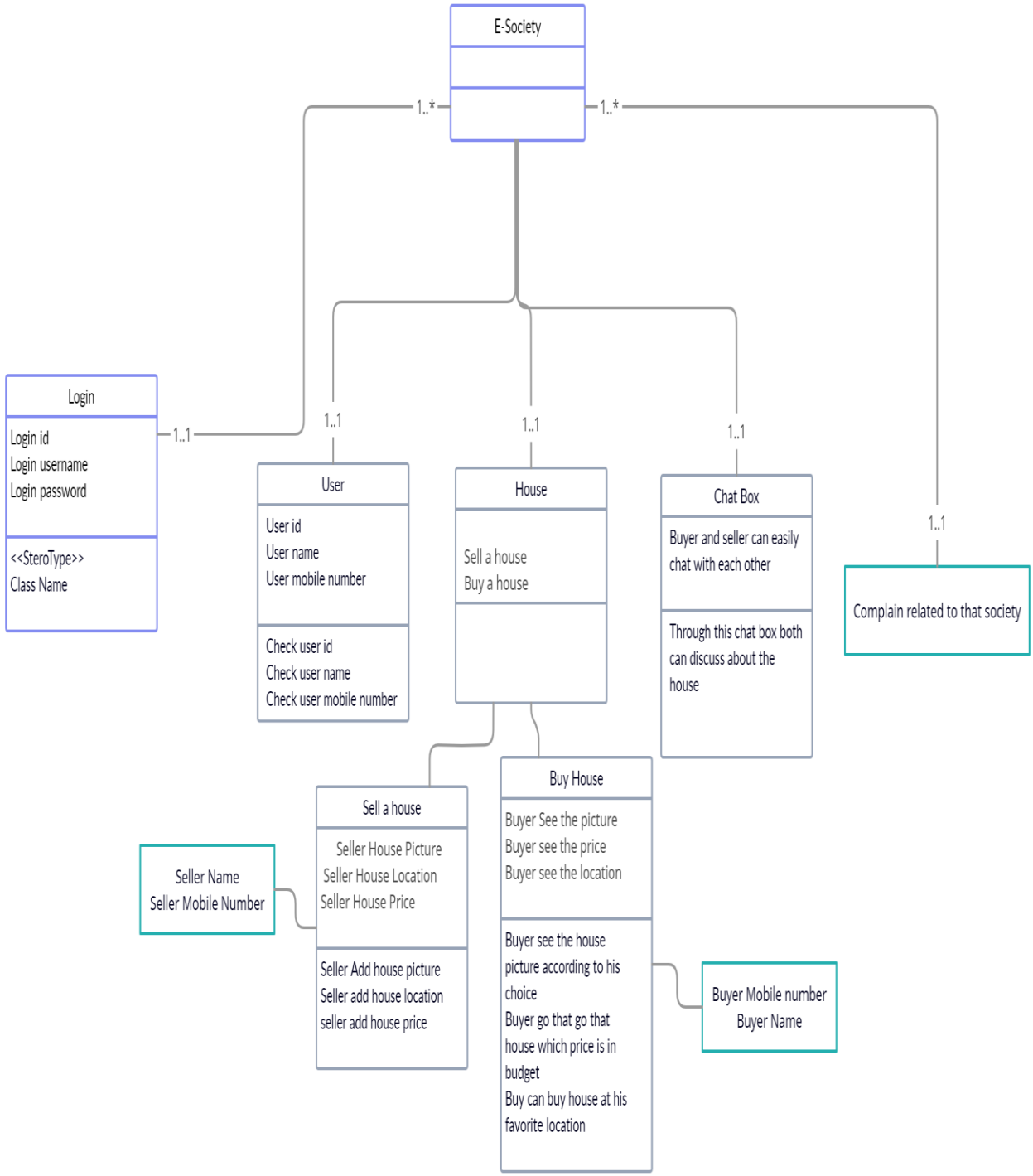

E-Society
E-Society
User Login
User Signup
User account
Cahnge Password
Sell A House
Buy A house
Buyer See the picture of the house which he want to buy
Buy buy a house which he want
Seller set a picture of house and price and the location of the house
Seller also update the picture
Buyer write a complain
Chat Box Where both of Buyer and seller chat about the house
1
0..*
1
0..*
1
1

4.3. Entity Relationship Diagram with data dictionary

An entity relationship model (ERM), also known as an entity relationship diagram (ERD), is a visual representation that shows relationships between various entities, such as individuals, things, locations, ideas, or events. An Entity Relationship (ER) Diagram is a form of flowchart that shows the relationships between entities like people, things, or concepts within a system. Rectangles, diamonds, ovals, and connecting lines are just a few of the symbols that are used in ER Models to represent how entities, relationships, and their properties are interconnected. Entity refers to a distinguishable thing that can have data saved about it, such as a person, object, concept, or event. In this entity in this our entity is user, house seller, chat box, house, house buyer, and login. Strong and weak attributes are categories for entities. In this user has relationship with user id, user mobile number, user name. And there are other entities which has relation with attributes. An attribute is a characteristics or property of an entity. Relation is often shown by circle or oval. The house buyer is an entity which has relationship with house location, house price, house location, buyer name. This is a data modeling technique that uses graphics to represent the entities in an information system and their interactions. The entity framework infrastructure is represented by an ERD, which is a conceptual and representational model of data. A business concept will be explained in greater detail using the Data Dictionary, which includes common definitions of data elements, their meanings, and permissible values. Data elements like entities and attribute are listed in a data dictionary together with explanations of their attributes.


