



BSIT-F20-008

03-135172-026 ZAINAB JAMSHAD

03-135172-020 RUBAB AKRAM

Architect Zone

In partial fulfilment of the requirements for the degree of
Bachelor of Science in Information Technology

Supervisor: Jehanzeb Ahmed

Department of Computer Sciences
Bahria University, Lahore Campus

July 2021

Certificate



We accept the work contained in the report titled

“Architect Zone”

Written by

ZAINAB JAMSHAD

RUBAB AKRAM

As a confirmation to the required standard for the partial fulfilment of the degree of
Bachelor of Science in Information Technology.

Approved by:

Supervisor: Jehanzeb Ahmed

(Signature)

July 30, 2021

DECLARATION

We hereby declare that this project report is based on our original work except for citations and quotations, which have been duly acknowledged. We also declare that it has not been previously and concurrently submitted for any other degree or award at Bahria University or other institutions.

Enrolment	Name	Signature
03-135172-026	ZAINAB JAMSHAD	
03-135172-020	RUBAB AKRAM	

Date : July 30, 2021

Specially dedicated to
My beloved grandmother, mother and father
ZAINAB JAMSHAI
My beloved grandmother, mother and father
RUBAB AKRAM

ACKNOWLEDGEMENTS

We would like to thank everyone who had contributed to the successful completion of this project. We would like to express our gratitude to our project supervisor, Mr JEHANZEB AHMED for his invaluable advice, guidance and his enormous patience throughout the development of the project.

In addition, we would also like to express our gratitude to our loving parent and friends who had helped and given us encouragement.

ZAINAB JAMSHAIID

RUBAB AKRAM

Architect Zone

ABSTRACT

We are going to build an application that will run on android phones titled “Architect Zone”. Architect zone is for the architects and for the public (who wants to design or construct their home, building or factory etc.). The architects are treated as an entrepreneur who are register in the application for providing their services to the public. Entrepreneur (architect) and end-users (public) first create their profile using their email and register in application by providing important information related to them and then they will be able to perform their relevant tasks (searching, hiring etc.).

TABLE OF CONTENTS

DECLARATION	ii
ACKNOWLEDGEMENTS	iv
ABSTRACT	v
TABLE OF CONTENTS	vi
LIST OF TABLES	x
LIST OF FIGURES	xi

CHAPTERS

1	INTRODUCTION	1
	1.1 Background	1
	1.2 Problem Statements	2
	1.3 Aims and Objectives	2
	1.4 Scope of Project	2
2	SOFTWARE REQUIREMENT SPECIFICATIONS	3
	2.1 Introduction	3
	2.1.1 Purpose	3
	2.1.2 Intended Audience and Reading Suggestions	3
	2.1.3 Product Scope	3
	2.1.4 References	4
	2.2 Overall Description	4
	2.2.1 Product Perspective	4
	2.2.2 User Classes and Characteristics	4
	2.2.3 Operating Environment	4

2.2.4	Design and Implementation Constraints	5
2.2.5	Assumptions and Dependencies	5
2.3	System Use Case	5
2.3.1	Registration (U1)	7
2.3.2	Admin Verify User Account (U2)	7
2.3.3	Login (U3)	8
2.3.4	Architect Add Details (U4)	9
2.3.5	Customer Select Service (U5)	9
2.3.6	Logout (U6)	10
2.4	System Features	11
2.5	Other Non-functional Requirement	11
2.5.1	Performance Requirements	11
2.5.2	Safety Requirements	11
2.5.3	Security Requirements	11
2.5.4	Software Quality Attributes	11
3	DESIGN AND METHODOLOGY	13
3.1	Introduction	13
3.2	User Description	13
3.2.1	Registration (U1)	14
3.2.2	Admin Verify User Account (U2)	14
3.2.3	Login (U3)	15
3.2.4	Architect Add Details (U4)	16
3.2.5	Customer Select Service (U5)	16
3.2.6	Logout (U6)	17
3.3	Use Case Diagram	18
3.3.1	Main Use Case Diagram	18
3.4	Domain Model	20
3.5	Sequence Diagram	20
3.5.1	Registration (Users)	20
3.5.2	Login (Users)	21
3.5.3	Registration (Architect)	22
3.5.4	Login (Architect)	23

