

E-State v 1.0

Obaid Ismail (190892)

Muhammad Umar Akhtar (190924)

Aamir Yasin (190928)



Fall-2022

Supervised By

Hafiz Muhammad Ahsan Shehzad

Department of Creative Technologies

Air University, Islamabad

Submission Form for Final-Year

PROJECT REPORT



Version	V 1.0
----------------	-------

NUMBER OF MEMBERS	3
--------------------------	---

TITLE	<i>E-State</i>
--------------	----------------

SUPERVISOR NAME	Hafiz Muhammad Ahsan Shehzad
------------------------	------------------------------

MEMBER NAME	REG. NO.	EMAIL ADDRESS
Obaid Ismail	190892	190892@students.au.edu.pk
Muhammad Umar Akhtar	190924	190924@students.au.edu.pk
Aamir Yasin	190928	190892@students.au.edu.pk

MEMBERS' SIGNATURES

Supervisor's Signature

Note 1: This paper must be signed by your supervisor

Note 2: The soft-copies of your project report, source codes, schematics, and executable should be delivered in a CD

APPROVAL CERTIFICATE

This project, entitled as “E-State ” has been approved for the award of

Bachelors of Science in Software Engineering

Committee Signatures:

Supervisor:

(Hafiz Muhammad Ahsan Shehzad)

Project Coordinator:

(Dr. Imran Ihsan)

Head of Department:

(Dr. Amman Ullah Yasin)

DECLARATION

I/We, hereby, declare that “No portion of the work referred to, in this project has been submitted in support of an application for another degree or qualification of this or any other university/institute or other institution of learning”. It is further declared that this undergraduate project, neither as a whole nor as a part thereof has been copied out from any sources, wherever references have been provided.

MEMBERS' SIGNATURES

ACKNOWLEDGEMENTS

Special Thanks to our Supervisor Sir Hafiz Muhammad Ahsan Shehzad.

Executive Summary

E-State is an e-commerce web application in which users can buy and sell real estate. This platform contains a web tool through which users can experience virtual tour of the real estate so users can view the product that they are about to buy while being in the comfort of their homes. Users can upload their real estates by simply providing related information and images of the product. Utilizing the images provided the tools creates 3d model and virtual tour. Users can also view different real estate and can buy them online, in the process saving time and money by just viewing the product online rather than going to the specific location. By viewing the virtual tour from the comfort of their homes, users would not have to visit the location in real time rather can experience the real thing on a device.

We have developed functional Graphical user interface. At the moment we don't have a functional backend and database. All the artifacts are present in the report.

Table of Contents

This should give a complete list of what the report contains starting with the abstract (the title page is not included in the contents list).

CHAPTER 1

INTRODUCTION

- 1.1. PROJECT INTRODUCTION
- 1.2. EXISTING EXAMPLES / SOLUTIONS
- 1.3. BUSINESS SCOPE
- 1.4. Useful TOOLS AND TECHNOLOGIES
- 1.5. PROJECT WORK BREAK DOWN
- 1.6. PROJECT TIME LINE

CHAPTER 2

REQUIREMENT SPECIFICATION AND ANALYSIS

- 2.1. FUNCTIONAL REQUIREMENTS
- 2.2. NON-FUNCTIONAL REQUIREMENTS
- 2.3. SELECTED FUNCTIONAL REQUIREMENTS
- 2.4. SYSTEM USE CASE MODELING
- 2.5. SYSTEM SEQUENCE DIAGRAM
- 2.6. DOMAIN MODEL

CHAPTER 3

SYSTEM DESIGN

- 3.1. LAYER DEFINITION
 - 3.1.1. Presentation LAYER
 - 3.1.2. Business LOGIC LAYER
 - 3.1.3. DATABASE LAYER
- 3.2. SOFTWARE ARCHITECTURE
- 3.3. CLASS DIAGRAM
- 3.4. SEQUENCE DIAGRAM
- 3.5. ENTITY RELATIONSHIP DIAGRAM
- 3.6. DATABASE SCHEMA
- 3.7. USER INTERFACE DESIGN

CHAPTER 4

SOFTWARE DEVELOPMENT

4.1. CODING STANDARDS

4.1.1. INDENTATION

4.1.2. DECLARATION:

4.1.3. STATEMENT STANDARDS

4.1.4. NAMING CONVENTIONS:

4.2. DEVELOPMENT ENVIRONMENT

4.3. DATABASE MANAGEMENT SYSTEM

4.4. SOFTWARE DESCRIPTION

List of Figures

Figure1. 1: Work breakdown Structure	Page:15
Figure 1.2: Gant Chart	Page:16
Figure 2.1: Use case Diagram	Page:31
Figure 2.2: System Sequence Diagram	page:47
Figure 2.2:1 Sign Up	Page:48
Figure 2.2:2 Sign In	Page:49
Figure 2.2:3 Switch Account	Page:50
Figure 2.2:4 List property	Page:51
Figure 2.2:5 Convert Unit	Page:52
Figure 2.2:6 Generate Walktrough	Page:53
Figure 2.2:7 Remove Property	Page:54
Figure 2.2:8 Update Property	Page:55
Figure 2.2:9 Search Property	Page:56
Figure 2.2:10 View property	Page:57
Figure 2.2:11 View 3D Walkthrough	Page:58
Figure 2.2:12 Buy Property	Page:59
Figure 2.2:13 Add Property to Favorites	Page:60
Figure 2.2:14 Payment Upon purchase	Page:61
Figure 2.2:15 Transfer of Ownership	Page:62
Figure 2.2:16 Return on Investment Calculator	Page:63
Figure 2.3: Domain Model	Page:64
Figure 3.1: Software Architecture Diagram	Page:67
Figure 3.2: UML Class Diagram	Page:69
Figure 3.3: Sequence Diagram	Page:70
Figure 3.3:1 Sequence Diagram	Page:71
Figure 3.3:2 Sequence Diagram	Page:72
Figure 3.3:3 Sequence Diagram	Page:73
Figure 3.3:4 Sequence Diagram	Page:74
Figure 3.4: Entity Relationship Diagram	Page:75
Figure 3.5: Database Schema	Page:76
Figure 3.6: GUI elements	Page:77

LIST OF TABLES

Table 2.1: Functional Requirements	Page:18
Table 2.1:1 Sign Up	Page:19
Table 2.1:2 Sign In	Page:19
Table 2.1:3 Switch Account Type	Page:20
Table 2.1:4 List property	Page:20
Table 2.1:5 Dimension Unit Converter	Page:22
Table 2.1:6 Generate 3D Walkthrough	Page:22
Table 2.1:7 Remove Property	Page:23
Table 2.1:8 Update Property	Page:23
Table 2.1:9 Search Property	Page:24
Table 2.1:10 View property	Page:25
Table 2.1:11 View 3D Walkthrough	Page:25
Table 2.1:12 Buy Property	Page:25
Table 2.1:13 Add Property to Favorites	Page:26
Table 2.1:14 Property Recommendation System	Page:26
Table 2.1:15 Payment Upon purchase	Page:27
Table 2.1:16 Transfer of Ownership	Page:27
Table 2.1:17 Return on Investment Calculator	Page:28
Table 2.2: Non Functional Requirements	Page:29
Table 2.3: Selected Functional Requirements	Page:29
Table 2.4: Fully Dressed Use Cases	Page:32
Table 2.4:1 Sign Up	Page:32
Table 2.4:2 Sign In	Page:33
Table 2.4:3 Switch Account	Page:34
Table 2.4:4 List property	Page:35
Table 2.4:5 Convert Unit	Page:36
Table 2.4:6 Generate Walkthrough	Page:37
Table 2.4:7 Remove Property	Page:37
Table 2.4:8 Update Property	Page:38
Table 2.4:9 Search Property	Page:40
Table 2.4:10 View property	Page:41
Table 2.4:11 View 3D Walkthrough	Page:42
Table 2.4:12 Buy Property	Page:43
Table 2.4:13 Add Property to Favorites	Page:44
Table 2.4:14 Payment Upon purchase	Page:45
Table 2.4:15 Transfer of Ownership	Page:46
Table 2.4:16 Return on Investment Calculator	Page:46

This page is kept blank

Chapter 1

Introduction

E-State is an e-commerce web application in which users can buy and sell real estate. This platform contains a web tool through which users can experience virtual tour of the real estate so users can view the product that they are about to buy while being in the comfort of their homes. Users can upload their real estates by simply providing related information and images of the product. Utilizing the images provided the tools creates 3d model and virtual tour. Users can also view different real estate in VR and can buy them online, in the process saving time and money by just viewing the product online rather than going to the specific location

1.1. Project Introduction

- E-State is an e-commerce web application in which users can buy and sell real estate. This platform contains a web tool through which users can experience virtual tour of the real estate so users can view the product that they are about to buy while being in the comfort of their homes.
- No real estate website offer this facility, we will be the first one to offer this facility of walkthrough from simple plain images and we will launch this project as startup in Rawalpindi/Islamabad.

1.2. Existing Examples / Solutions

1: Zameen.com "Har Pata, Humain Pata Hai"

Their mission was to make Pakistani real estate accessible and convenient for everyone and their vision was to revolutionize the real estate industry, real estate portal that gives buyers, sellers and investors a single platform to interact conveniently.

2: Graana.com the next-generation real estate solutions platform for Pakistan

Their mission was too develop a product that provided safe and transparent property transactions and result-oriented profitable deals and their vision was Shafiq Akbar's dream to transform the real estate sector of Pakistan and elevate it to bring it at par with the best international standards.

Zameen.com and Graana.com both offer to List property on Website with Details, pictures and Location. E-State will offer same feature to list property with details, pictures and Location.

- Zameen.com and Graana.com Both don't have 3d Walkthrough but E-State will offer.*
- Dimensions of property will be mentioned in 3d Walkthrough.*
- You cannot make payment of Property on Website but E-State will offer in much secure way (Blockchain).*
- You have to physically visit and change ownership but on E-State every property is NFT and ownership can be changed in Seconds.*

1.3. Business Scope

- We are creating a platform where a person or client can view a virtual design and can experience virtual tour of the building or a house.
- We will be creating a tool for producing a virtual tour by using some images from the seller and our tool will convert it into the virtual tour. The tool would be imbedded into the website.
- Virtual tours can bring locations to life and provide an immersive experience that offers the feeling of going on a journey through real or conceptual environments.

- Our target audience would be the property dealers, sellers and buyers.
- Users can enlist real estate on our platform and can also buy them with the facility of a experiencing a virtual tour of the real estate

1.4. Useful Tools and Technologies

This section should mention possible technologies that could be used during the designing, development and testing of your project.

You should mention the technologies that you intend to use with a brief but technical justification for your decision. Your discussion should include

1. *What programming language is you using and why?*
2. *Which development environment do you intend to use and why?*
3. *What database (if any) is you using and why?*
4. *Which operating systems will support your software?*
5. *What network protocol (if any) is implemented and why?*

1. *HTML*
2. *CSS*
3. *Java Script*
4. *Bootstrap*
5. *React*
- 5: *MERN stack*
6. *React XR*
7. *Mongo DB Database*

Our website can be operated on any device PC, Mobile, Mac or Linux just browser is needed .

1.5. Project Work Break Down

All projects require planning, including an outline of **who** on the team is doing **what** and **when**; thus, you will need to include a **Work Breakdown** chart. *You must*

identify all the components of the project and also specify how much time you will spend on each component. The justification should include your strengths and weakness from the project point of view and it should indicate that you have allocated appropriate time period for those modules that you find yourself as your weak points. A typical software project is divided into several parts as shown in Figure 1

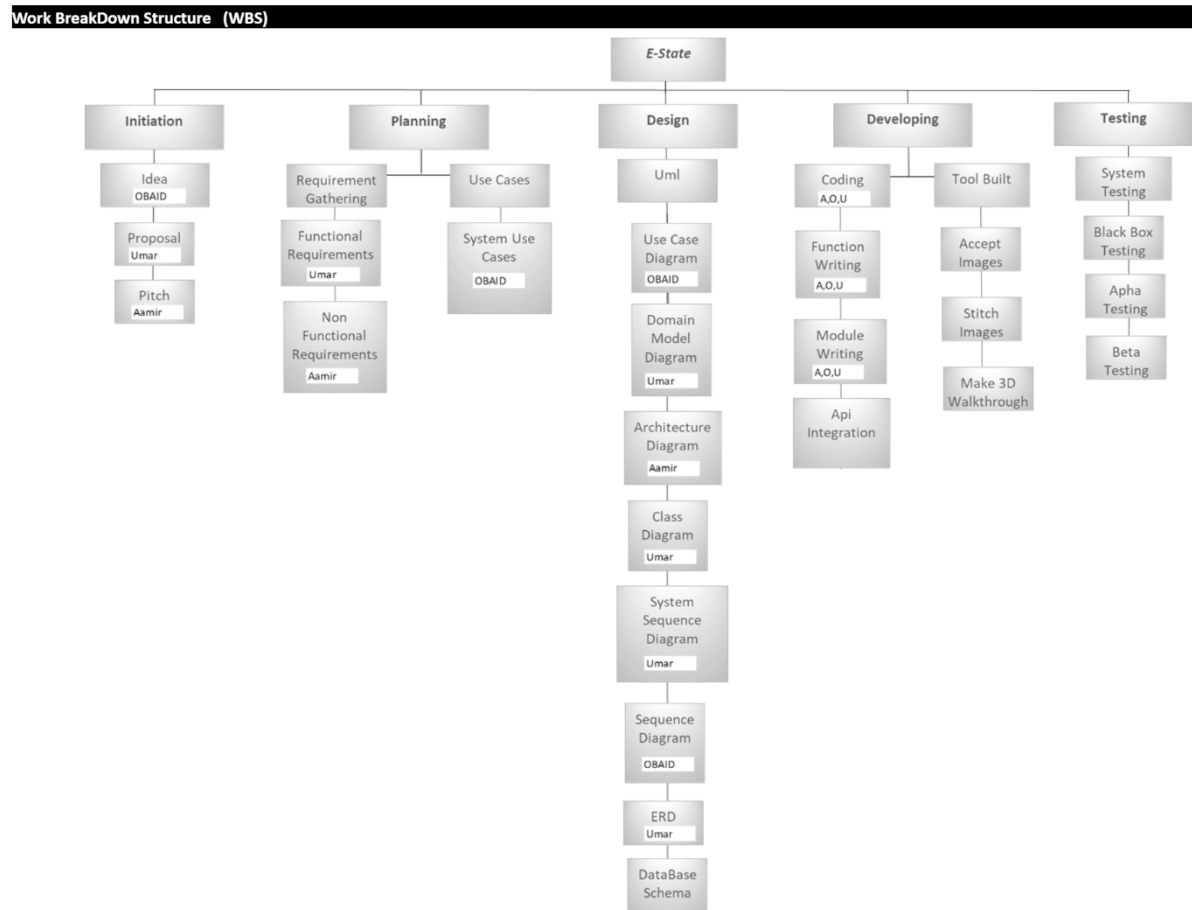
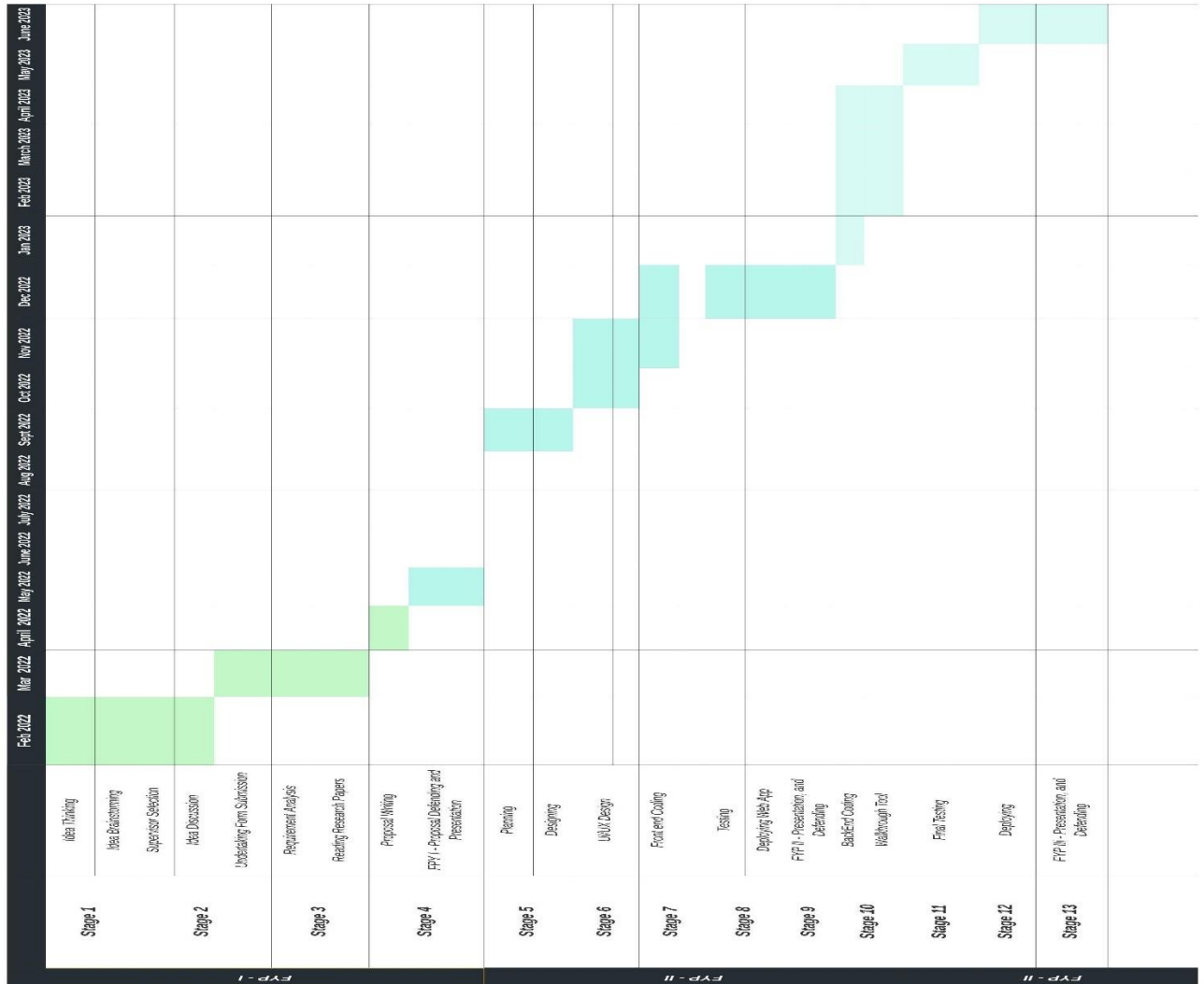


Figure 1.1: WBS

1.6. Project Time Line

A Gantt chart outlines what aspects of the project will be completed and by when. It is an important component of good project management and something you will probably be asked to do as a part of your job.