login id
login username
login password
Login
subject
User mobile
User name
Complain
user id
User
E-Society
Chat box
Seller and buyer chat with each other about the house
complain
House price
House pictures
House
House Buyer
Buyer mobile number
House location
Buyer name
House picture
House price
House seller
House location
Seller mobile number
Seller name

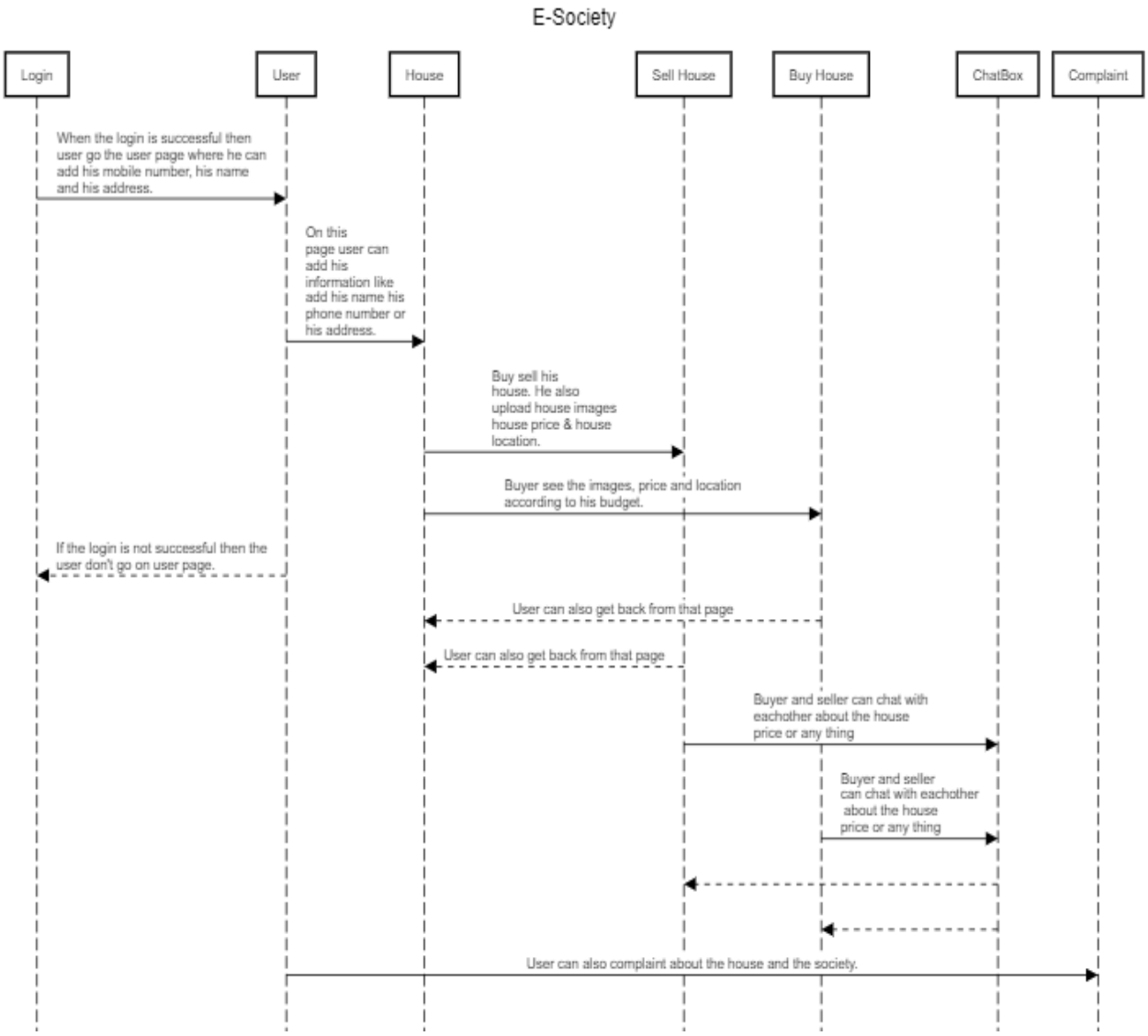
4.4. Class Diagram



A description of a collection of items with related functions in the system. One of the UML diagram types used to describe static diagrams is the class diagram, which maps system structure utilizing classes, attributes, relations, and operations amongst distinct objects. A class diagram contains different classes, each of which is divided into three parts: the first part of the class diagram contains the class name, which is the name of the class or entity that participated in the activity; the second part of the class diagram contains the class attributes; the third part of the class diagram contains the class operations; It contains the relationships between the classes. In this class diagram there is our platform name after that our attributes are connected to our class. At the third part there is our operation which we are performing on our attributes.