3.5.5	Details for Architect	24
3.5.6	Services for Customer/Users	25
3.5.7	Update Details	26
3.6	Operation Contracts	26
3.6.1	Registration	26
3.6.2	Admin Verify User Account	27
3.6.3	Login	27
3.6.4	Architects Add Details	28
3.6.5	Customers Select Service	28
3.6.6	Logout	29
3.7	Design Class Diagram	29
3.8	Data Model	31
3.8.1	Entity Relationship Diagram	31
4	IMPLEMENTATION	32
4.1	Splash Screen	33
4.2	Admin Panel	34
4.2.1	Admin Sign-in	34
4.2.2	Admin Dashboard	36
4.2.3	Admin View Customer	38
4.2.4	Admin Verified Images	40
4.2.5	Admin Logout	42
4.3	Architect Panel	44
4.3.1	Admin Sign-up	45
4.3.2	Architect Login	46
4.3.3	Architect Dashboard	47
4.3.4	Architect Add Data	48
4.3.5	Architect Update Account	49
4.3.6	Architect Logout	50
4.4	Customer Panel	51
4.4.1	Customer Sign-up	52
4.4.2	Customer Login	53
4.4.3	Customer View Images	54

	4.4.4 Customer Logout	55
5.	RESULTS AND DISCUSSIONS	57
6.	CONCLUSION	58

LIST OF TABLES

TABLE	TITLE	PAGE
	Table 2-1 Registration (U1)	7
	Table 2-2 Admin Verify User Account (U2)	7
	Table 2-3 Login (U3)	8
	Table 2-4 Architect Add Details (U4)	9
	Table 2-5 Customer Select Service (U5)	9
	Table 2-6 Logout (U6)	10
	Table 3-1 Registration (U1)	14
	Table 3-2 Admin Verify User Account (U2)	14
	Table 3-3 Login (U3)	15
	Table 3-4 Architect Add Details (U4)	16
	Table 3-5 Customer Select Service (U5)	16
	Table 3-6 Logout (U6)	17
	Table 3-7 Registration	26
	Table 3-13 Admin Verify User Account	27
	Table 3-10 Login	27
	Table 3-12 Architects Add Details	28
	Table 3-11 Customer Select Service	28
	Table 3-16 Logout	29

LIST OF FIGURES

FIGURE	TITLE	PAGE
Figure 2-1	Main System Use Case	6
Figure 3-1	Main System Use Case Diagram	19
Figure 3-2	Main System Domain Model Diagram	20
Figure 3-3	Sequence Diagram 1	21
Figure 3-4	Sequence Diagram 2	22
Figure 3-5	Sequence Diagram 3	23
Figure 3-6	Sequence Diagram 4	24
Figure 3-7	Sequence Diagram 5	25
Figure 3-8	Sequence Diagram 6	25
Figure 3-9	Sequence Diagram 7	26
Figure 3-10	Main System Design Class Diagram	30
Figure 3-12	Main System Entity Relationship Diagram	31
Figure 4-1	Splash Screen	33
Figure 4-2	Admin Sign-in	35
Figure 4-3	Admin Dashboard	37
Figure 4-4	Admin View Customer	39
Figure 4-5	Verified Images	41
Figure 4-6	Admin Logout	43
Figure 4-7	Architect Sign-up	45

Figure 4-8 Architect Login	46
Figure 4-9 Architect Dashboard	47
Figure 4-10 Add Data	48
Figure 4-11 Update Account	49
Figure 4-12 Architect Logout	50
Figure 4-13 Customer Sign-up	52
Figure 4-14 Customer Login	53
Figure 4-15 View Images	54
Figure 4-16 Customer Logout	55

CHAPTER 1

INTRODUCTION

1.1 Background

The businesses are evolving and it is important to upgrade yourself for better survive in the competitive environment. Every business is switching from manually managing to digital and upgrading to e-commerce. E-commerce also helps in managing the business even if you are extending your business to a global level. It also reduces the workforce and helps you to manage things with just one application with better checks and control. This is very important to adopt digital technology and follow the trend and demand of the time. Every market is upgrading and if you want to compete with your rivals, you need to make some better strategy and fulfil the demand of the public before the rival party to keep progressing in your business, which is the ultimate goal.

We are doing this research and to develop a framework to help people to deal with the issues that they are facing and better facilitate them by making a mobile application. Because everybody is using a smartphone, which is mostly android based operating system. Mobile applications are facilitating all people who are using mobile applications E.g. for e-commerce people are using mobile applications like Daraz.pk, aliexpress. For online payments people are using easy paisa, jazz cash, Etc. for traveling, people are using uber, SWVL, airlift ETC. so the trend of application is very much high in smartphone users they go for mobile applications for their daily routine tasks. Because it provides ease and facilitates them in managing their tasks easily.