Chapter 2

Requirement Specification and Analysis

The emphasis of this chapter is on getting an idea of what the requirements are for the intended software. Students who are doing a research related project would provide literature survey for their problems. They are expected to understand the relevant papers and provide summary of the existing work presented in each research paper. Such students should consult their project supervisor for the detailed instructions related to this chapter.

You should write SRS in precise, clear and plain language so that it can be reviewed by a business analyst or customer representative with minimal technical expertise. However it also contains analytical models (use case diagrams, entity relationship diagrams, data dictionary etc.), which can be used for the detailed design and the development of the software system.

Requirement Specification

Requirements specification involves frequent communication with system users to determine specific feature expectations, resolution of conflict or ambiguity in requirements as demanded by the various users or groups of users and documentation of all aspects of the project development process from start to finish. Requirements are a description of how a system should behave or a description of system properties or attributes. It can alternatively be a statement of 'what' an application is expected to do.

2.1. Functional Requirements

The Functional Requirements Specification documents the operations and activities that a system must be able to perform. The Functional Requirements Specification is described in such a way that anyone from non-technical audience can understand. Readers should understand the system, but no particular technical knowledge should be required to understand the document.

- Functional Requirements should include:
- Descriptions of data to be entered into the system
- Descriptions of operations performed by each screen
- Descriptions of work-flows performed by the system
- Descriptions of system reports or other outputs
- Who can enter the data into the system
- How the system meets applicable regulatory requirements

Table 2.1: Functional Requirements

S. No.	Functional Requirement	Type	Status
1	Sign Up	Authorization levels	Under Development
2	Sign in	Authentication	Under Development
3	Switch account type(Buyer/Seller)	Authorization levels	Under Development
4	List property	User Facility	Under Development
5	Dimensions unit convertor calculator	User Facility	Under Development
6	Generate 3D walkthrough	Administrative functions	Under Development
7	Remove property	User Facility	Under Development
8	Update property	User Facility	Under Development
9	Search property	User Facility	Under Development
10	View properties	User Facility	Under Development
11	View 3D walkthrough	User Facility	Under Development
12	Buy property	User Facility	Under Development
13	Add a property to favorites	User Facility	Under Development
14	Property recommendation system	Administrative functions	Under Development
15	Payment upon purchase	Transaction Handling	Under Development
16	Transfer of ownership on purchase	Legal and Regulatory	Under Development
17	Return on investment calculator	Audit Tracking	Under Development

1 Sign Up	
Data	<ul style="list-style-type: none"> • Email • Password • Address • Username • CNIC
Operations	The sign up screen intakes data and responds according to the data entered either signup successful or incorrect data format.
Work-flow	<ul style="list-style-type: none"> • The system processes the input data, then generate and display output according to the nature of the data entered. • If the data entered is according to the format then a new account is created and user is directed to sign in page. • Otherwise the system prompts the user to follow the data format guidelines.
Reports	<ul style="list-style-type: none"> • Signup successful. Please sign into the system to proceed. • Incorrect data format.
Who can enter data	User

2 Sign in	
Data	<ul style="list-style-type: none"> • Email • Password
Operations	The sign in screen intakes data and responds according to the data entered.
Work-flow	<ul style="list-style-type: none"> • The system processes the input data, then generate and display output according to the nature of the data entered. • If the credentials entered are correct, the user is signed into the system and the user is directed to the Home page • Otherwise the system prompts the user that the entered credentials are incorrect.

Reports	<ul style="list-style-type: none"> • Sign in successful. • Incorrect credentials.
Who can enter data	<ul style="list-style-type: none"> • User • Admin

3 Switch account type (Buyer/Seller)	
Data	Nil
Operations	The user clicks the “Switch to buyer/seller” button.
Work-flow	The system switches the interface and the functionalities on the basis of the type switched.
Reports	Interface switches
Who can enter data	Nil

4 List Property	
Data	<ul style="list-style-type: none"> • Property name • Property type • Location • Address • Dimensions • Price • Plain images • 360 degree images
Operations	<ul style="list-style-type: none"> • The user enters the required information.

	<ul style="list-style-type: none"> • According to the data entered the screen displays the output.
Work-flow	<ul style="list-style-type: none"> • The system processes the data. • If the data matches the required format then the property is listed. • Otherwise the system prompts the user to follow the format guidelines
Reports	<ul style="list-style-type: none"> • The property is listed successful. • Incorrect data format. Please match the provided format.
Who can enter data	<ul style="list-style-type: none"> • User • Admin

5 Dimensions unit convertor	
Data	Dimensions of the property
Operations	<ul style="list-style-type: none"> • The user enters the required information. • According to the unit chosen the screen displays the converted dimensions.
Work-flow	<ul style="list-style-type: none"> • The system processes the data. • The system converts the input dimensions unit to the required unit. • Otherwise the system prompts the user to follow the format guidelines.
Reports	<ul style="list-style-type: none"> • Display converted unit. • Incorrect data format. Please match the provided format.
Who can enter data	<ul style="list-style-type: none"> • User • Admin

6 Generate 3D walkthrough	
Data	<ul style="list-style-type: none"> • Plain images • 360 degree images
Operations	<ul style="list-style-type: none"> • The user enters the required information. • The 3d walkthrough is generated. • The 3d walkthrough is displayed.
Work-flow	<ul style="list-style-type: none"> • The system processes the data. • The system uses the images to create a 3D walkthrough and displays it. • Otherwise the system prompts the user to follow the format guidelines.
Reports	<ul style="list-style-type: none"> • Display 3D walkthrough. • Incorrect data format. Please match the provided format.
Who can enter	<ul style="list-style-type: none"> • User

data	<ul style="list-style-type: none"> • Admin
------	---

7 Remove Property	
Data	Nil
Operations	<ul style="list-style-type: none"> • The user selects the property to remove. • The property is removed.
Work-flow	<ul style="list-style-type: none"> • The system checks whether the selected property is in the process of purchase or not. • If the property is not in the process then the system removes it. • Otherwise the system prompts the user “The property is in the purchase process and can’t be removed”.
Reports	<ul style="list-style-type: none"> • Property removed. • The property is in the purchase process and can’t be removed.
Who can enter data	Nil

8 Update property	
Data	User selected field
Operations	<ul style="list-style-type: none"> • The user just overwrites the already filled fields with the updated information. • The user clicks the update information button. • The information is updated.
Work-flow	<ul style="list-style-type: none"> • The system processes the data. • If the data matches the format then the information is

	update. <ul style="list-style-type: none"> • Otherwise the system prompt the user to follow the format guidelines.
Reports	<ul style="list-style-type: none"> • Update successful. • Incorrect data format. Please match the provided format.
Who can enter data	<ul style="list-style-type: none"> • User • Admin

9 Search Property	
Data	Location
Operations	<ul style="list-style-type: none"> • The user enters the location • The property listed with the location provided are displayed
Work-flow	<ul style="list-style-type: none"> • The system processes the data. • If any property matches with the location provided the property is displayed. • Otherwise the system prompt the user that no properties are listed for the location provided.
Reports	<ul style="list-style-type: none"> • Display a list of properties • No properties are listed for the location provided.
Who can enter data	<ul style="list-style-type: none"> • User • Admin

10 View Property	
Data	Nil
Operations	<ul style="list-style-type: none"> • The clicks on the property from the list of properties • The property is displayed
Work-flow	<ul style="list-style-type: none"> • The system displays the property selected by the user.
Reports	<ul style="list-style-type: none"> • Display selected property.
Who can enter data	Nil

11 View 3D walkthrough	
Data	Nil
Operations	<ul style="list-style-type: none"> • The user clicks view 3D walkthrough button • The 3D walkthrough is displayed
Work-flow	<ul style="list-style-type: none"> • The system checks if the 3D walkthrough of the property exists. • If it exists then it is displayed. • Otherwise the system prompt the user that 3D walkthrough of this property does not exist.
Reports	<ul style="list-style-type: none"> • Display 3D walkthrough • 3D walkthrough of this property does not exist.
Who can enter data	Nil

12 Buy Property	
Data	Nil
Operations	<ul style="list-style-type: none"> • The user clicks on the Buy property button • The user is redirected to the payment page
Work-flow	<ul style="list-style-type: none"> • The system checks if the property is already in the purchase phase or not. • If it is in the phase then the system prompts the user that

	<p>the property is already in purchase phase.</p> <ul style="list-style-type: none"> • Otherwise the user is redirected to the payment page
Reports	<ul style="list-style-type: none"> • Property is already in purchase phase .
Who can enter data	Nil

13 Add Property to favorites	
Data	Nil
Operations	<ul style="list-style-type: none"> • The user clicks on the Add to favorites button • The property is added to the favorites
Work-flow	<ul style="list-style-type: none"> • The system checks if the property is already in the favorites • If not then the property is added to the favorites • Otherwise the system prompts the user that the selected property is already in the favorites
Reports	<ul style="list-style-type: none"> • Property is already in favorites • Property is added to the favorites .
Who can enter data	Nil

14 Property recommendation system	
Data	Nil
Operations	<ul style="list-style-type: none"> • The related properties are displayed to the user
Work-flow	<ul style="list-style-type: none"> • The system gathers the user search data and favorites • Based on the data the system recommends user the properties which matches the user interests
Reports	<ul style="list-style-type: none"> • Display properties .
Who can enter data	Nil

15 Payment upon purchase	
Data	<ul style="list-style-type: none"> • Payment method (credit/debit) • Credentials
Operations	<ul style="list-style-type: none"> • The user enters the credentials • A receipt is sent to the user
Work-flow	<ul style="list-style-type: none"> • The system verifies the credentials entered • If the credentials are correct then the system sends a receipt as a record of the purchase to the user • Otherwise the system prompts the user that the entered credentials are incorrect
Reports	<ul style="list-style-type: none"> • Payment successful • Receipt • Incorrect credentials • .
Who can enter data	<ul style="list-style-type: none"> • User • Admin

16 Transfer of ownership on purchase	
Data	Nil
Operations	<ul style="list-style-type: none"> • The ownership is transferred to the user
Work-flow	<ul style="list-style-type: none"> • The system validates the payment • Then transfers the ownership to the user
Reports	<ul style="list-style-type: none"> • Transfer successful • .
Who can enter data	Nil

17 Return on investment calculator	
Data	Nil
Operations	<ul style="list-style-type: none"> • The user clicks on the view ROI • The ROI is calculated and displayed
Work-flow	<ul style="list-style-type: none"> • The system based on the total sales of the property for the user, calculates the ROI • The system displays the ROI
Reports	<ul style="list-style-type: none"> • Display ROI
Who can enter data	Nil

2.2. Non-Functional Requirements

Non-functional requirements cover all the remaining requirements, which are not covered by the functional requirements. They specify criteria that judge the operation of a system, rather than specific behaviors, for example: “Modified data in a database should be updated for all users accessing it within 2 seconds”. Some typical non-functional requirements include performance, scalability, availability, reliability, maintainability, usability and security.

You are supposed to discuss both the functional and non-functional requirement for your project in the Part 1 mid term report. Functional requirements describe what the system should do while non-functional requirements describe how the system works. The Format for presenting these requirements is given in Table 2.

Table 2.2: Functional and Non-Functional Requirement

S. No.	Non-Functional Requirements	Category
1	Scalable databases.	Scalability
2	Blockchain technology's	Security
3	Data accuracy and consistency	Data integrity
4	Customer and user satisfaction	Availability
5	effectiveness and efficiency of maintenance.	Maintainability
6	probability of failure-free operation	Reliability

2.3. Selected Functional Requirements

List of selected functional requirements for current iteration.

Table 2.3: Selected Functional Requirement

S. No.	Functional Requirement	Type
1	Sign Up	Authorization levels
2	Sign in	Authentication
3	Switch account type(Buyer/Seller)	Authorization levels
4	List property	User Facility
5	Unit convertor calculator	User Facility
6	Remove property	User Facility
7	Update property	User Facility
8	Search property	User Facility
9	View properties	User Facility
10	Buy property	User Facility
12	Add a property to favorites	User Facility
12	Payment upon purchase	Transaction Handling
13	Return on investment calculator	Audit Tracking

2.4. System Use Case Modeling

A use case defines a set of use-case instances, where each instance is a sequence of actions a system performs that yields an observable result of value to a particular actor. The functionality of a system is defined by different use cases, each of which

represents a specific goal (to obtain the observable result of value) for a particular actor.

You should develop fully dressed use cases. One way of conceptualize correct use case is by imaging the user interface of all the features of your project. This will help you to improve your project well in time.