4.5. Sequence / Collaboration Diagram

A sequence diagram is a diagram created using the Unified Modeling Language (UML) that shows the flow of messages sent and received by objects during an interaction. A group of objects that are represented by lifelines and the messages they exchange over the course of an interaction make up a sequence diagram. A sequence diagram, also known as a system sequence diagram, displays process interactions in chronological order. In this diagram every process works with a sequence. First there is a login if login is successful then goes to user page otherwise it stays on login page. After login page there is a page of user where user can add his name his mobile number and his address. In this there is also a chat box where buyer and seller chat about the house and they also complain about the house and the society.


E-Society
Login
User
House
Sell House
Buy House
ChatBox
Complaint
When the login is successful then user go the user page where he can add his mobile number, his name and his address.
On this page user can add his information like add his name his phone number or his address.
Buy sell his house. He also upload house images house price & house location.
Buyer see the images, price and location according to his budget.
If the login is not successful then the user don't go on user page.
User can also get back from that page
User can also get back from that page
Buyer and seller can chat with eachother about the house price or any thing
Buyer and seller can chat with eachother about the house price or any thing
User can also complaint about the house and the society.

4.6. Operation contracts

- **Operation Contracts #1**

Name: Register.

Responsibilities: To register the new user.

Cross References: Use case: Register.

Exceptions: None.

Preconditions: Register interface must be opened and user must enter valid information.

Post conditions: Message pop-ups that you are successfully signed up.

- **Operation Contracts #2**

Name: Login.

Responsibilities: A user is logged in.

Cross References: Use case Login.

Exceptions: Invalid user name/email or password.

Pre conditions : User must have account.

Post conditions: User successfully login.

- **Operation Contracts #3**

Name: Logout.

Responsibilities: User is logged out.

Cross References: Use case :Logout.

Exceptions: None.

Pre conditions: User must be login.

Post conditions: Log out successfully.

- **Operation Contracts #4**

Name: Complain

Responsibilities: User must login and complaint.

Cross References: Use Case complain.

Exceptions: None.

Pre conditions: Must be logged in.

Post conditions: complaint successfully.

- **Operation Contracts #5**

Name: View complaint

Responsibilities: View detail of the complaint.

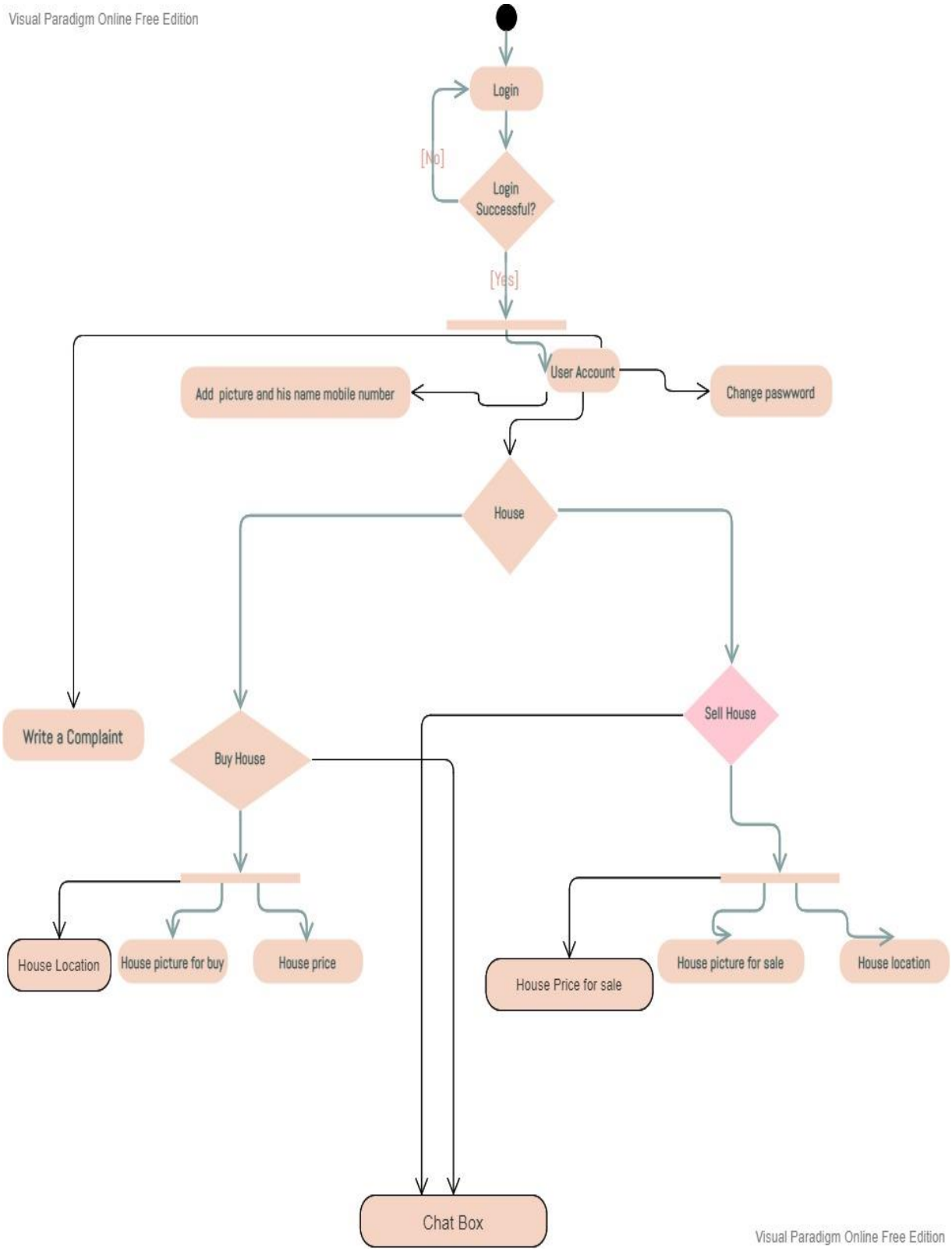
Cross References: Use Case View complaint.