1.2 Problem Statements

Construction is multibillion-dollar industry. People are investing a lot of money in construction. There are huge fees are paid to the architect and contractor for building houses, bridges, roads, buildings. Some architects and contractors are overcharging and some are paid less than their worth. It is very important to evaluate the worth of contractors and Architects. so that is why we are going to make a platform where we will be capture all information about the contractors and architects and other concerned parties.

1.3 Aims and Objectives

The objectives of the project is shown as following:

- Providing easy way of interaction between end user and entrepreneur.
- Providing detail and verified profiles of entrepreneur and end-user.
- Providing fast access to the relevant entrepreneur.
- Facilitates the end users and Architect.

1.4 Scope of Project

Architect Zone is an android-based platform for the architects and public. A person who wants to design or construct something (home, building or factory etc.) can contact with the architect by using the platform who are registered in Architect Zone. An end-user can search for their relevant entrepreneur (architects) and also can see their profile.

CHAPTER 2

SOFTWARE REQUIREMENT SPECIFICATIONS

2.1 Introduction

2.1.1 Purpose

The purpose of this document is to present a detailed description of the android application Architect Zone. It will explain the purpose and features of the application, the interfaces of the application, what the application will do and the constraints under which it must operate. Our objective is to help the people who wants to construct their houses. This document is intended for users of the application.

2.1.2 Intended Audience and Reading Suggestions

- General people who are looking to build their houses.
- Entrepreneurs who will develop the houses.

2.1.3 Product Scope

Architect Zone is an android-based platform for the architects and public. A person who wants to design or construct something (house, building, commercial area etc.) can contact with the architect by using the platform who are registered in Architect Zone. An end user can search for their relevant entrepreneur (architects) and also can see their profile and choose according to their demands.

2.1.4 References

None.

2.2 Overall Description

2.2.1 Product Perspective

Architect Zone is for the architects and for the public who wants to design or construct their houses building, commercial areas etc. The architects are treated as an entrepreneur who are register in the application for providing their services to the public.

Entrepreneur (architects) and end-users (general public) first create their profile as entrepreneurs or general using their email or mobile number to register in the application then they will be able to perform their relevant task.

2.2.2 User Classes and Characteristics

There will be three user classes in our application

- **General Public:**

General People who want to design their houses, buildings or commercial area.

- **Entrepreneur:**

Entrepreneur will be the architects who will design the houses, buildings or commercial area.

- **Admin:**

Admin will be the one who manage the application.

2.2.3 Operating Environment

The minimum requirement for the development of Architect Zone is on android 4.0 but to get more accurate and precise result we may perform it on updated versions.

2.2.4 Design and Implementation Constraints

Android Studio is a professional tool for developing android applications so we use it for the development purpose. We use xml, java for front-end development. We use postman for testing API of application. We use Java, Jason, Kotlin for implanting backend development. We also use different online tool for development like firebase, Jason to Java converter etc. We use adobe illustrator and adobe photoshop for designing frontend logo and buttons.

2.2.5 Assumptions and Dependencies

Architect Zone will be an android application therefore; updated version of android will be requiring running the app smoothly.

2.3 System Use Case

Main Use Case:

In Figure 2-1 Main System Use Case is shown.