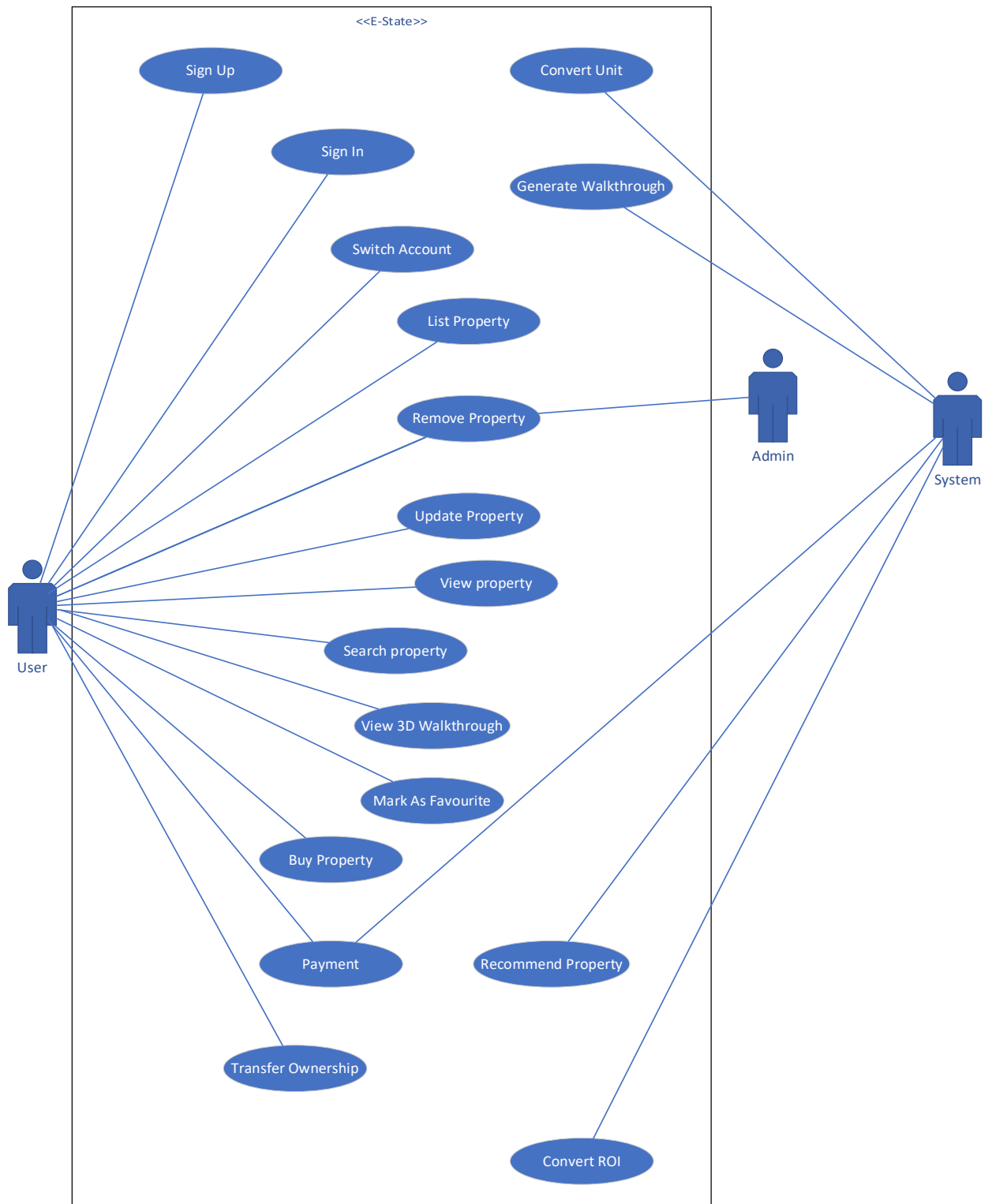


Figure 2.1: Sample Use case Diagram

Use Case1 Title:

Describe the use case (expected behavior of the software) in the form of steps and sub steps in the format given below. You should also provide a brief description of user interface that will satisfy the requirement of each use case

Table 2.1: Use Case 1

Use Case ID:	[01]		
Use Case Name:	Sign Up		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	New User will Register Account on Website.		
Trigger:	User Open website and Click on Sign up.		
Preconditions:	User Must Open Website to click on Sign up.		
Post conditions:	Account will be created and user will be asked to login.		
Normal Flow:	Actor	System	
	User will enter details.	System will check if password is strong and email	

	User Name Email CNIC Password	is not already registered and create account for that user.
Alternative Flows:	If user already registered then user will be told that login with that email. If password is not strong user will be prompted.	
Exceptions:		

Use Case ID:	[02]		
Use Case Name:	Sign In		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	User will enter email and password and click login.		
Trigger:	User Open website and Click on Sign In.		
Preconditions:	User Must Open Website to click on Sign In.		
Post conditions:	Website will be Opened with user Details.		
Normal Flow:	Actor	System	

	User will enter details. Email Password	The system processes the input data, then generate and display output as Website will be opened with user details.
Alternative Flows:	If the credentials entered are incorrect the system prompts the user that the entered credentials are incorrect.	
Exceptions:		

Use Case ID:	[03]		
Use Case Name:	Switch Account		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	The user clicks the “Switch to buyer/seller” button.		
Trigger:	User Open website and Click on Seller/Buyer Button.		
Preconditions:	User Must Open Website to click on Seller/Buyer Button.		
Post conditions:	The system switches the interface and the functionalities on the basis of the type switched.		
Normal Flow:	Actor	System	
	User clicks the “Switch to	The system switches the interface and the	

	buyer/seller" button.	functionalities on the basis of the type switched.
Alternative Flows:		
Exceptions:		

Use Case ID:	[04]		
Use Case Name:	List Property		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	User enter Details to List Property on Website.		
Trigger:	User Open website and Click on List Property.		
Preconditions:	User should have account created on E-State to list property and user must be in seller mode.		
Post conditions:	Property will be successfully listed.		
Normal Flow:	Actor	System	
	The user enters the required information.	The system processes the data, If the data matches the required format, then the property is listed	
Alternative Flows:	If data entered is wrong then system prompts the user to		

	follow the format guidelines
Exceptions:	

Use Case ID:	[05]		
Use Case Name:	Convert unit		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	The user enters the required information. According to the unit chosen the screen displays the converted dimensions.		
Trigger:	User click on convert unit after entering unit.		
Preconditions:	User should have account created on E-State .		
Post conditions:	Display converted unit.		
Normal Flow:	Actor	System	
	The user enters the units in Marla or sq/ft.	The system converts the input dimensions unit to the required unit.	
Alternative Flows:	System prompts the user to follow the format guidelines.		
Exceptions:			

Use Case ID:	[06]		
Use Case Name:	Generate Walkthrough		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	The user enters plain, 360 images. The 3d walkthrough is generated and displayed.		
Trigger:	Click on create Walkthrough.		
Preconditions:	User should have account created on E-State and user should have images.		
Post conditions:	The 3d walkthrough is generated.		
Normal Flow:	Actor	System	
	The user enters plain, 360 images.	The system processes the data. The system uses the images to create a 3D walkthrough and displays it.	
Alternative Flows:	The system prompts the user to follow the format guidelines.		
Exceptions:			

Use Case ID:	[07]
---------------------	------

Use Case Name:	Remove Property		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User, Admin		
Description:	The user selects the property to remove and the property is removed.		
Trigger:	User Open website and Click on Remove Property.		
Preconditions:	User must have listing on website.		
Post conditions:	The property is removed.		
Normal Flow:	Actor	System	
	The user selects the property to remove.	The system checks whether the selected property is in the process of purchase or not. If the property is not in the process, then the system removes it.	
Alternative Flows:	The system prompts the user “The property is in the purchase process and can’t be removed”.		
Exceptions:			

Use Case ID:	[08]
---------------------	------

Use Case Name:	Update property		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	<p>The user just overwrites the already filled fields with the updated information.</p> <p>The user clicks the update information button.</p> <p>The information is updated.</p>		
Trigger:	User Select listing and click on update.		
Preconditions:	User must have listing on website.		
Post conditions:	The property is successfully removed.		
Normal Flow:	Actor	System	
	<p>The user just overwrites the already filled fields with the updated information.</p> <p>The user clicks the update information button.</p>	<p>The system processes the data.</p> <p>If the data matches the format, then the information is update</p>	
Alternative Flows:	The system prompts the user to follow the format guidelines.		
Exceptions:			

Use Case ID:	[09]		
Use Case Name:	Search property		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	The user enters the location The property listed with the location provided are displayed		
Trigger:	User enter location and click on search.		
Preconditions:	User should have account created on E-State.		
Post conditions:	The property listed with the location provided are displayed		
Normal Flow:	Actor	System	
	The user enters the location The property listed with the location provided are displayed	The system processes the data. If any property matches with the location provided the property is displayed.	
Alternative Flows:	The system prompt the user that no properties are listed for the location provided.		
Exceptions:			

Add further use cases in the given format

Use Case ID:	[10]		
Use Case Name:	View property		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	User clicks on the property from the list of properties and the property is displayed		
Trigger:	User Click on property from list.		
Preconditions:	There should be at least 1 property listed on Website.		
Post conditions:	Selected property is displayed		
Normal Flow:	Actor	System	
	User clicks on the property from the list of properties.	The system displays the property selected by the user	
Alternative Flows:			
Exceptions:			

Use Case ID:	[11]		
Use Case Name:	View Walkthrough		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	The user clicks view 3D walkthrough button The 3D walkthrough is displayed		
Trigger:	The user clicks view 3D walkthrough button		
Preconditions:	3D walkthrough of the property must exist.		
Post conditions:	The 3D walkthrough is displayed		
Normal Flow:	Actor	System	
	The user clicks view 3D walkthrough button.	The system checks if the 3D walkthrough of the property exists. If it exists then it is displayed	
Alternative Flows:	The system prompt the user that 3D walkthrough of this property does not exist.		
Exceptions:			

Use Case ID:	[12]		
Use Case Name:	Buy Property		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	The user clicks on the Buy property button The user is redirected to the payment page.		
Trigger:	The user clicks on the Buy property button.		
Preconditions:	User Must Open Website to click on Sign In.		
Post conditions:	Website will be Opened with user Details.		
Normal Flow:	Actor	System	
	The user clicks on the Buy property button.	The system checks if the property is already in the purchase phase or not. The user is redirected to the payment page	
Alternative Flows:	If it is in the phase then the system prompts the user that the property is already in purchase phase.		
Exceptions:			

Use Case ID:	[13]		
Use Case Name:	Mark As Favorite		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	The user clicks on the Add to favorites button and the property is added to the favorites		
Trigger:	The user clicks on the Add to favorites button		
Preconditions:	User Must have account and there should be at least 1 property listed on Website.		
Post conditions:	The property is added to the favorites		
Normal Flow:	Actor	System	
	The user clicks on the Add to favorites button	The system checks if the property is already in the favorites. If not then the property is added to the favorites	
Alternative Flows:	The system prompts the user that the selected property is already in the favorites		
Exceptions:			

Use Case ID:	[14]		
Use Case Name:	Payment		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	System		
Description:	The user enters the credentials to buy property and receipt is sent to the user		
Trigger:	The user Click on Payment.		
Preconditions:	User must have account and select property.		
Post conditions:	Payment will be done and Property will be transferred to user.		
Normal Flow:	Actor	System	
	The user enters the credentials to buy property	The system verifies the credentials entered. If the credentials are correct then the system sends a receipt as a record of the purchase to the user.	
Alternative Flows:	System prompts the user that the entered credentials are incorrect		
Exceptions:			

Use Case ID:	[15]		
Use Case Name:	Transfer Ownership		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	The ownership of property is transferred to the buyer.		
Trigger:	Seller makes buyer the owner.		
Preconditions:	Payment must be verified before changing ownership.		
Post conditions:	Buyer will be new owner.		
Normal Flow:	Actor	System	
	Seller makes buyer the owner	The system validates the payment Then transfers the ownership to the user	
Alternative Flows:			
Exceptions:			

Use Case ID:	[16]
---------------------	------

Use Case Name:	Convert ROI		
Created By:	Obaid Ismail	Last Updated By:	Obaid Ismail
Date Created:	28-10-22	Last Revision Date:	28-10-22
Actors:	User		
Description:	The user can check Return on Investment before buying property.		
Trigger:	User click on View ROI.		
Preconditions:	There must be listing on website.		
Post conditions:	The system displays the ROI.		
Normal Flow:	Actor	System	
	User click on View ROI	The system based on the total sales of the property for the user, calculates the ROI The system displays the ROI	
Alternative Flows:			
Exceptions:			

2.5. System Sequence diagrams

Sequence diagrams are created to show the sequence of events among user and the system to complete an action / use case. A sample is presented in Fig 2.2.

You are required to provide SSD of all the uses cases that you have provided above.