Exceptions: None.

Pre conditions: User must login.

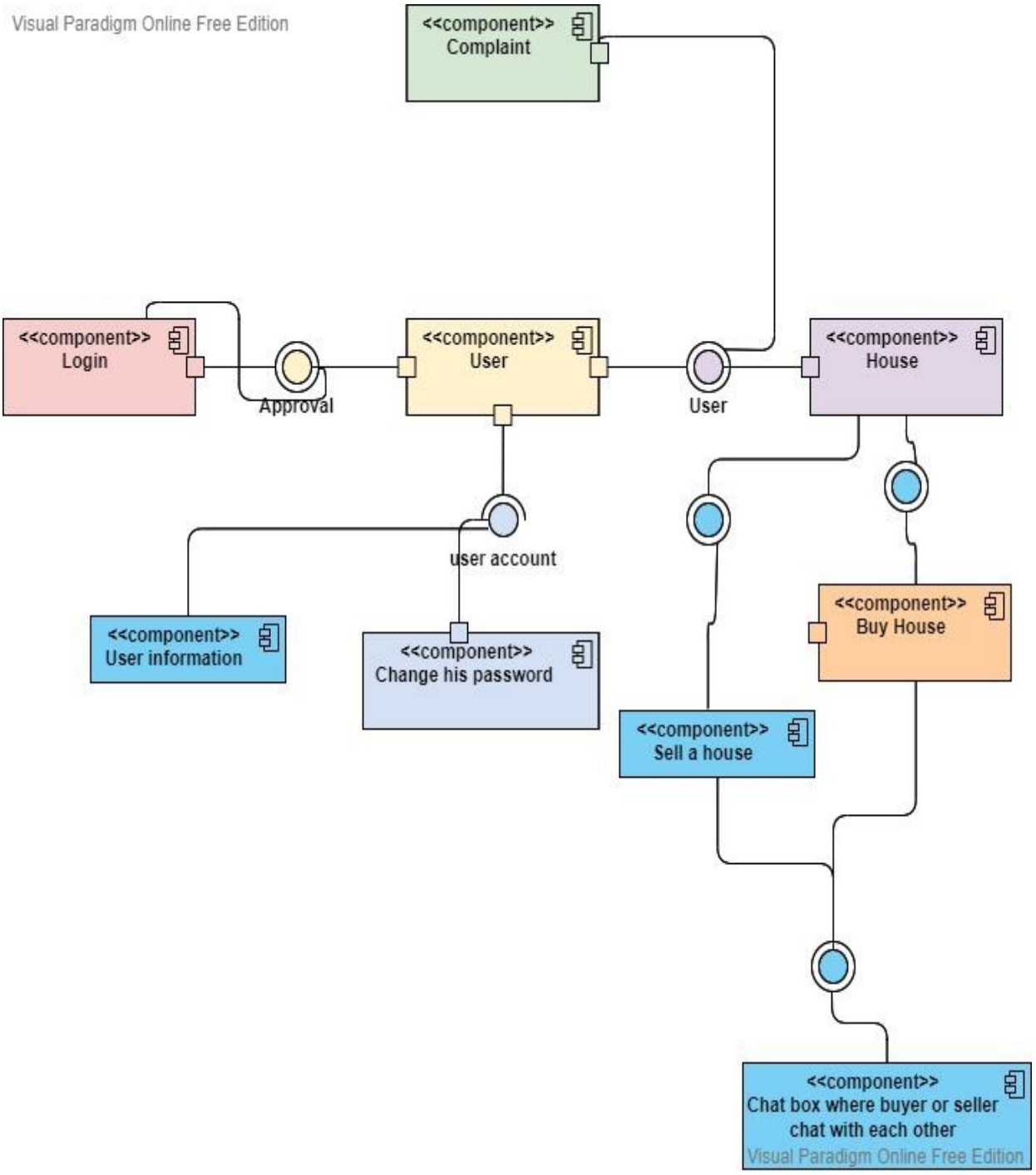
Post conditions: View detail successfully.