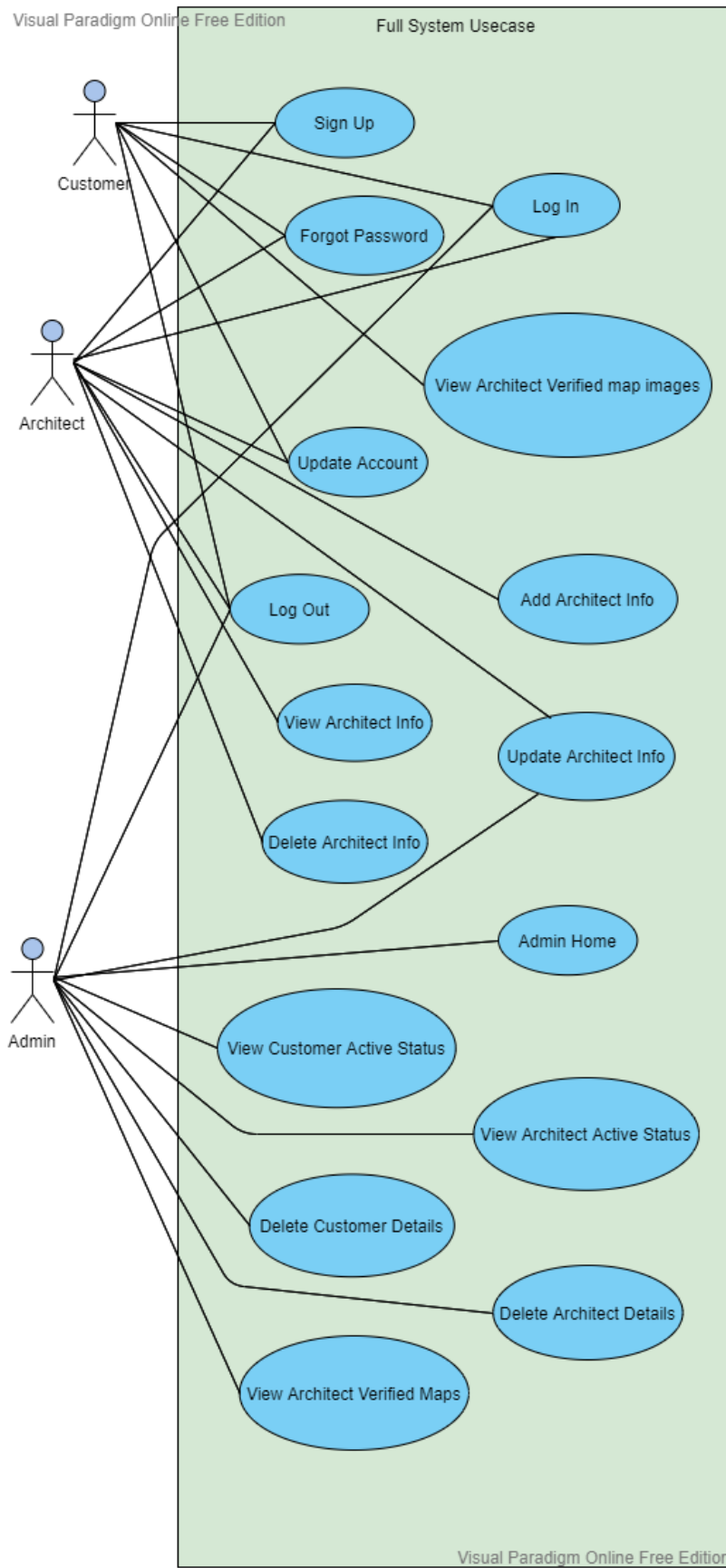


Figure 2-1 Main System Use Case

2.3.1 Registration (U1)

There are many parts of the use case description:

Table 2-1 Registration (U1)

Name and ID	Sign Up (U1)
Objective	Customer and Architect can register themselves to use the application.
Priority	High.
Actor	Customer and Architecture.
Preconditions	User must open the application.
Basic flow or Happy path	User will simply give his/her credentials to sign up on App.
Trigger	User wants to view services.
Alternate flows	User can simply search the application but would not register himself.
Post conditions	User can get the access to view available services.
Issues	User type incorrect credentials.

2.3.2 Admin Verify User Account (U2)

There are many parts of the use case description:

Table 2-2 Admin Verify User Account (U2)

Name and ID	Admin Verify User Account (U2)
Objective	After registration, user account will verified by the admin so user can work further.
Priority	High.
Actor	Admin.
Preconditions	User must Sign up.

Basic flow or Happy path	Admin can active and inactive user account.
Trigger	Admins accept or reject the Request of user.
Alternate flows	Architect is not able to provide the service.
Post conditions	User can login.
Issues	No issue.

2.3.3 Login (U3)

There are many parts of the use case description:

Table 2-3 Login (U3)

Name and ID	Login (U3)
Objective	User login to the application to avail the services.
Priority	High.
Actor	Customer and Architect.
Preconditions	User should have signed up into application.
Basic flow or Happy path	User can login to application by giving required credentials.
Trigger	User wants to view services.
Alternate flows	User can simply search the application but would not register himself.
Post conditions	User can get the access to view the available services.
Issues	User give wrong credentials.

2.3.4 Architect Add Details (U4)

There are many parts of the use case description:

Table 2-4 Architect Add Details (U4)

Name and ID	Architect Add Details (U4)
Objective	Architect can add his/her work to his/her account.
Priority	High.
Actor	Architect.
Preconditions	Architect should have login to his/her account and his/her account should verified by admin.
Basic flow or Happy path	Architect will login to his/her account and add map of his/her work.
Trigger	Architect must provide the service.
Alternate flows	Architect did not add his/her work.
Post conditions	Architect will go to the customer and perform the required service.
Issues	No issue.