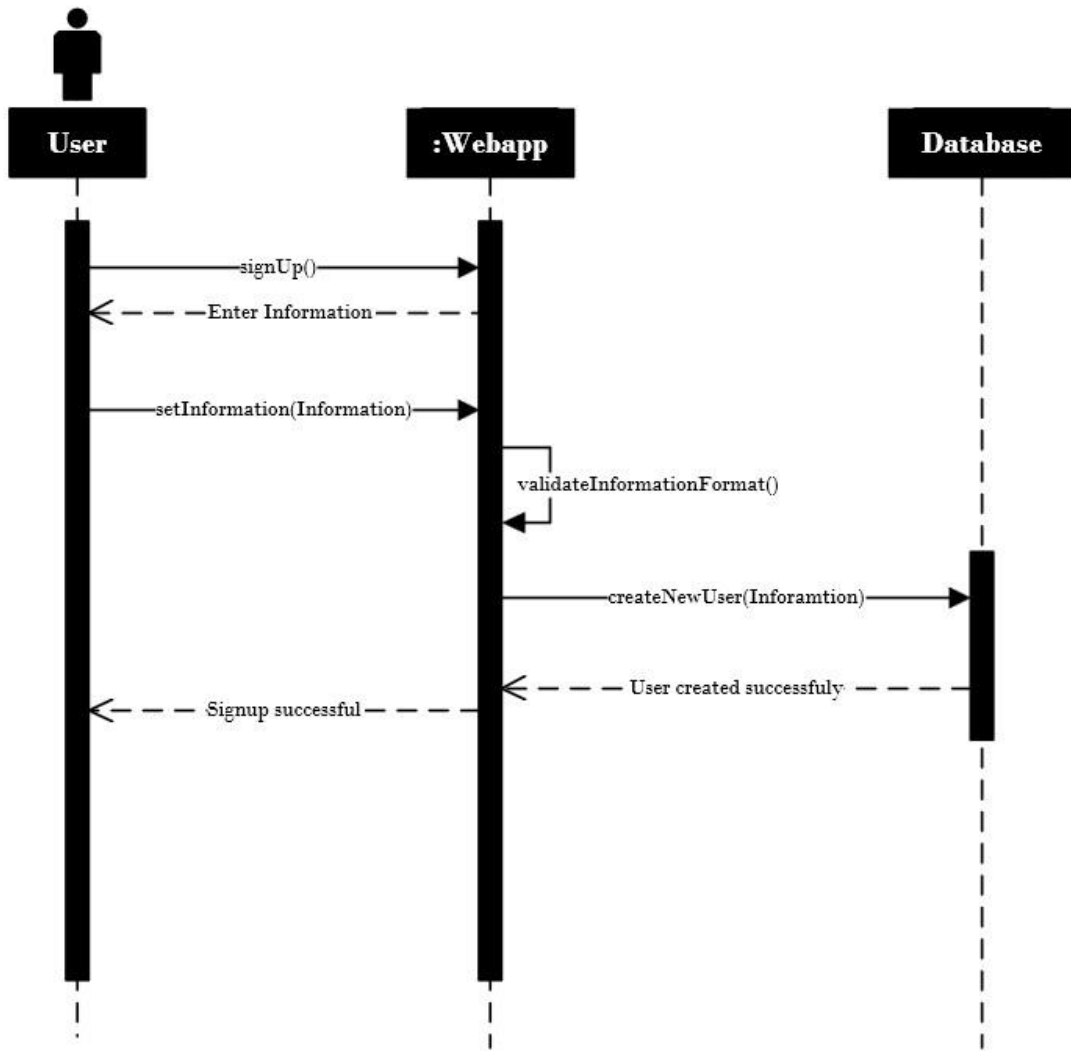


Figure 2.2:1 Sign Up

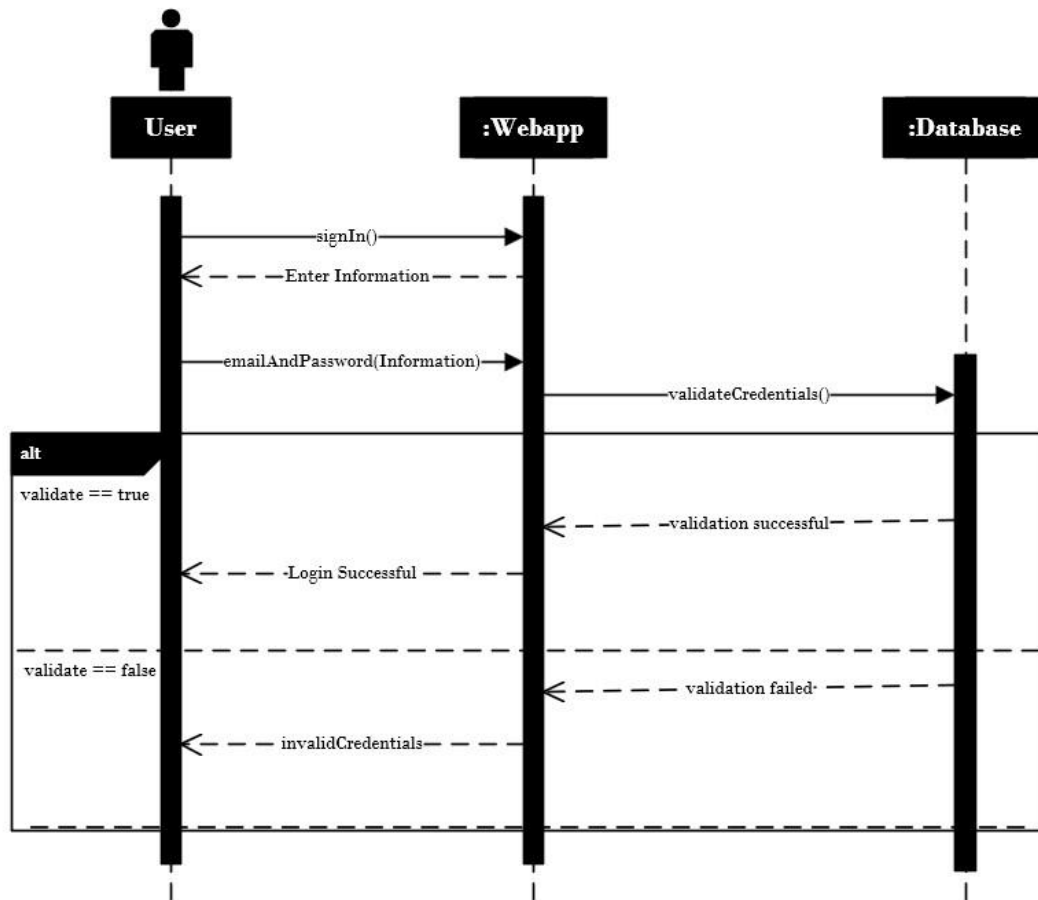


Figure 2.2:2 Sign In

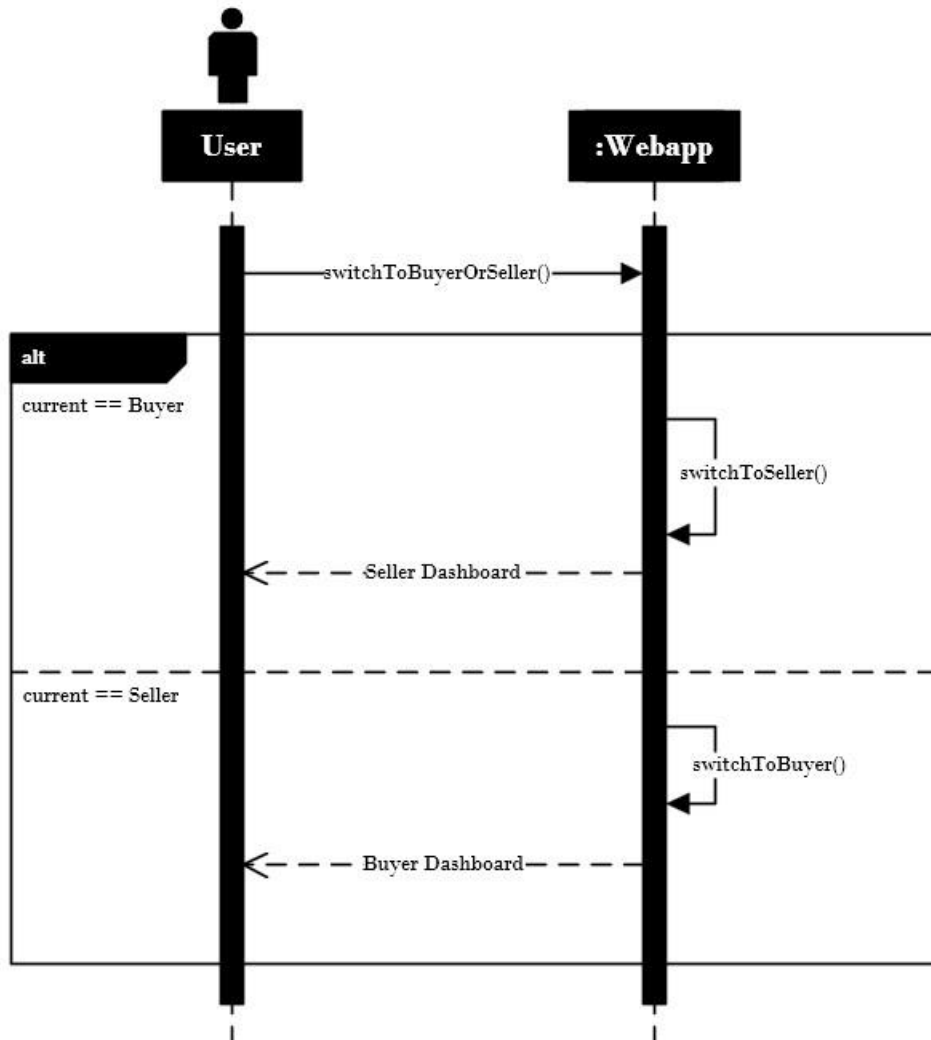


Figure 2.2:3 Switch Account

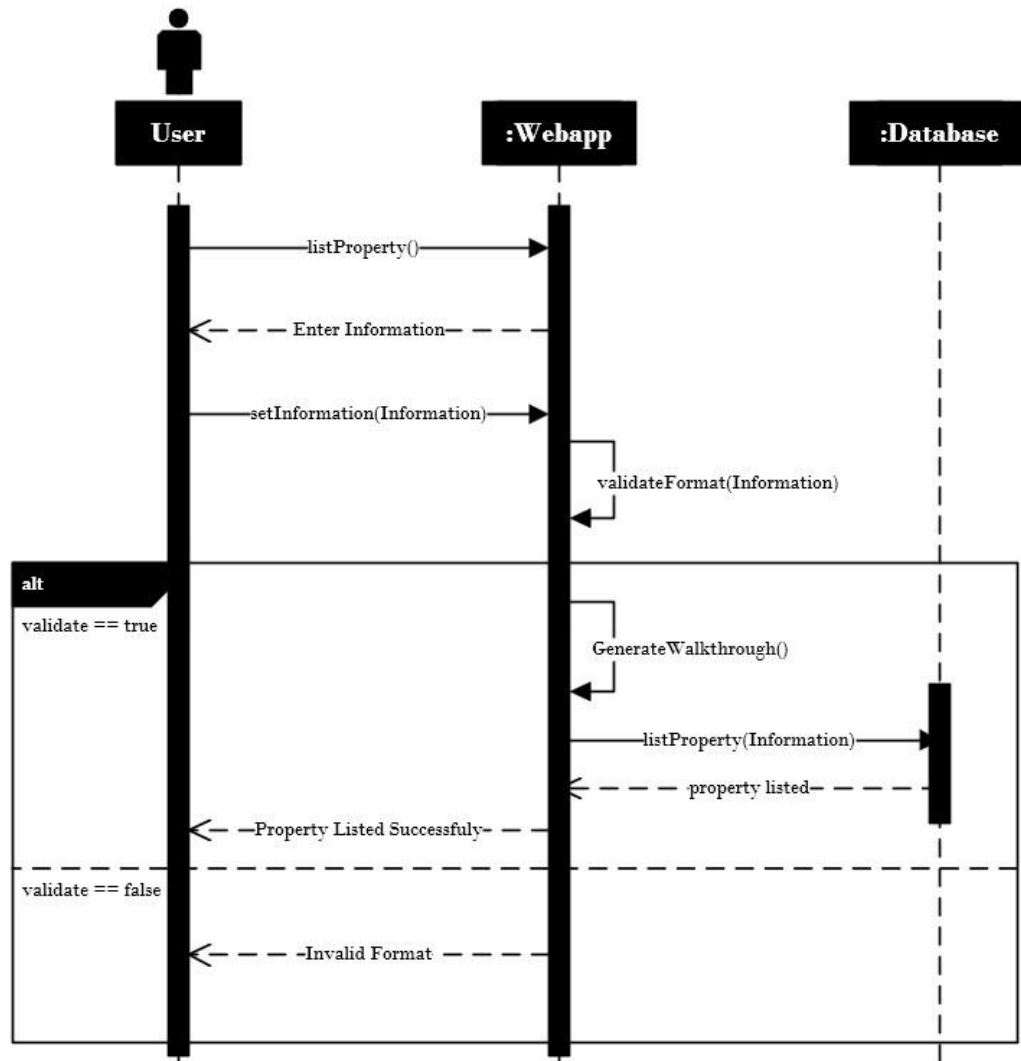


Figure 2.2:4 List Property

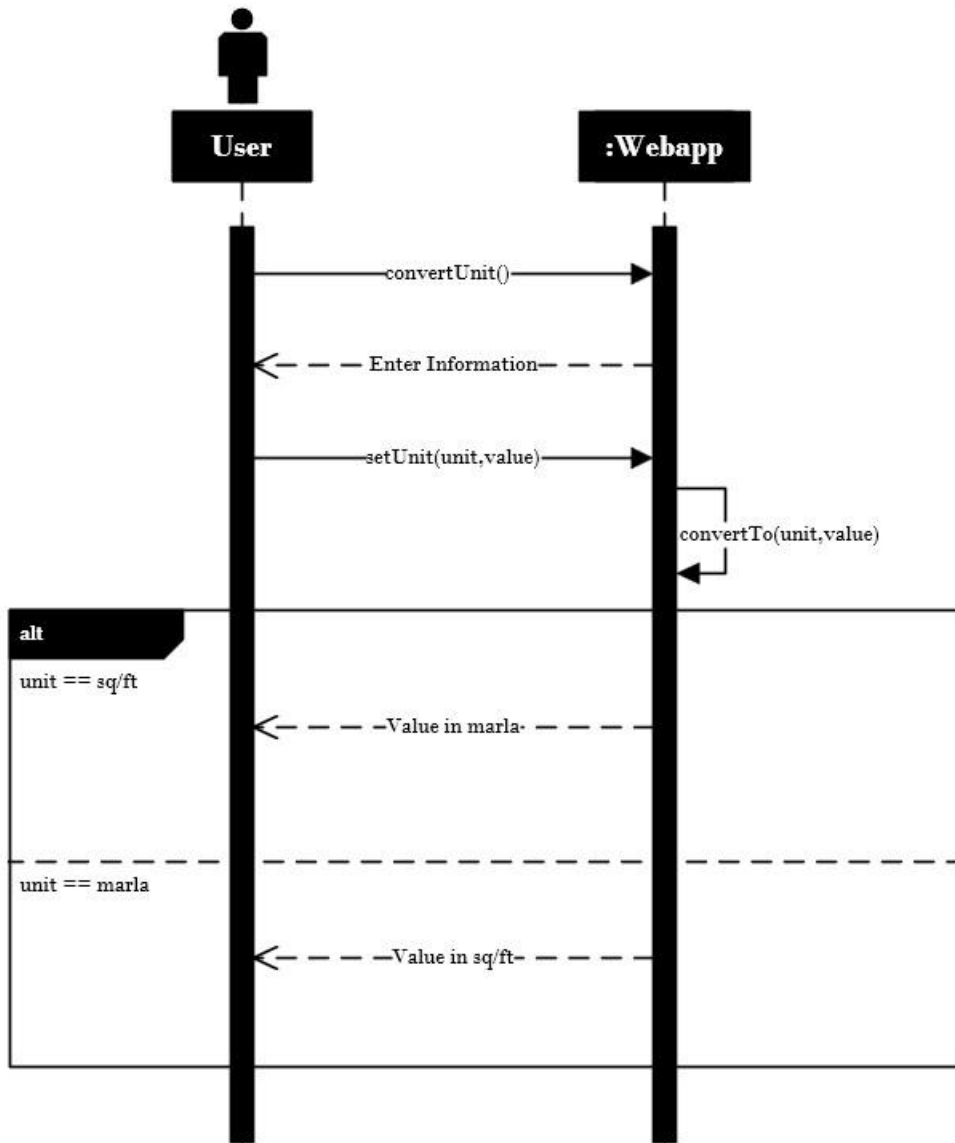


Figure 2.2:5 Convert Unit

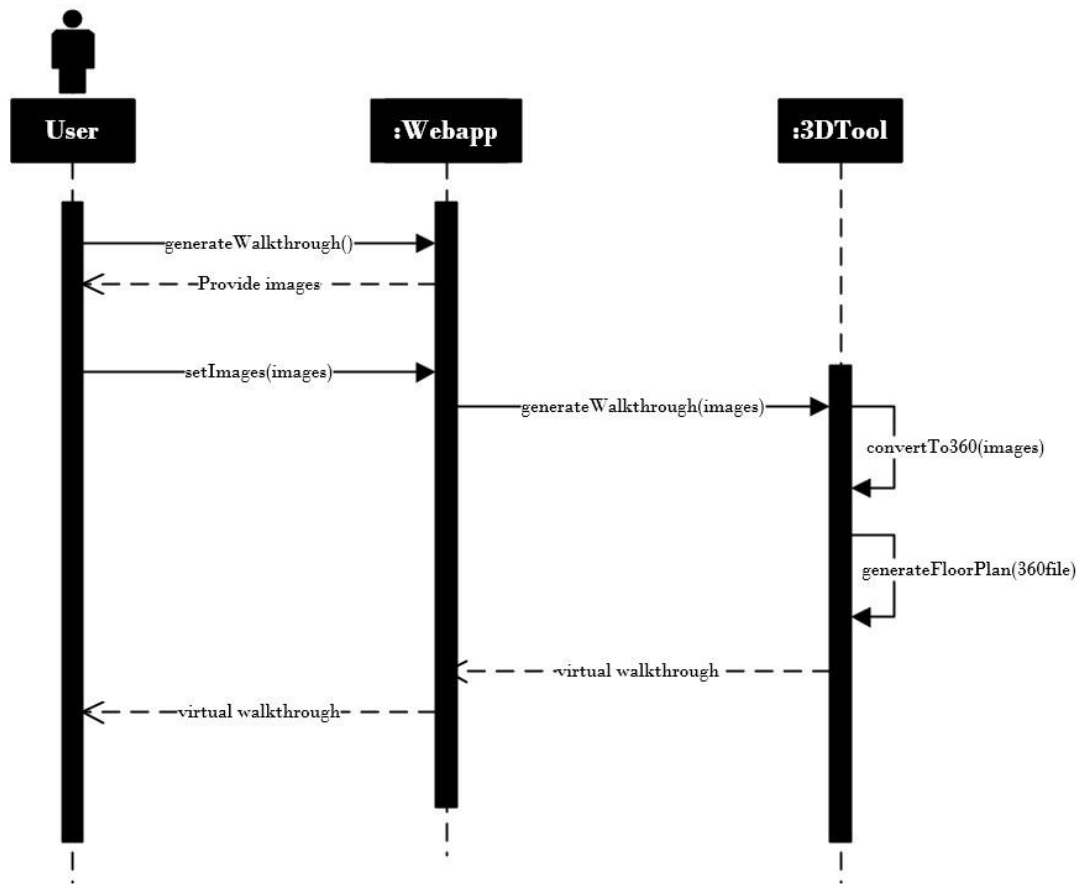


Figure 2.2:6 Generate WalkThrough

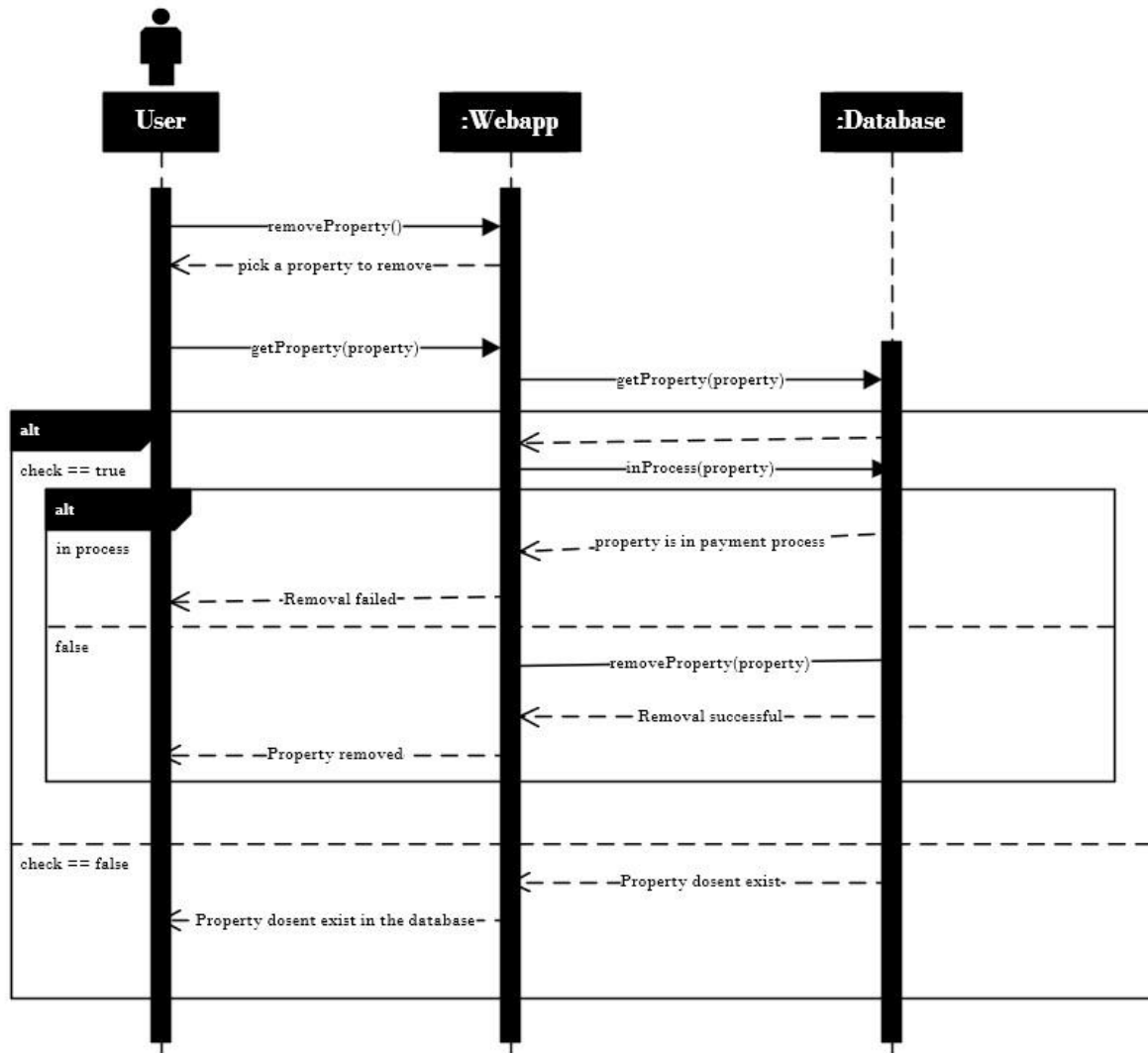


Figure 2.2:7 Remove property

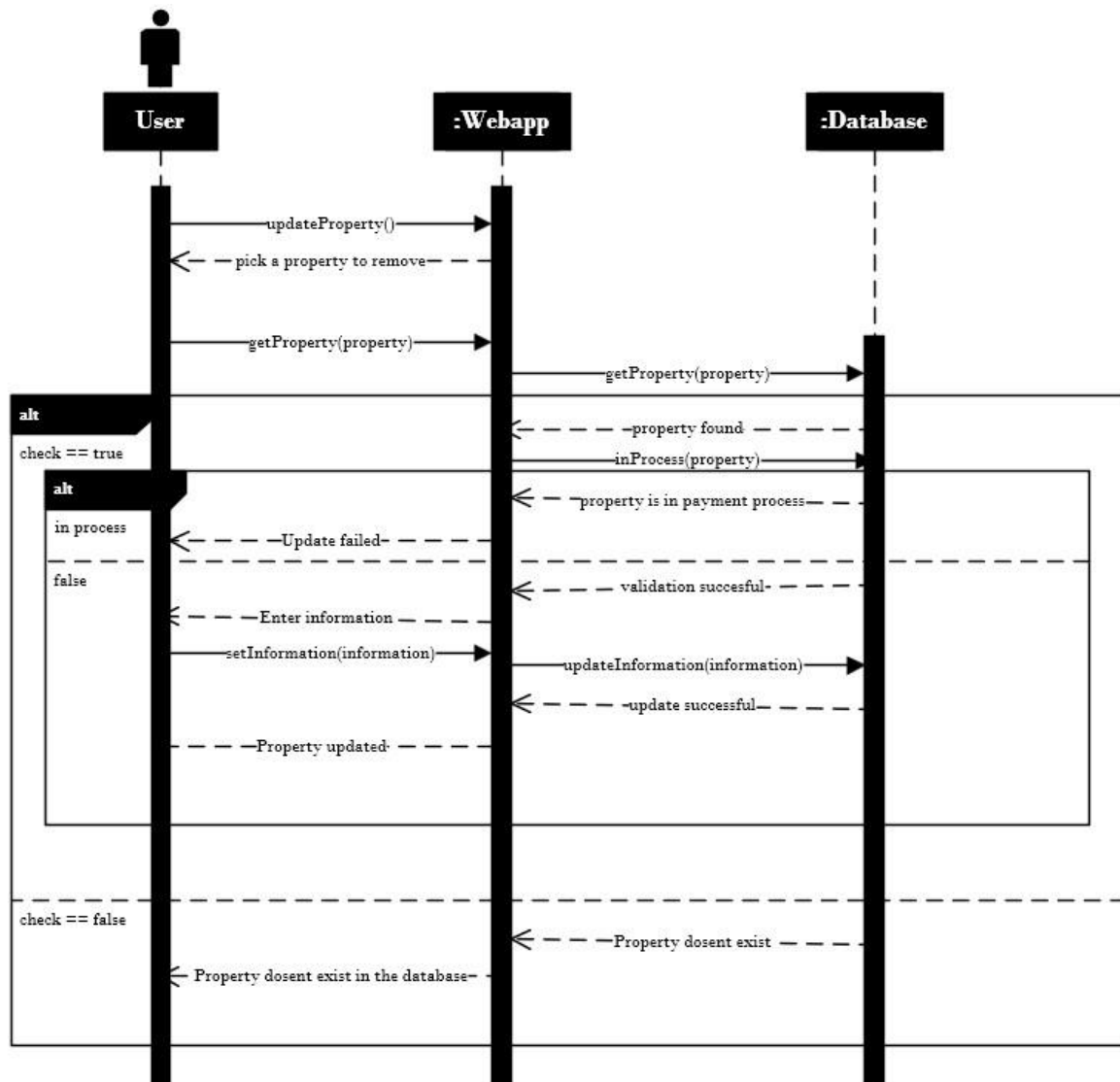


Figure 2.2:8 Update property

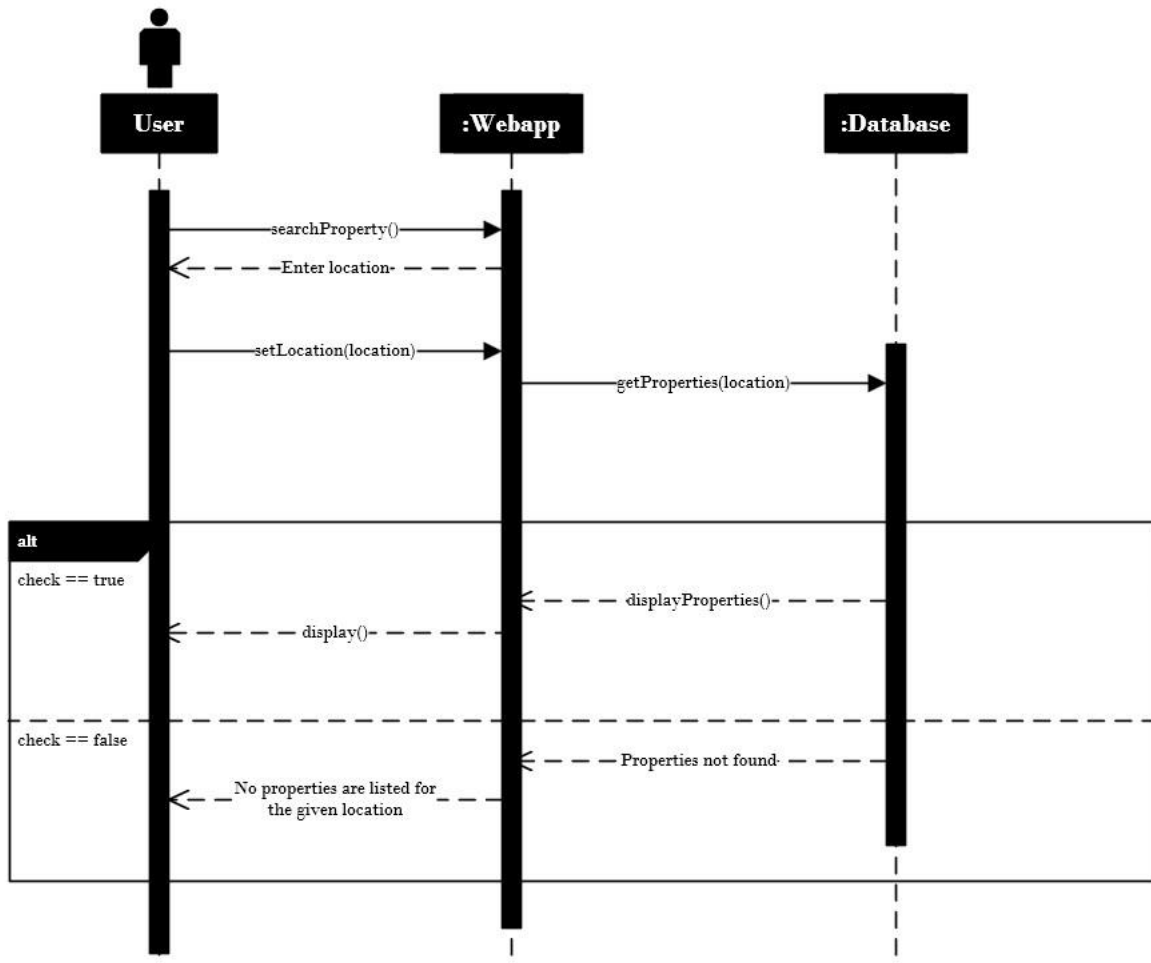


Figure 2.2:9 Search Property