4.7. Activity Diagram



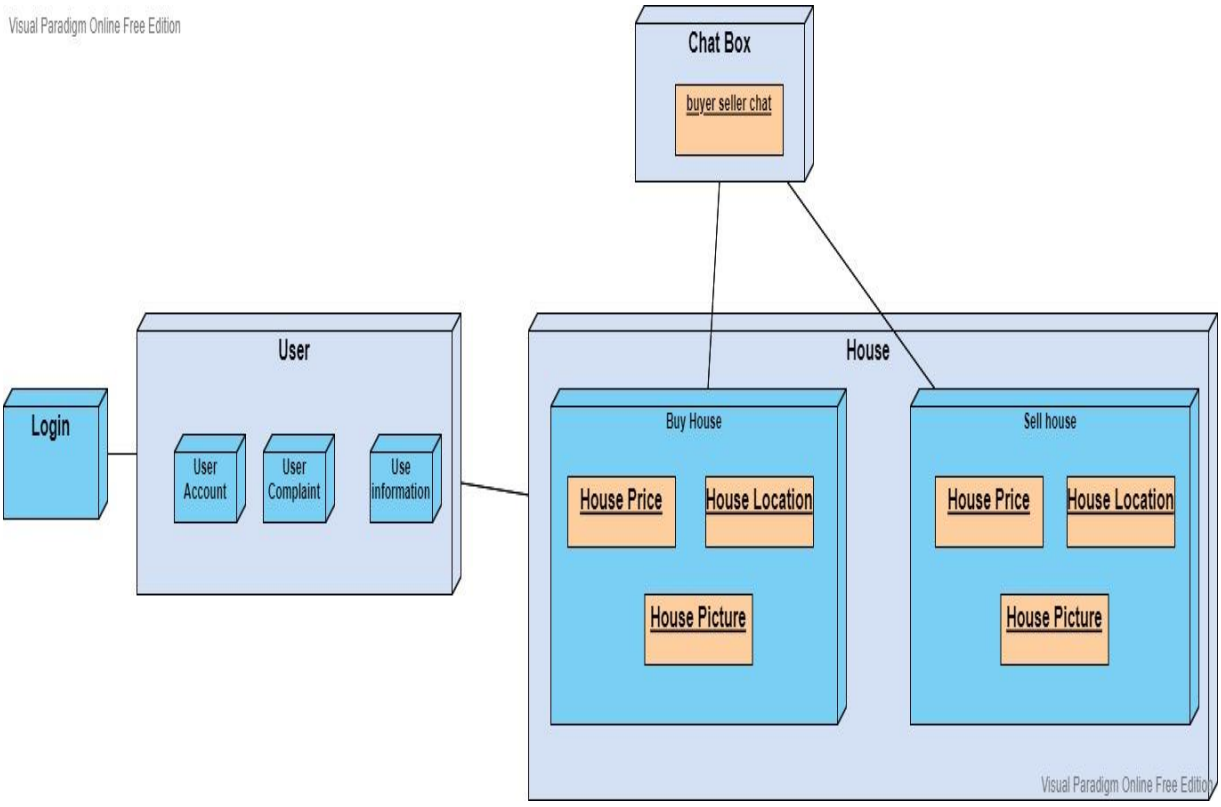
Similar to a flowchart or data flow diagram, an activity diagram visually displays a series of actions or the flow of control in a system. A common tool in business process modeling is the activity diagram. Additionally, they can outline the procedures in a use case diagram. The workflows of stepwise activities and actions are graphically represented in activity diagrams, which also support choice, iteration, and concurrency. Business and software processes are displayed as a series of actions in an activity diagram. People, computer components, or software can all perform these tasks. Activity diagrams are used to explain system processes. These processes are working line wise according to his sequence. First login page after that user account page after house page from where user got buy or sell house page. After that the page will open and start selling and buying.

4.8. Component Diagram



A component diagram's goal is to depict the relationship between various system components; a component diagram's objective is to depict the relationship between various system components. It describes a group of classes that stand in for separate systems or subsystems that can communicate with other parts of the system. In this diagram we show our component which is relate with their component. Like our first component is login after that there us database which check the login is correct or not, if the login is correct then user goes on user page and if the login is wrong then it don't goes on user page. After this our next component is user. User accounts also have two components one is user password if he wants to change or not. The second is user information. Here he adds his information like his name, address and every relevant information of user. The user has a component of complain. Here he write complain. Then user goes on house page here house also two components the one is buying house and the second is sell house. The representation of a component is a rectangle with optional compartments. A component can be displayed as just a rectangle bearing its name.

4.9. Deployment Diagram



Deployment diagrams represent a system's physical architecture. Diagrams of deployment illustrate the connections between system parts and the physical distribution of processing. The actual installation of artifacts on nodes. The topology of the physical parts of a system, where the software components are deployed, is depicted using deployment diagrams. The static deployment view of a system is described using deployment diagrams. In this diagram we have a node of login which is connected to user. After that user has three nodes the first is user account, second is user complaint, and third is user information. The user is connected with the house. After this we have house. In house there are two nodes first is bought house and second node is selling. These two nodes are connected with Chat box. In Buy house there are also three nodes. The first one is house price, second is house location, and the third is house price. Same sell house also has three nodes. The first one is house price, second is house location, and the third is house price.