2.3.5 Customer Select Service (U5)

There are many parts of the use case description:

Table 2-5 Customer Select Service (U5)

Name and ID	Customer Select Service (U5)
Objective	Customer can see the maps of architects and then choose architect according to his/her requirements.
Priority	Medium.
Actor	Customer.
Preconditions	Customer should have login to the application.

Basic flow or Happy path	All the available services and works of Architects are given to Application. Customer just must choose the required service by clicking on the service.
Trigger	Customer wants to use our services.
Alternate flows	Customer can simply see the services and works of Architectures but would not select any service.
Post conditions	Customer will select the architect of his choice.
Issues	No issue.

2.3.6 Logout (U6)

There are many parts of the use case description:

Table 2-6 Logout (U6)

Name and ID	Logout (U6)
Objective	Customer and architect can logout themselves from App after signing in.
Priority	Medium.
Actor	Customer and Architect.
Preconditions	A user must Sign In first to logout.
Basic flow or Happy path	Users will logout by clicking the logout button.
Trigger	User wants to logout from App.
Alternate flows	User did not want to logout after sign In.
Post conditions	No post condition as it is the last step.
Issues	No issue.

2.4 System Features

First, End-User or Entrepreneur can create their accounts using email. A person who wants to design or construct something (home, building or factory etc.) can contact with the architect, constructor or both by using the application who are registered in Architect Zone. An end-user can search for their relevant entrepreneur (constructors, architects) and can see their profile.

- Registering, Creating and updating profile
- Searching for required Entrepreneur
- Messaging / Chatting service
- Contact info
- Organizing meeting between end-user and entrepreneur.
- Detail view of all the services that will provided in this platform.

2.5 Other Non-functional Requirement

2.5.1 Performance Requirements

The performance of the application depends upon the android version of mobile. Fast internet connection will be requiring for better performance of the application.

2.5.2 Safety Requirements

To ensure that no one of Architects Zone's users loses any data while using the app due to a crash or a bug of some kind the developer team updates the app regularly. There is a bug tracker available where users can report any bugs they have encountered so that the developers can fix it in the next release.

2.5.3 Security Requirements

The main security concern is for users account therefore right login mechanism must be used to avoid hacking. The mobile id registration is much to spam check for increasing the safety. Hence, security is provided from unwanted use of recognition software.

2.5.4 Software Quality Attributes

Some to consider are availability,

- Correctness – our project will provide correctness to the client and administrator i.e. it will be free from errors.
- Flexibility – more modules can be added so it means it provide flexibility.
- Interoperability – different modules of system can easily exchange information with one another.
- Maintainability – to keep the system preserve
- Reliability – the project is fully reliable.
- Robustness – the condition of the project is good to be used.
- Testability – different modules of the system are easily testable.
- Usability – easy to use without any complexity.

CHAPTER 3

DESIGN AND METHODOLOGY

3.1 Introduction

Third deliverable is all about the system designing and use case modelling and different diagram. In previous deliverable, include the analysis of the system. Therefore, now in this deliverable we design use case diagram and other diagrams in which we describe our project. What is more, we comprehend the current circumstance of the difficult space. Presently we prepared for take a stab at an answer for the issue plan by utilizing object-oriented methodology. Following artefacts must be included in this deliverable.

1. Use Case Description
2. Use Case Diagram
3. Domain Model
4. Sequence Diagram
5. Collaboration Diagram
6. Operation Contracts
7. Design Class Diagram
8. Data Model

Now we discuss these artefacts one by one as follows

3.2 User Description

In use case description, we portray the point-by-point usefulness of a use case. The description of the use-case is following:

3.2.1 Registration (U1)

There are many parts of the use case description:

Table 3-1 Registration (U1)

Name and ID	Sign Up (U1)
Objective	Customer and Architect can register themselves to use the application.
Priority	High.
Actor	Customer and Architecture.
Preconditions	User must open the application.
Basic flow or Happy path	User will simply give his/her credentials to sign up on App.
Trigger	User wants to view services.
Alternate flows	User can simply search the application but would not register himself.
Post conditions	User can get the access to view available services.
Issues	User type incorrect credentials.

3.2.2 Admin Verify User Account (U2)

There are many parts of the use case description:

Table 3-2 Admin Verify User Account (U2)

Name and ID	Admin Verify User Account (U2)
Objective	After registration, user account will verified by the admin so user can work further.
Priority	High.
Actor	Admin.
Preconditions	User must Sign up.