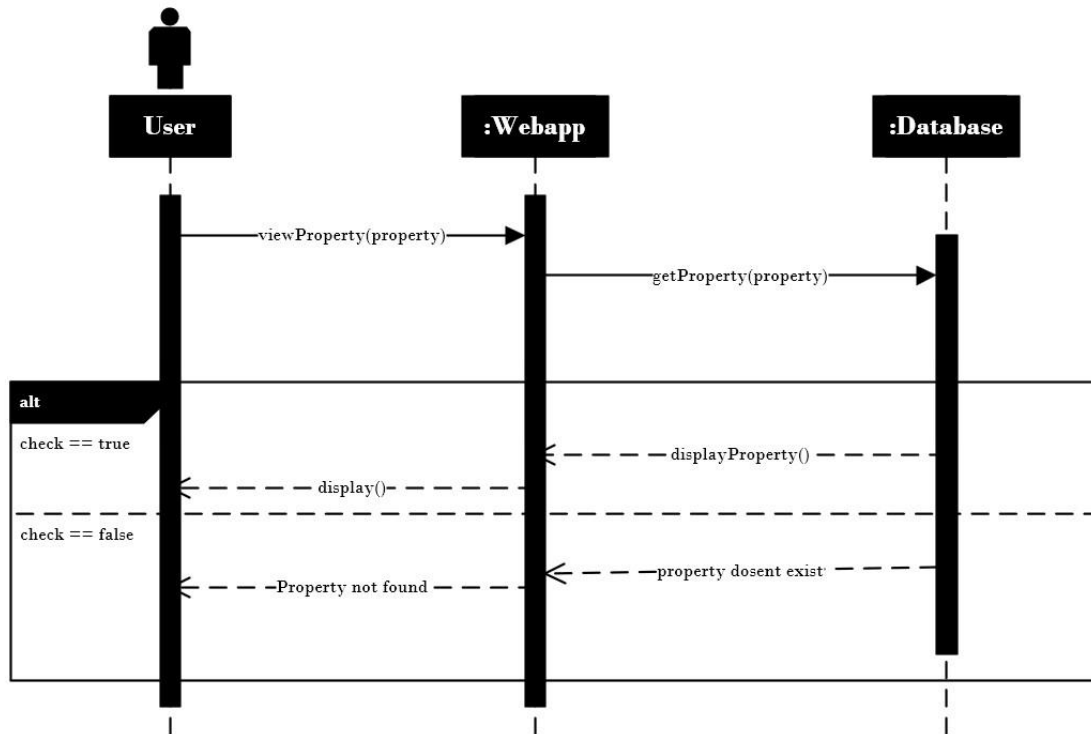


Figure 2.2:10 View property

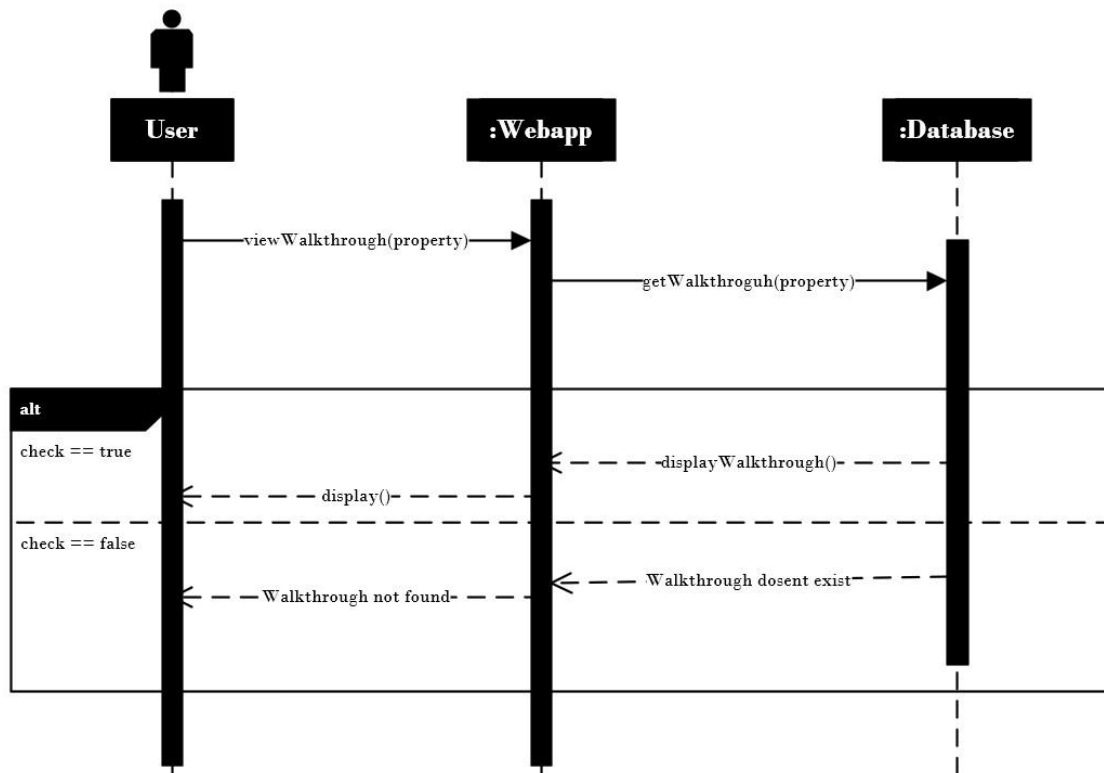


Figure 2.2:11 View Walkthrough

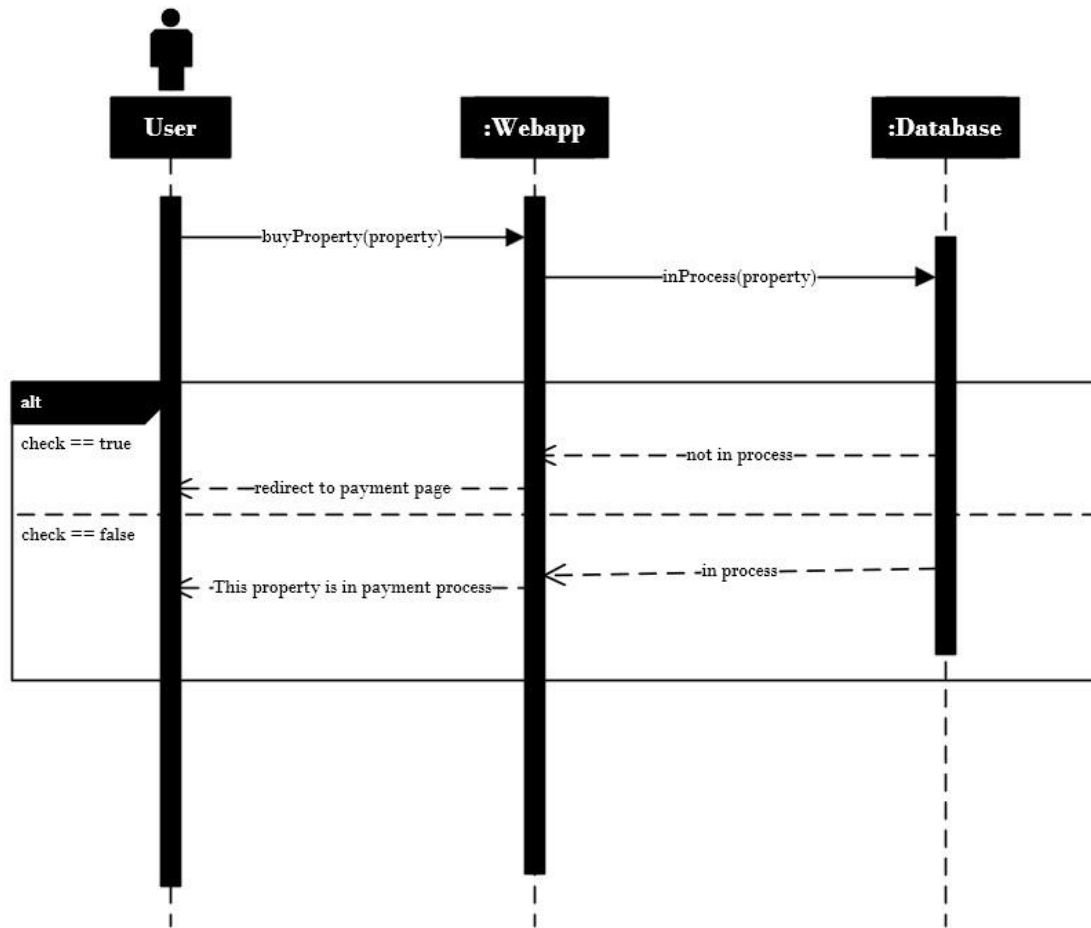


Figure 2.2:12 Buy Property

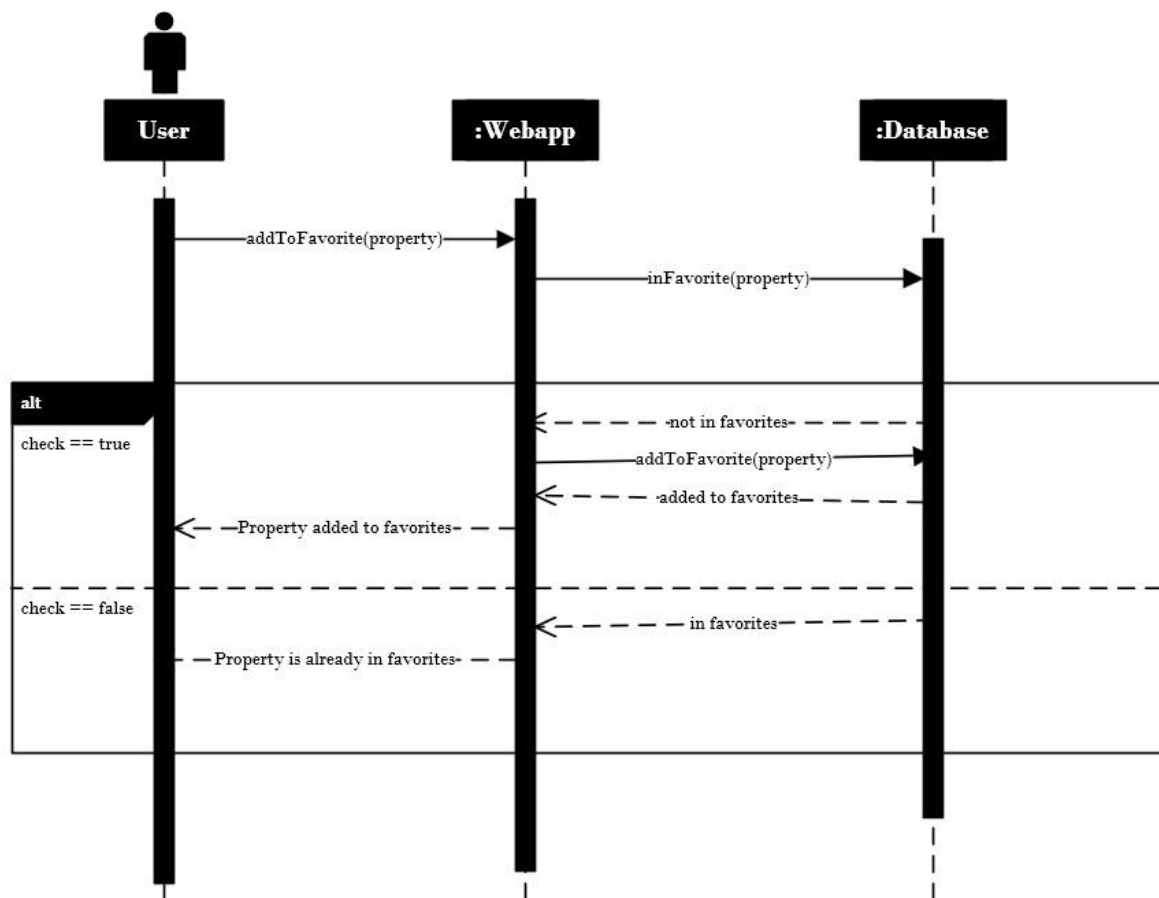


Figure 2.2:13 AddToFavorite

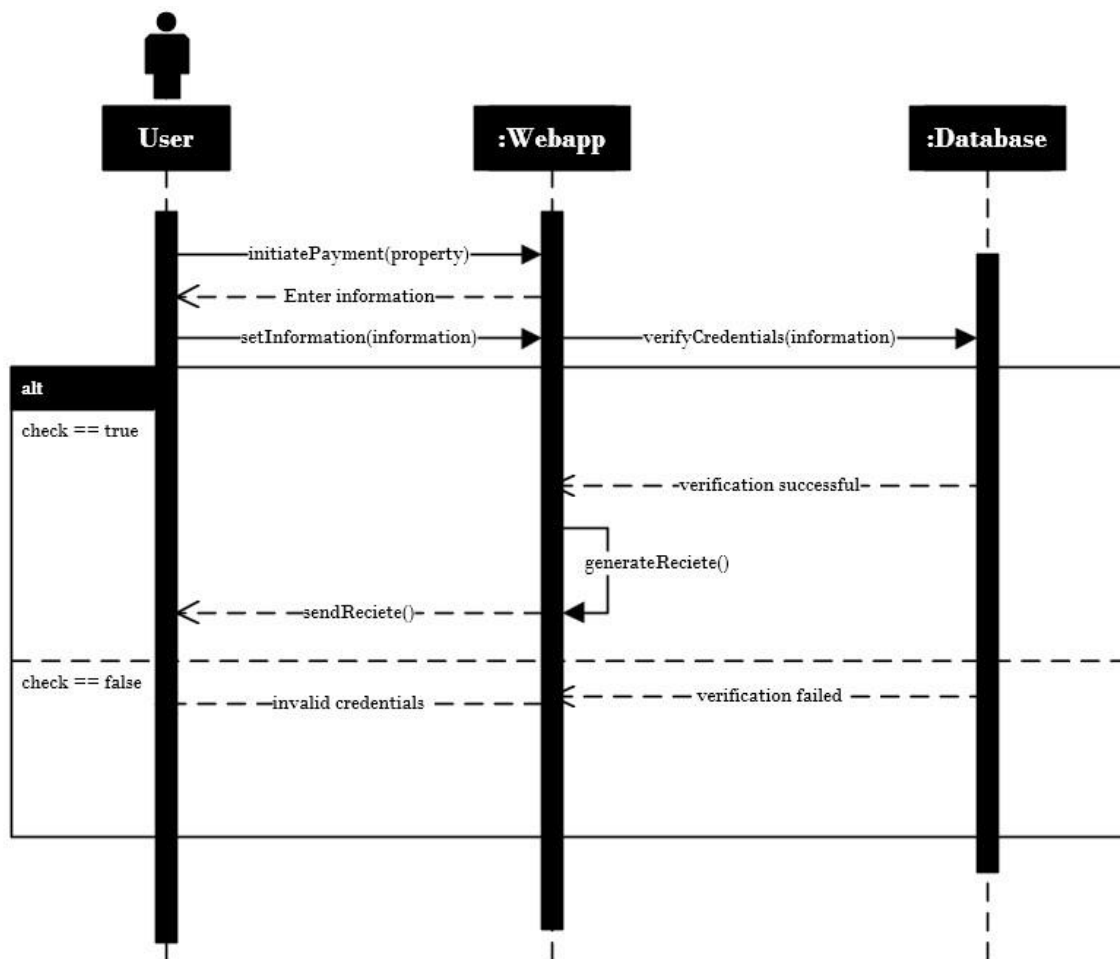


Figure 2.2:14 InitiatePayment

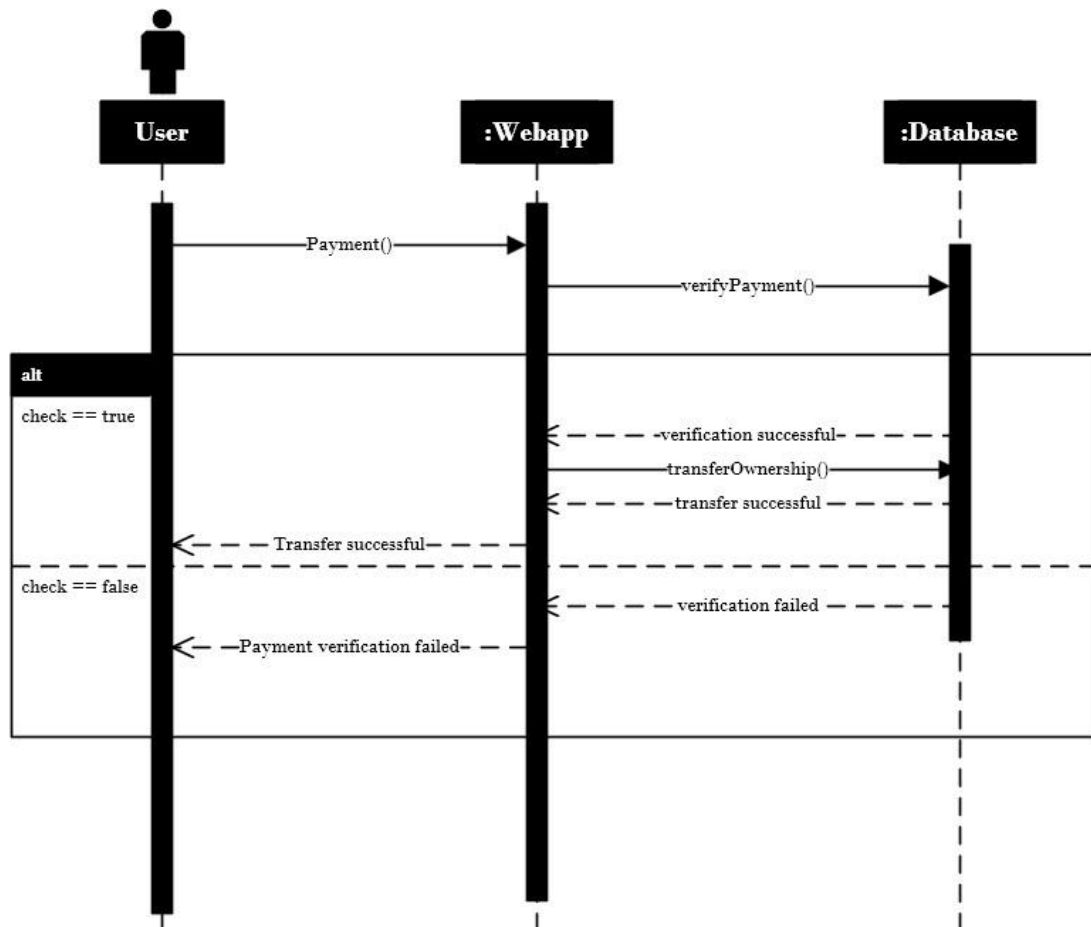


Figure 2.2:15 Payment

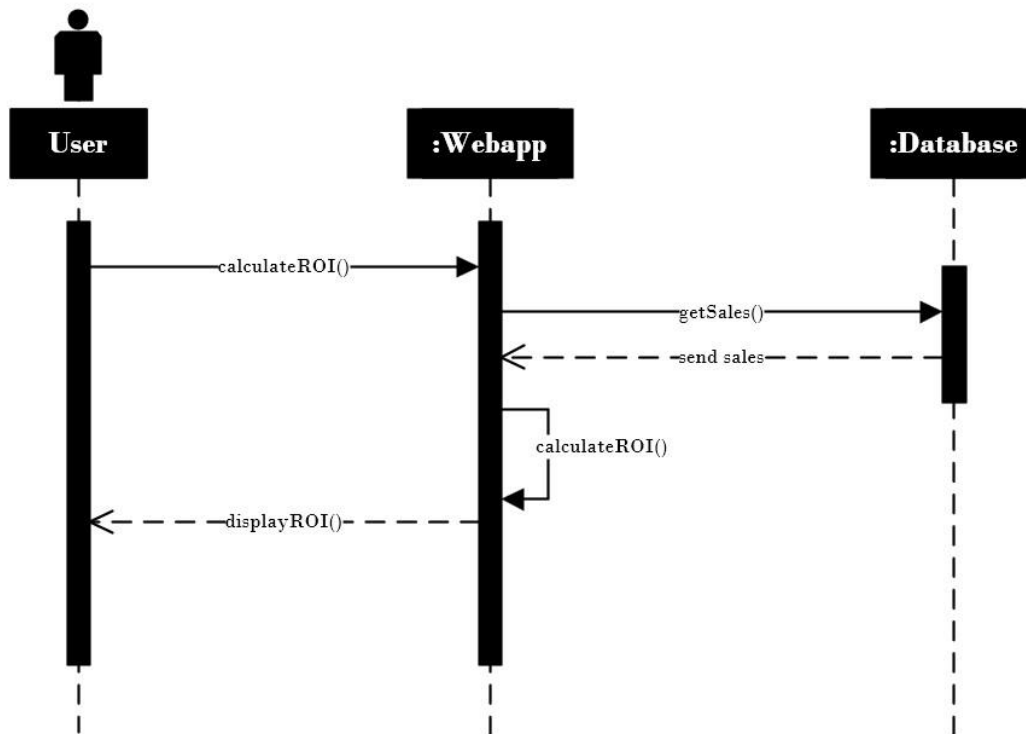


Figure 2.2:16 Calculate ROI

2.6. Domain Model

Part of your initial architectural modeling efforts, particularly for a business application, will likely include the development of high-level domain