Chapter 5

Implementation

Chapter 5: Implementation

In this chapter we implement our diagram our system. In this we execute our system, our model and our idea. The process of making formal plans often highly intricate, widely influential plans into reality.

5.1. Important Flow Control/Pseudo codes

Flow Control:

1. Start Application.
2. Introduction Screen.
3. Optional Screen (Log in or Sign Up)
4. If Sign Up.
    - **4.1.** User enters all the required info fields.
    - **4.2.** Get Signed up.
    - **4.3.** Click Sign up.
    - **4.4.** Move to the log in screen.
    - **4.5.** Fills the required fields as user name, email, password.
    - **4.6.** To main menu.
5. If Log in.
6. Fills the required field
7. Next to the main screen.
8. All features of application are on main screen.
    - **8.1.** Register Complaints
    - **8.2.** Show Complaints Status
    - **8.3.** Chat Box in Complaint Status
    - **8.4.** Google Map
9. User can choose any option according to his/her requirement.
10. END!

5.2. Components, Libraries, Web Services and stubs

The component of which we used in project in flutter is:

- App structure and navigation.
- Buttons.
- Input and selections.
- Dialogs, alerts, and panels.
- Information displays.
- Layout.

The libraries of flutter are:

- URL Launcher. Every page can have a plug-in added using URL Launcher. ...
- FL Chart
- Get It
- Path Provider
- Intro Slider
- Local Auth
- Dart

Web Service:

- We develop professional android, iOS and web app with flutter. ...
- We design our flutter app for designing app and for ease. ...
- We create a creative platform android app using flutter. ...
- We develop a app with angular, ionic, flutter.

5.3. Deployment Environment

Deployment Environment needs:

- Mobile Application.
- Server.
- Database.

5.4. Tools and Techniques

We use different tool like in our project like:

- Android Studio
- Flutter
- Firebase
- Dart language

These are basic tool which we use to make this project of E-society

5.5. Best Practices / Coding Standards

The best practices for designing and developing with Flutter to improve code quality, readability, maintainability, and productivity.

- Refractor code into widgets rather than methods.
- Make build function pure.
- Use state management.
- Have a well-defined Architecture.
- Follow effective dart style guide.

5.6. Version Control

The management of modifications to computer programmers, papers, extensive web sites, or other collections of information is handled by a class of systems known as version control. A part of software configuration management is version control.The process of maintaining and tracking code is known as version control, commonly referred to as source control.

Chapter 6

Testing and Evaluation

Chapter 6: Testing and Evaluation

Use Case Testing is a functional black box testing technique that helps testers to identify test scenarios that exercise the whole system on each transaction basis from start to finish.

6.1. Use Case Testing

4. 6.1.1.

Test Suite ID	TS001
Test Case ID	TC001
Test Case Summary	To verify that by clicking Signup button username, password, email, phone# and address store in users' detail in database.
Related Requirement	RSoo1: User should able to Sign up.
Prerequisites	No
Test Procedure	1. Select fields in Signup form. 2. Enter user data in fields. 3. Click Signup button.
Test Data	Valid username: haseeb, HASEEB Invalid username: 12 haseeb ./haseeb ,haseeb %^3 Valid password: 123Abc@5, Abc567$%9 Invalid password: 1_2hasseb, _haseeb12 Valid email:haseebali@gmail.com, 12ali@yahoo.com Invalid email: abdullah.com, shery@yahoo Valid phone# :03123456578 Invalid phone: @56rfgf7999, 2wstyyA
Expected Result	1. If username, password, email and phone# are valid then store user data by clicking signup button. 2. If given inputs are invalid then display error message. 3. If fields are empty then show warning message.
Actual Result	1. If name is valid, the result is as expected. 2. If name is not valid then invalid message displayed. 3. If fields are empty then warning massage displayed.
Status	Pass

Remarks	This test case is simple and easy.
Created By	Sheraz Hussain
Date of Creation	01/12/2022
Executed By	Sheraz Hussain
Date of Execution	01/12/2022
Text Environment	OS: Android Studio Version 3.4

5. *Table 7 : Use case of Sign up*

6. 6.1.2:

Test Suite ID	TS001
Test Case ID	TC002
Test Case Summary	To verify Authentication or Login.
Related Requirement	RS002: User should able to Login.
Prerequisites	User should Signup first.
Test Procedure	1. Select Username field and enter username. 2. Select Password field and enter password. 3. Click Login button.
Test Data	Valid username: Haseeb,HASEEB Invalid username: 1haseeb, ./haseeb , haseeb%^3 Valid password: 123Aabc@, abMc567$% Invalid password: 1_2hasseb, _haseeb12
Expected Result	1. If username and password are valid then clicking the Login button user successfully login. 2. If username and password are invalid then clicking the Login button invalid message display. 3. If fields are empty then display warning message
Actual Result	1. If fields are valid, the result is as expected. 2. If fields are not valid then invalid message displayed. 3. If fields are empty then warning message displayed.