Basic flow or Happy path	Admin can active and inactive user account.
Trigger	Admins accept or reject the Request of user.
Alternate flows	Architect is not able to provide the service.
Post conditions	User can login.
Issues	No issue.

3.2.3 Login (U3)

There are many parts of the use case description:

Table 3-3 Login (U3)

Name and ID	Login (U3)
Objective	User login to the application to avail the services.
Priority	High.
Actor	Customer and Architect.
Preconditions	User should have signed up into application.
Basic flow or Happy path	User can login to application by giving required credentials.
Trigger	User wants to view services.
Alternate flows	User can simply search the application but would not register himself.
Post conditions	User can get the access to view the available services.
Issues	User give wrong credentials.

3.2.4 Architect Add Details (U4)

There are many parts of the use case description:

Table 3-4 Architect Add Details (U4)

Name and ID	Architect Add Details (U4)
Objective	Architect can add his/her work to his/her account.
Priority	High.
Actor	Architect.
Preconditions	Architect should have login to his/her account and his/her account should be verified by admin.
Basic flow or Happy path	Architect will login to his/her account and add map of his/her work.
Trigger	Architect must provide the service.
Alternate flows	Architect did not add his/her work.
Post conditions	Architect will go to the customer and perform the required service.
Issues	No issue.

3.2.5 Customer Select Service (U5)

There are many parts of the use case description:

Table 3-5 Customer Select Service (U5)

Name and ID	Customer Select Service (U5)
Objective	Customer can see the maps of architects and then choose architect according to his/her requirements.
Priority	Medium.
Actor	Customer.
Preconditions	Customer should have login to the application.

Basic flow or Happy path	All the available services and works of Architects are given to Application. Customer just must choose the required service by clicking on the service.
Trigger	Customer wants to use our services.
Alternate flows	Customer can simply see the services and works of Architectures but would not select any service.
Post conditions	Customer will select the architect of his choice.
Issues	No issue.

3.2.6 Logout (U6)

There are many parts of the use case description:

Table 3-6 Logout (U6)

Name and ID	Logout (U6)
Objective	Customer and architect can logout themselves from App after signing in.
Priority	Medium.
Actor	Customer and Architect.
Preconditions	A user must Sign In first to logout.
Basic flow or Happy path	Users will logout by clicking the logout button.
Trigger	User wants to logout from App.
Alternate flows	User did not want to logout after sign In.
Post conditions	No post condition as it is the last step.
Issues	No issue.

3.3 Use Case Diagram

A use case diagram is a dynamic or behaviour diagram in UML. Use case diagrams model the functionality of a system using actors and use cases. Use cases are a set of actions, services, and functions that the system needs to perform.

3.3.1 Main Use Case Diagram

In Figure 3-1 Main System Use Case is shown below.