model as you see in Fig. 2.3. This model should be very slim, capturing the main business entities and the relationships between them. Some people consider this type of model to be an initial requirements model instead of an initial architecture model

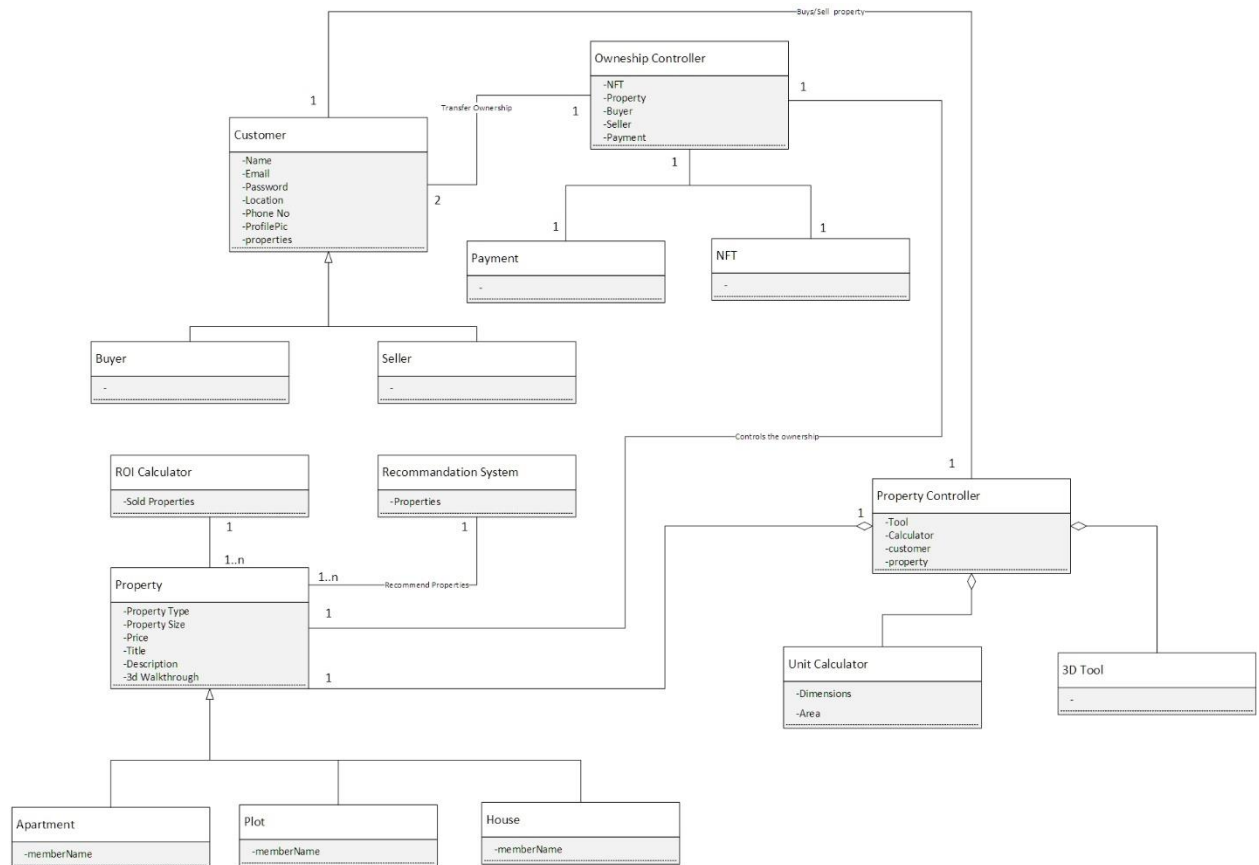


Figure 2.3: Domain Model

Chapter 3

System Design

The purpose of this chapter is to provide information that is complementary to the code. Without an adequate design that delivers required function as well as quality attributes, the project will fail. But communicating architecture to its stakeholders is as important a job as creating it in the first place.