Status	Pass
Remarks	This test case is simple and easy.
Created By	Sheraz Hussain
Date of Creation	01/12/2022
Executed By	Sheraz Hussain
Date of Execution	01/12/2022
Text Environment	OS: Android Studio Version 3.4

7. *Table 8 : use case of login*

8. 6.1.3:

Test Suite ID	TS001
Test Case ID	TC003
Test Case Summary	To verify that society user, admin and state builder once signup if they are forgot password then they can update it by email.
Related Requirement	RS003: user should able to update data.
Prerequisites	user should login.
Test Procedure	1. Select fields. 2. Enter data in fields. 3. Click forgot password button.
Test Data	01/12/2022
Expected Result	1. If enter data is valid then update data by clicking update button 2. If given inputs are invalid then display error message. 3. If fields are empty then display warning message.
Actual Result	1. If fields are valid, the result is as expected. 2. If fields are not valid then invalid message displayed. 3. If fields are empty then warning message displayed.

Status	Pass
Remarks	This test case is simple and easy.
Created By	Sheraz Hussain
Date of Creation	01/12/2022
Executed By	Sheraz Hussain
Date of Execution	01/12/2022
Text Environment	OS: Android Studio Version 3.4

9. *Table 9 : use case of update password*

10. 6.1.4:

11.

Test Suite ID	TS002
Test Case ID	TC004
Test Case Summary	To verify that society user, admin and state builder once signup if they are deleted in database by performing crud operation by clicking Delete button.
Related Requirement	RS004: Admin can delete data.
Prerequisites	Admin should login.
Test Procedure	Click delete button.
Expected Result	If data is deleted by click button then message will appear
Actual Result	If action is valid, the result is as expected.
Status	Pass
Remarks	This test case is simple and easy.
Created By	Sheraz Hussain
Date of Creation	01/12/2022

Executed By	Sheraz Hussain
Date of Execution	01/12/2022
Text Environment	OS: Android Studio Version 3.4

12. *Table 10 : use case of delete data*

13. 6.1.5:

Test Suite ID	TS002
Test Case ID	TC005
Test Case Summary	To verify that data is searched by clicking search button.
Related Requirement	RS005: User should be able to search data.
Prerequisites	User should login.
Test Procedure	1. Select search field. 2. Enter name in fields. 3. Click search button.
Expected Result	1. If name is valid then show detail. 2. If name is invalid then display error message. 3. If field is empty then display warning message.
Actual Result	1. If name is valid, the result is as expected. 2. If name is not valid then invalid message displayed. 3. If fields are empty then warning message displayed.
Status	Pass
Remarks	This test case is simple and easy.
Created By	Sheraz Hussain
Date of Creation	01/12/2022
Executed By	Sheraz Hussain
Date of Execution	01/12/2022

Text Environment	OS: Android Studio Version 3.4

14. *Table 11: use case of search data*

15. 6.1.6:

Test Suite ID	TS002
Test Case ID	TC006
Test Case Summary	To verify that record is display.
Related Requirement	RS006: User should be able to view data.
Prerequisites	User should login.
Test Procedure	1. Enter name in fields. 2. Click View button.
Expected Result	1. If name is valid then show detail. 2. If name is invalid then display error message. 3. If field is empty then display warning message.
Actual Result	1. If name is valid, the result is as expected. 2. If name is not valid then invalid message displayed. 3. If fields are empty then warning message displayed.
Status	Pass
Remarks	This test case is simple and easy.
Created By	Sheraz Hussain
Date of Creation	01/12/2022
Executed By	Sheraz Hussain
Date of Execution	01/12/2022
Text Environment	OS: Android Studio Version 3.4

16. *Table 12 : use case of record to display*

6.1. Equivalence partitioning

1	User name is alphabetic.	valid
2	User name is not alphabetic.	invalid
3	Password is equal to 10 or greater than 15 characters in length.	valid
4	Password is 2 to 16 characters in length.	invalid
5	Password include one uppercase letter & one lowercase letter.	valid
6	Password include '_'.	invalid
7	Email without '@' and '.com'.	invalid
8	Email with '@' and '.com'.	valid
9	Price only be numeric.	valid
10	Price is alphabetic.	invalid
11	Search product with alphabetic.	valid
12	Search product with special character (/,%,^,#,@).	invalid
13	No search product name entered	invalid
14	Brand name alphabetic	valid
15	Brand name with special character	invalid
16	Empty field	invalid

6.2. Boundary value analysis

Sr.		Partition 1	Partition 2	Partition 3
1.	Password	Less than 8 character	1 – 8 character	9 – 12
2.	Phone	<=0	1 - 11	9 – 12

6.3. Data flow testing

The relationship between one entity and another while performing a specific task in during data flow. Such as between the sign up and registration process etc.

6.4. Unit testing

In unit testing we have testified our different panel codes individually by performing different tests and by executing them individually, separately on different computers and they were successfully executed and they performed well.