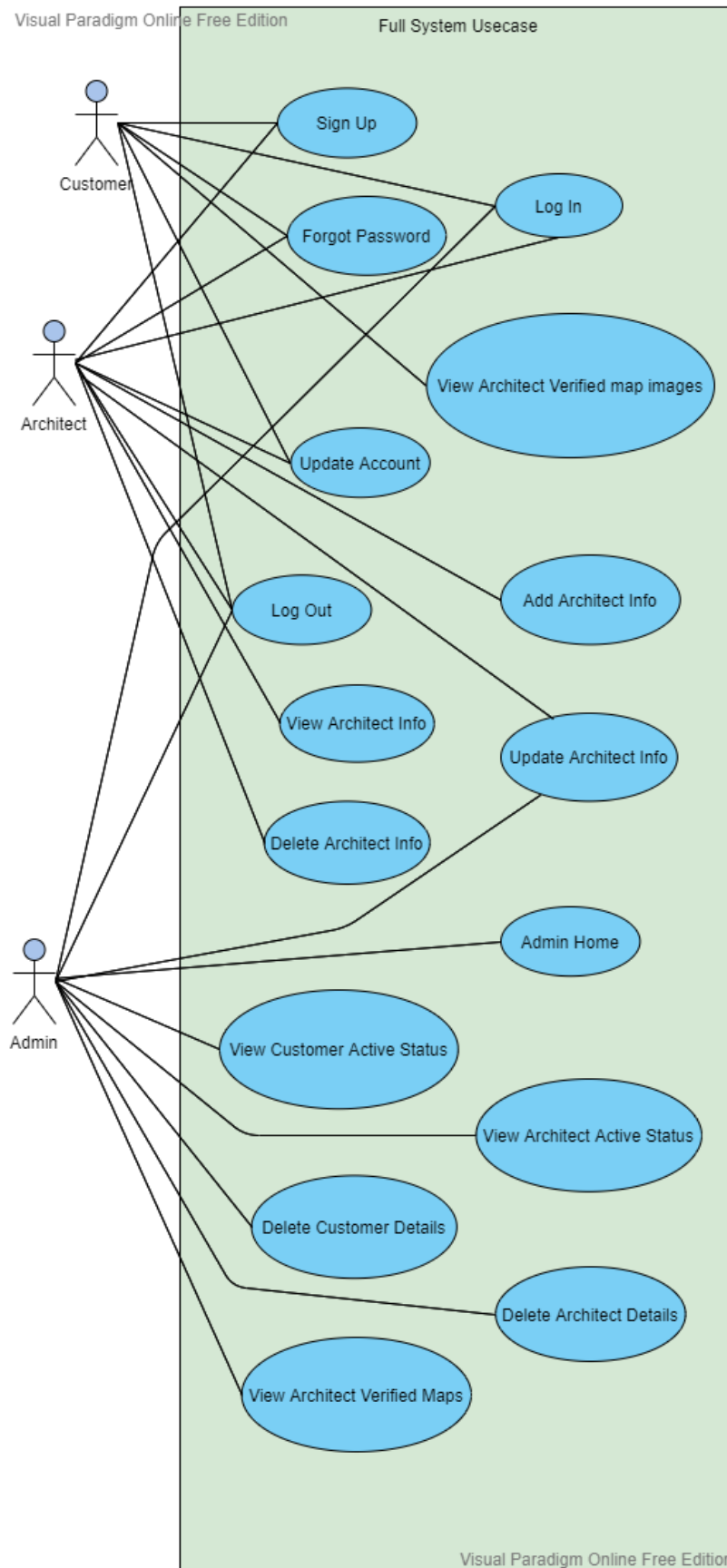


Figure 3-1 Main System Use Case Diagram

3.4 Domain Model

A domain is a collection of related concepts, relationships, and workflows. Domain modelling is a technique used to understand the project problem description and to translate the requirements of that project into software components of a solution. A domain model contains conceptual classes, associations between conceptual classes, and attributes of a conceptual class. The domain model of our project **Architect Zone** is shown below in Figure 3-2:

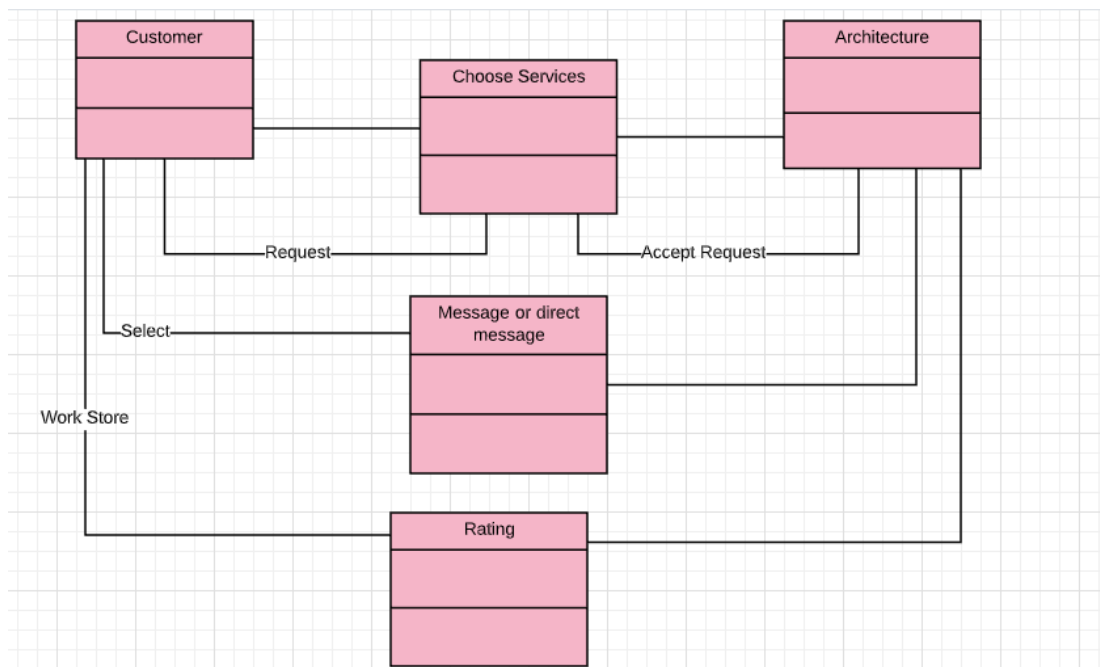


Figure 3-2 Main System Domain Model Diagram

3.5 Sequence Diagram

Sequence diagram shows object interaction in the arranged time. A sequence diagram is a good way to visualize and validate various runtime scenarios. These can help to predict how a system will behave and to discover responsibilities a class may need to have in the process of modelling a new system.