There are two views that are considered while defining software architecture. There are specific design artifacts that belong to each view. Description of such artifacts is given below. *You may select the artifacts depending on the nature of your project.*

- Structural View
 - Architecture diagram
 - Module structure diagram
 - Component diagram
 - Class diagram
- Behavioral View
 - Sequence diagram
 - Activity diagram
 - State machine diagram

At a high level, a software architecture document includes:

1. An outline description of the software design, including major software components and their interactions.
2. A common understanding of requirements, constraints and principles that influence the architecture.
3. A description of the hardware and software platforms on which the system is built and deployed.
4. Explicit justification of how the architecture satisfies the above mentioned points.

Design pattern is a description or template for how to solve a problem that can be used in many different situations. Object-oriented design patterns typically show relationships and interactions between classes or objects, without specifying the final application classes or objects that are involved.

It is important that you justify its design, for example, by discussing the implications of constraints on your solution and different design choices, and then giving reasons for making the choices you did. At each stage of the design you should mention what kind of design patterns have you followed while designing your system. You should identify which design pattern among the existing patterns are you following while designing your project.

3.1. Software Architecture

Software architecture is described as the organization or structure of a system, where the system represents a collection of components that accomplish a specific function or set of functions. When getting started with your design, keep in mind the key principles that will help you to create an architecture that adheres to proven principles, minimizes costs and maintenance requirements, and promotes usability and extendibility. The major decisions that you must make, and which help to ensure that you consider all of the important factors as you begin and then iteratively develop your architecture design are

- Determine the Application Type
- Determine the Deployment Strategy
- Determine the Appropriate Technologies
- Determine the Quality Attributes
- Determine the Crosscutting Concerns

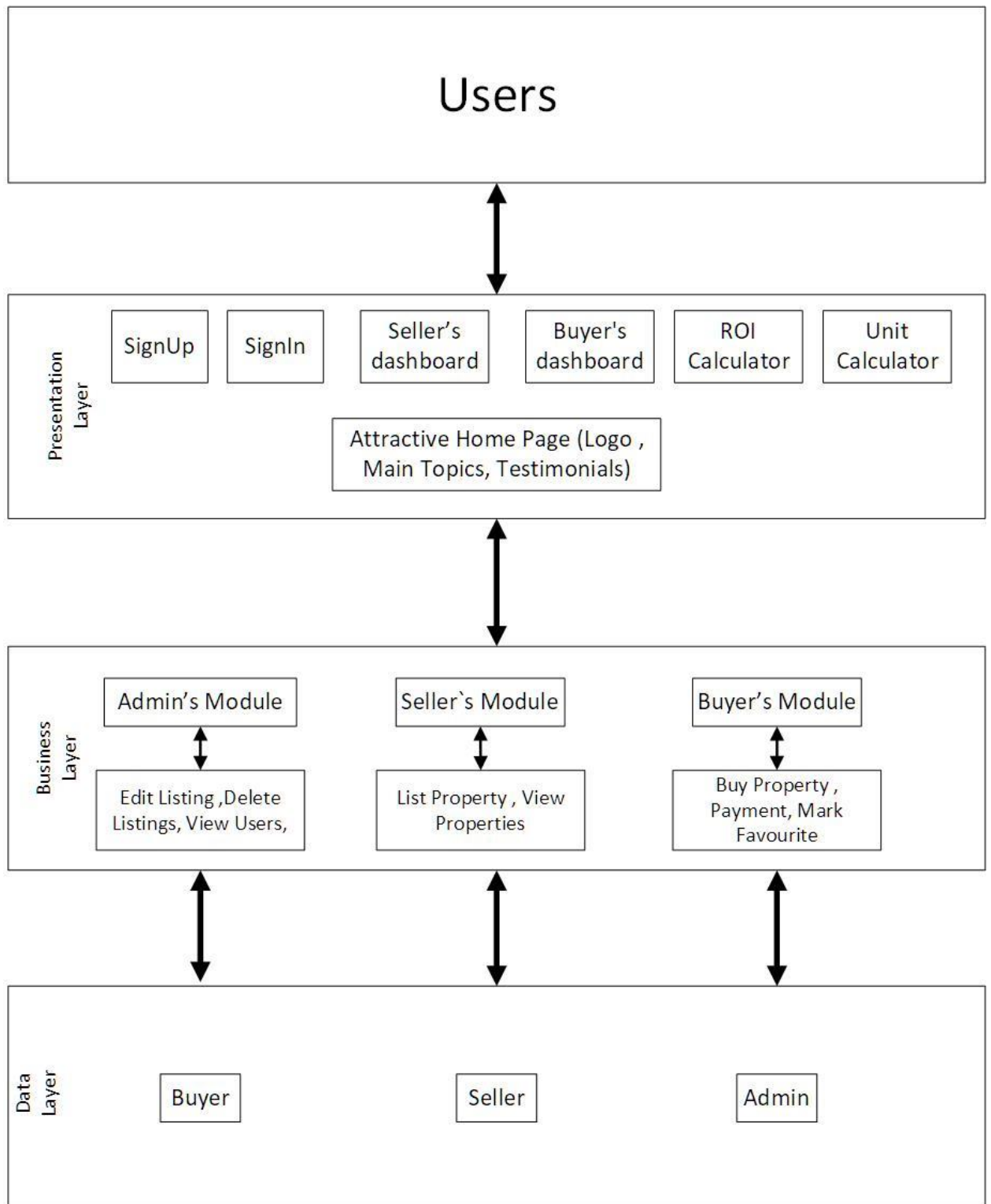


Figure 3.1: Software Architecture Diagram

Fig. 3.1 illustrates common application architecture with components grouped by different areas of concern. You should think of architecture as the strategy for how you will build the system. An architectural “layer” is the top-level logical view, or an abstraction, of your design.

3.2. Class Diagram

Class Diagram as shown in Fig. 3.2 provides an overview of the target system by describing the objects and classes inside the system and the relationships between them. It provides a wide variety of usages; from modeling the domain-specific data structure to detailed design of the target system.

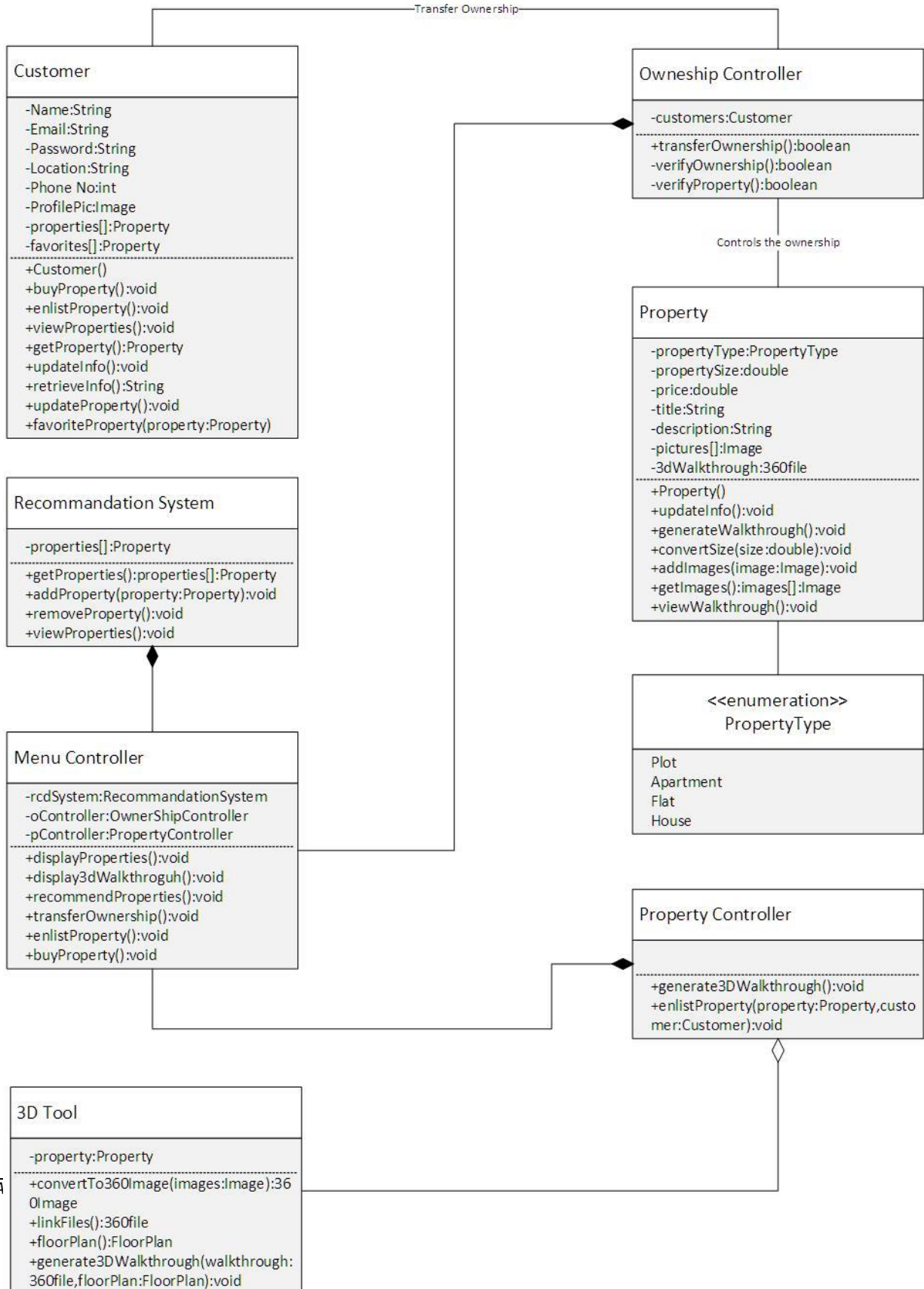


Figure 3.2: Class Diagram

3.3. Sequence Diagram

Sequence diagrams, when used in conjunction with class diagrams; provide an extremely effective communication mechanism. UML sequence diagrams as shown in Fig. 3.3 are used to show how objects interact in a given situation.

You can use a class diagram to illustrate the relationships between the classes, and the sequence diagram lets you show the messages sent among the instances of these classes and the order in which they are sent. When an object sends a message to another object, it implies that the two classes have a relationship that must be shown on a class diagram.