6.5. Integration testing

In performance testing a particular certain situation is given to the website let's just say 100 users try to register at the same time so how the system recovers back gracefully with complete results. So according to our extracted results, around 100+ people were able to register in our website.

6.6. Performance testing

In performance testing a particular certain situation is given to the website let's just say 100 users try to register at the same time so how the system recovers back gracefully with complete results. So according to our extracted results, around 100+ people were able to register in our website

6.7. Stress Testing

Let's say if 100 customers which increases the stress on the website at a maximum load how will the website will perform under these circumstances.

Activity	Description
Register	The user can register and save his/her information into the system
Login	The user can login using his/her credentials
Type of user	Upon registering, the user specifies which type of user they are

Manage profiles	The user can edit/disable their accounts
Browse Categories	The user can browse the categories of lop they want to buy or sale the house.
Search Profiles	The user can search profiles based on the base of their interest
Call user	The user can call the phone number of the user
Add Reviews	The user can add reviews
Check Notification	The user can check notifications for possible according to their interest
Log out	The user can log out

Chapter 7

Summary, Conclusion and Future Enhancements

Chapter 7: Summary, Conclusion & Future Enhancements

7.1. Project Summary

Now that the project E-Society is finished, I can describe the app's function. The project is essentially a mobile app. The major purpose of this app is to address the problems of the community and help people to purchase and sell houses using the map. According to the needs of both businesses, our app will produce the UI and related functionalities. Numerous functionalities will be offered by this system on a single platform.

Nevertheless, there is always room for development. Numerous functionalities will be offered by this system on a single platform.

Nevertheless, there is always room for development.

7.2. Achievements and Improvements

Our ability to improve our talents to a professional level and apply them to this project, which we learned in four years of university, is what we consider to be our biggest accomplishments. We acquired knowledge in many areas, including project management, testing, UML modeling, software architecture design approaches, and much more, and we were able to put it all to use on this project. The lessons we learned from this project are the next major accomplishment. new programming languages, frameworks, libraries, and database diagramming software. Everything will be helpful to us in the future. We acquire creative thinking skills.

7.3. Critical Review

The key features of our system are its quick response time, user-friendly interface, low cost, ability to meet users' specific needs in the moment, and availability of all features that a large society might need. It took a lot of effort and time to manage all of these aspects.

7.4. Lessons Learnt

We learn a great deal from this project. This project improves our knowledge of the flutter dart and android studio tools, as well as many other management ideas, problem-solving techniques, and how to persevere until you discover the answer to any problem. This project not only improves our technical skills but also our personal development qualities, such as teamwork and dedication.

7.5. Future Enhancements/Recommendations

As has already been mentioned, there is always potential for development. Additionally, updates will continue to be made since we intend to launch this application as a startup. The system was created using Android Studio's Flutter Dart, but as we improve its effectiveness, usability, and adaptability, we will undoubtedly scale it up to become whatever we anticipate ours will ultimately become. We improve the interface's usability.

Reference and Bibliography

Reference and Bibliography

[1] M. Sher, M. Rehman, "*Title of the Paper*" Conference name/Journal Name, Edition, Volume, Issue, ISBN/ISSN, PP, Publisher/City-Country, Year.

[2] ……

Index

Index

[A]

[B]

[C]

References

References

1. *Automatic Text Summarization Methods: A Comprehensive Review.* **Divakar Yadav.**

2. *NAREN RAMAKRISHNAN. (n.d.). Neural Abstractive Text Summarization with Sequence-to-Sequence Models. https://dl.acm.org/doi/fullHtml/10.1145/3419106.*

3. *Jason Brownlee. (2017, May 24). A Gentle Introduction to Long Short-Term Memory Networks by the Experts. Machine Learning Mestary. https://machinelearningmastery.com/gentle-introduction-long-short-term-memory-networks-experts/.*

4. *QualMiner: Text mining for the Venezuelan Humanitarian Response in Ecuador. (2022, April). https://bedatadriven.github.io/QualMiner/textual-data-preparation.html.*

5. *Jaiprakash Verma, & PatelAtul Patel. (2017, may). Evaluation of Unsupervised Learning based Extractive Text Summarization Technique for Large Scale Review and Feedback Data. https://www.researchgate.net/publication/317610956_Evaluation_of_Unsupervised_Lea.*

6. *STUDFUZZ. (n.d.). Validation Techniques in Text Mining (with Application to the Processing of Open-ended Questions). In Text Mining and its Applications (Vol. 138). https://link.springer.com/chapter/10.1007/978-3-540-45219-5_13.*

7. **Divakar Yadav, Jalpa Desai and Arun Kumar Yadav.** Automatic Text Summarization Methods: A Comprehensive Review. [Online] [Cited: February 13, 2023.] https://arxiv.org/ftp/arxiv/papers/2204/2204.01849.pdf.

8. *Journal of King Saud University - Computer and Information Sciences. (2022, April). ScienceDirect. https://www.sciencedirect.com/science/article/pii/S1319157820303712.*

9. *Divakar Yadav. (n.d.). Automatic Text Summarization M*