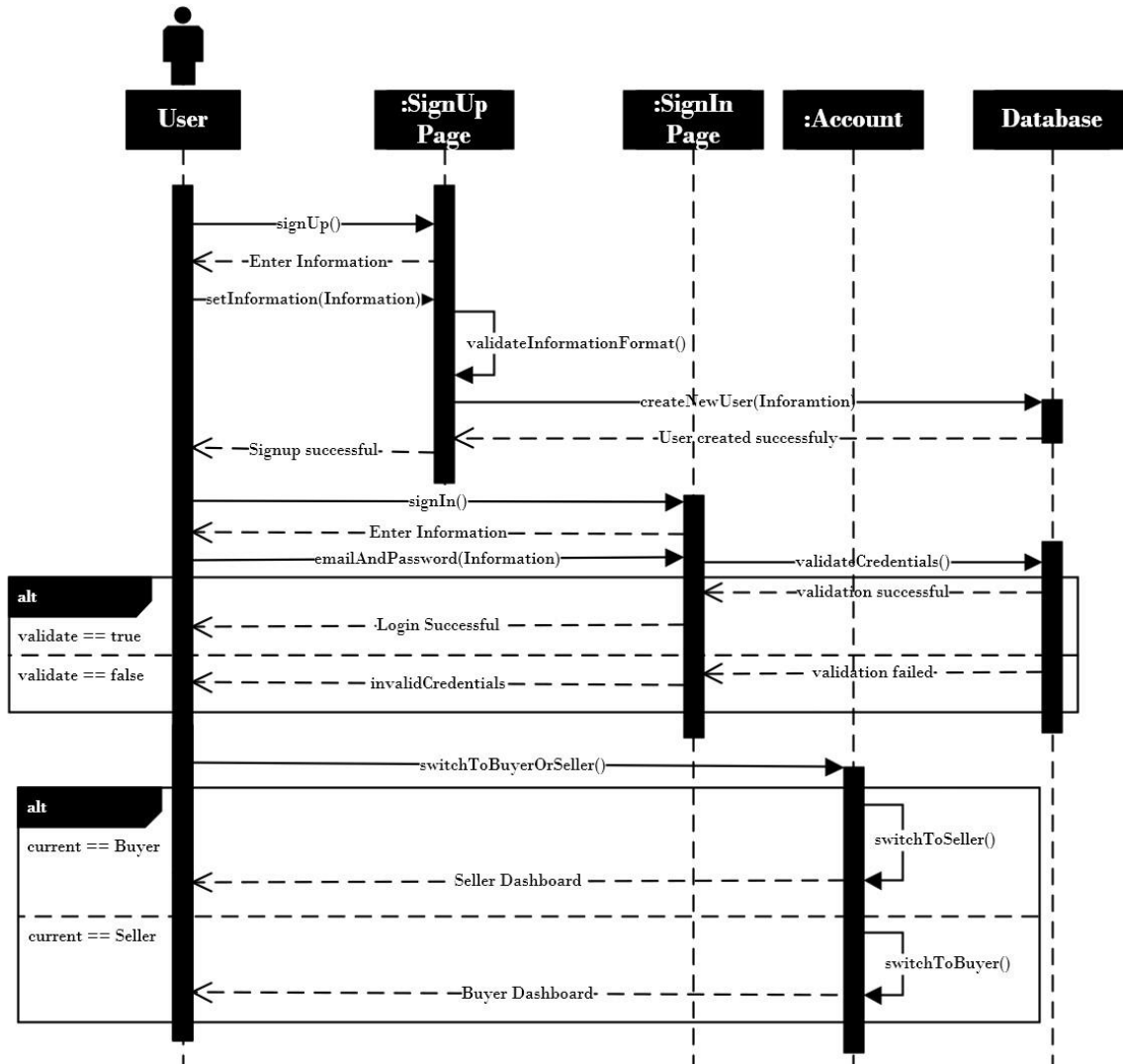


Figure 3.3:1 Sequence Diagram

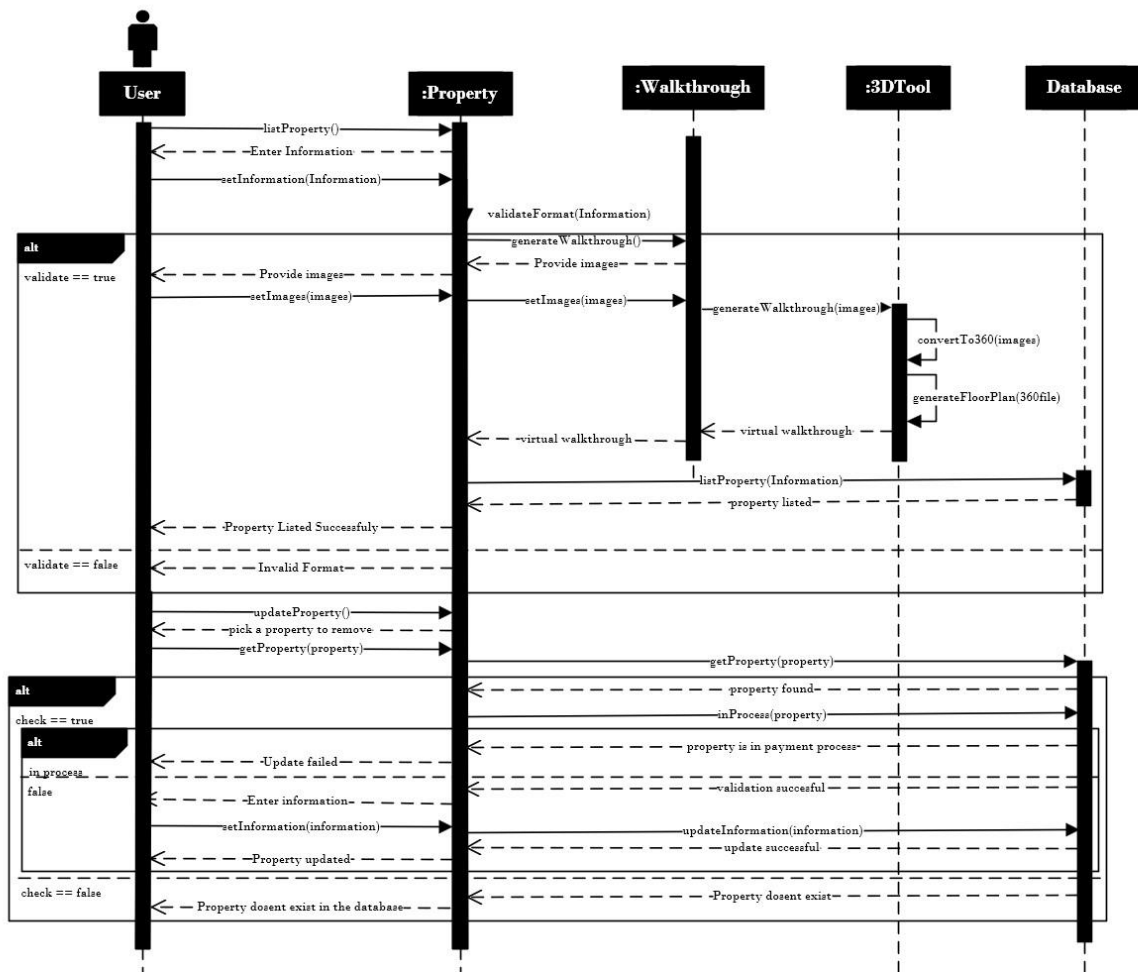


Figure 3.3:2 Sequence Diagram

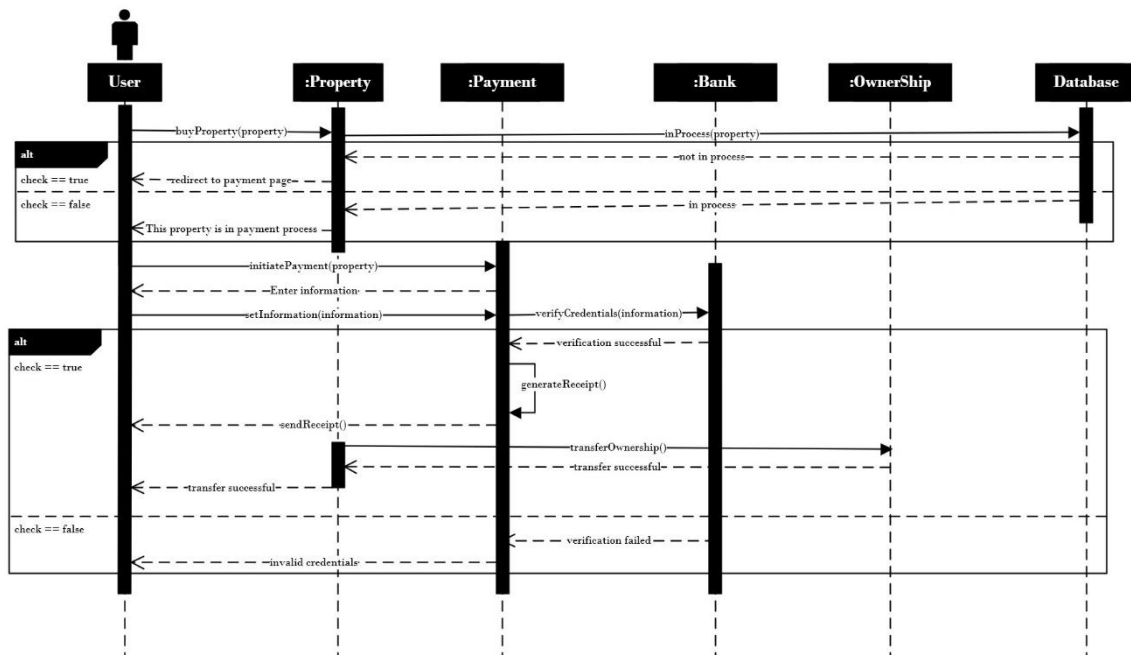


Figure 3.3:3 Sequence Diagram

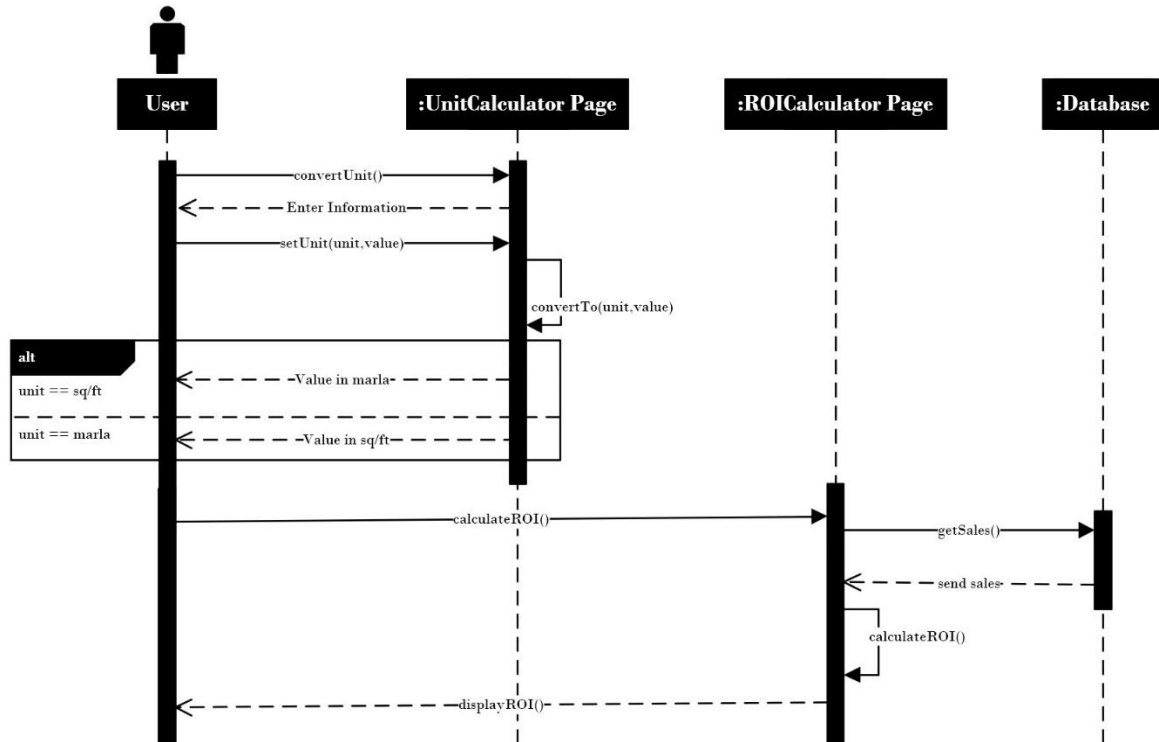


Figure 3.3:4 Sequence Diagram

3.4. Entity Relationship Diagram

Entity relationship model diagram (ERD) is a conceptual representation of the data in a software system. During detail design this model is mapped in to the physical database model. There are different diagramming conventions available for creating ER diagrams. A sample ERD is shown in Fig. 3.4.

.At this stage you may not be able to provide a comprehensive ERD therefore you may provide the refined ERD later in design chapter of the document where you will provide the fully attributed ERD such as primary keys, cardinality constraints etc.

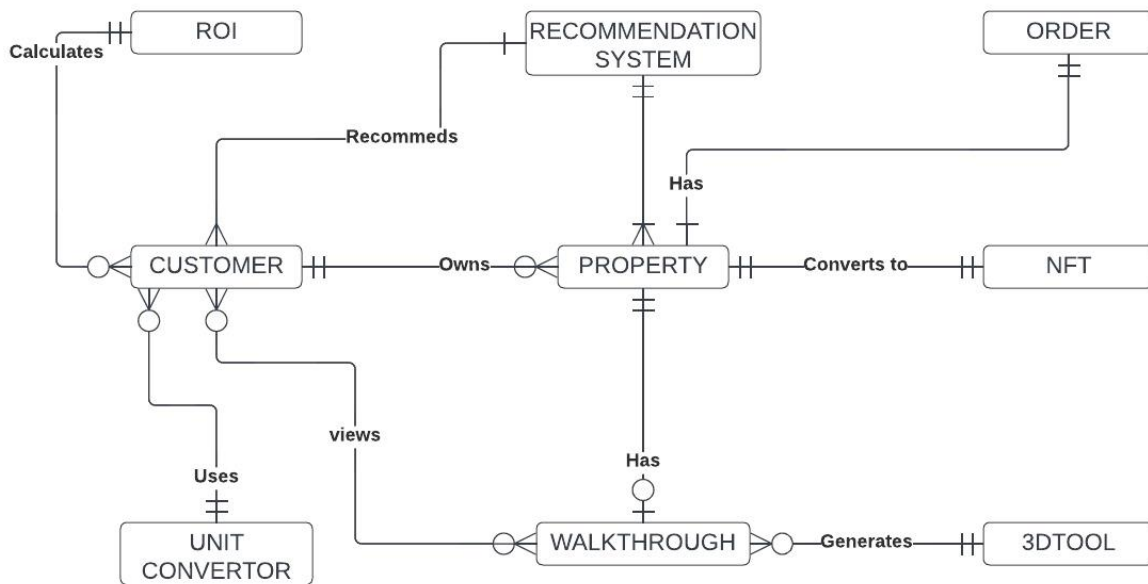


Figure 3.4: Entity Relationship Diagram

3.5. Database Schema

A database schema represents the logical configuration of all or part of a relational database. It can exist both as a visual representation and as a set of rules known as integrity constraints that govern a database. These rules are expressed in a data definition language, such as SQL. A database schema indicates how the entities that make up the database relate to one another, including tables, views, stored procedures, and more. A database schema includes information related to primary and secondary keys, normalization and indexing.

You may present database scheme using front end tool of any DBMS or any other design tools such as Visio or Enterprise Architecture. A sample database scheme is shown in Fig. 3.5.

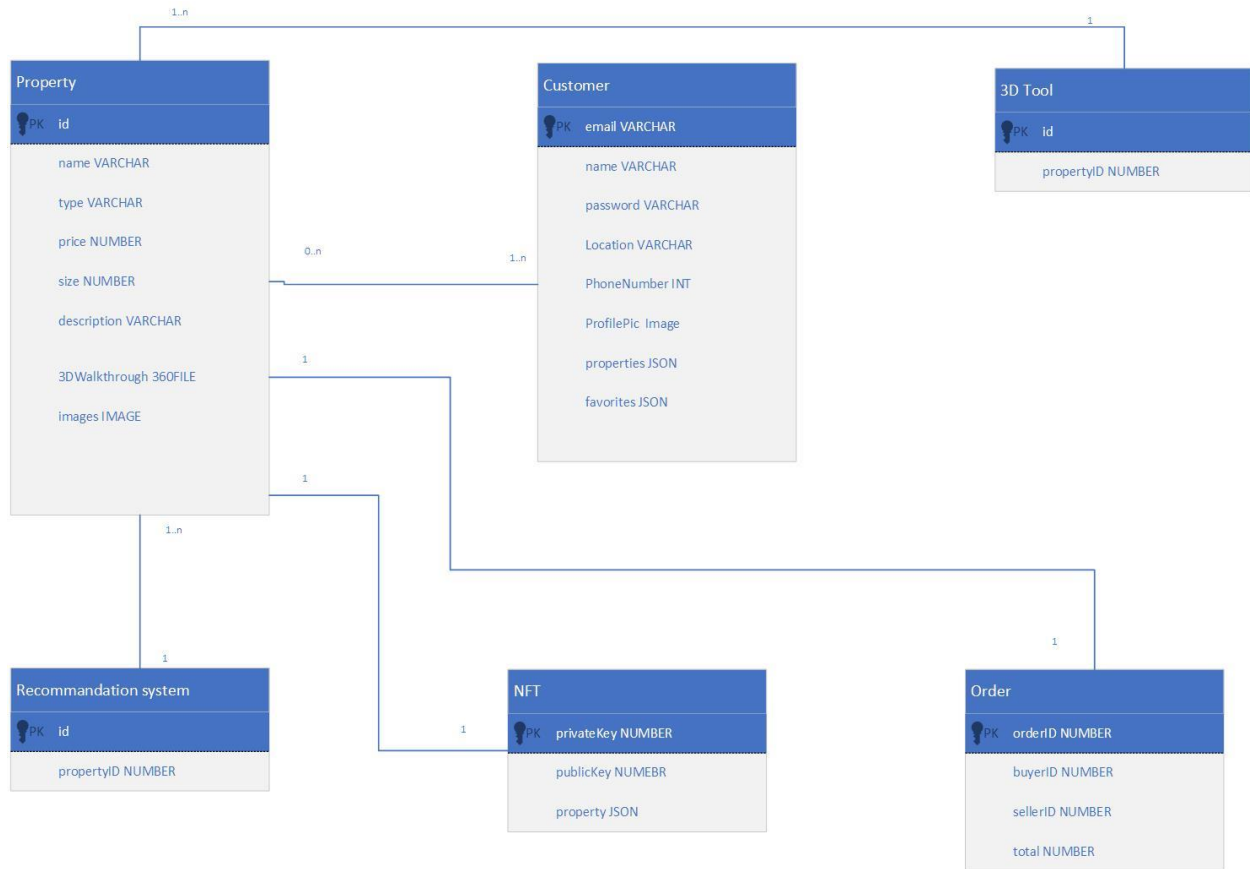



Figure 3.5: Database Schema

3.6. User Interface Design


User Interface (UI) Design focuses on anticipating what users might need to do and ensuring that the interface has elements that are easy to access, understand, and use to facilitate those actions. UI brings together concepts from interaction design, visual design, and information architecture.

You should describe the UI design in such a way that it remains simple and consistent along different views. Common GUI elements are shown in the Fig. 3.6. You should describe the UI design of each page.



Search

Update Your Profile



Change Photo

Change Profile Picture

Username

Email Address

First Name

Last Name

Save Settings

Update profile

Complete Payment

Owner


CVV

Card Number

Expiration Date

January

2016




AREA IN SQUARE FEET

Type To calculate

AREA IN MARLA

Figure 3.6: GUI elements

Login Page as shown in Fig. 3.7 will contain one text field and one password field. Email must include @ and .com and we cannot leave both fields empty.



Create an account

Sign in

Your Email

Password

[Forgot password?](#)

Sign In

Figure 3.7 Login Page UI Design

Chapter 4

Software Development

The Implementation section is similar to the Specification and Design section in that it describes the system, but it does so at a finer level of detail, down to the code level. This section is about the realization of the concepts and ideas developed earlier. It can also describe any problems that may have arisen during implementation and how you dealt with them.

Make sure that the system design corresponds to the implementation of the project. If there is no relationship between design and implementation, it may downgrade your score in FYP.

You should also mention any unforeseen problems you encountered when implementing the system and how and to what extent you overcame them. Common problems are:

- Difficulties involving existing software, because of e.g.,
 - Its complexity,
 - Lack of documentation;
- Lack of suitable supporting software
- Overambitious project aims.

A seemingly disproportionate amount of project time can be taken up in dealing with such problems. The Implementation section gives you the opportunity to show where that most of the effort has been spent.

4.1. Coding Standards

Proper Indentation is used in website to increase Readability.

Proper names are used for class because it helps in code reuse and helps to detect error easily.

4.2. Development Environment

We have used VScode to develop the project and have used Html/Css/bootstrap and React to code frontend.

Npm is used to create react app and and install router-dom and react-bootstrap dependencies.

4.3. Software Description

Unit Calculator

Snippet 1

```
function weightConverter(event) { let n = (event.target.value *  
272.25).toFixed(2); setInSqft(n); }
```

Description: This function takes 1 argument , value in Marla and formula is applied to convert unit to Square Feet.

Snippet 2

```
function marla(event) { let m = (event.target.value /  
272.25).toFixed(2); setInMarla(m); }
```

Description: This function takes 1 argument , value in Squarefeet and formula is applied to convert Square Feet to Marla.