3.5.1 Registration (Users)

Figure 3-3 shows the Sequence Diagram of Registration of User.

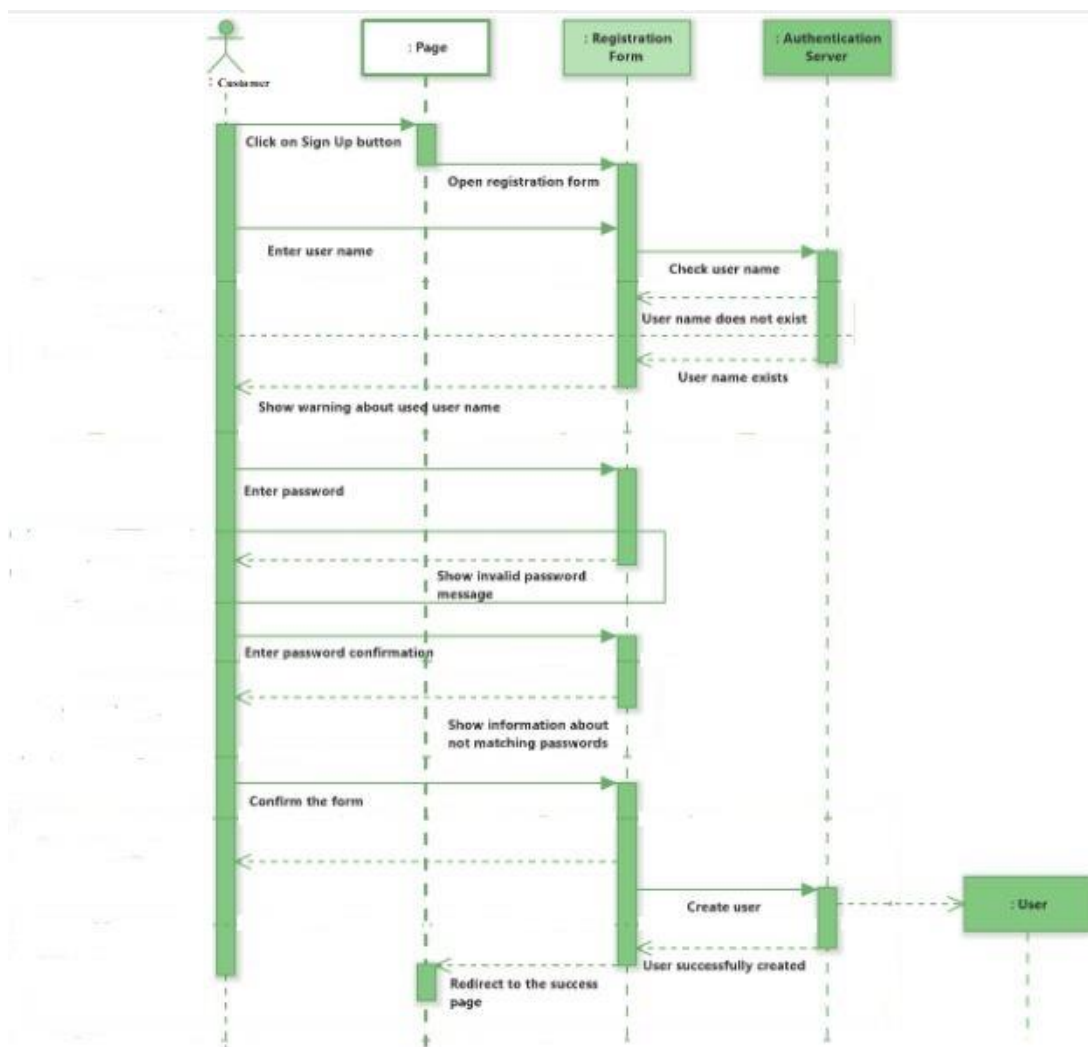


Figure 3-3 Sequence Diagram 1

3.5.2 Login (Users)

Figure 3-4 shows the Sequence Diagram of Login of User

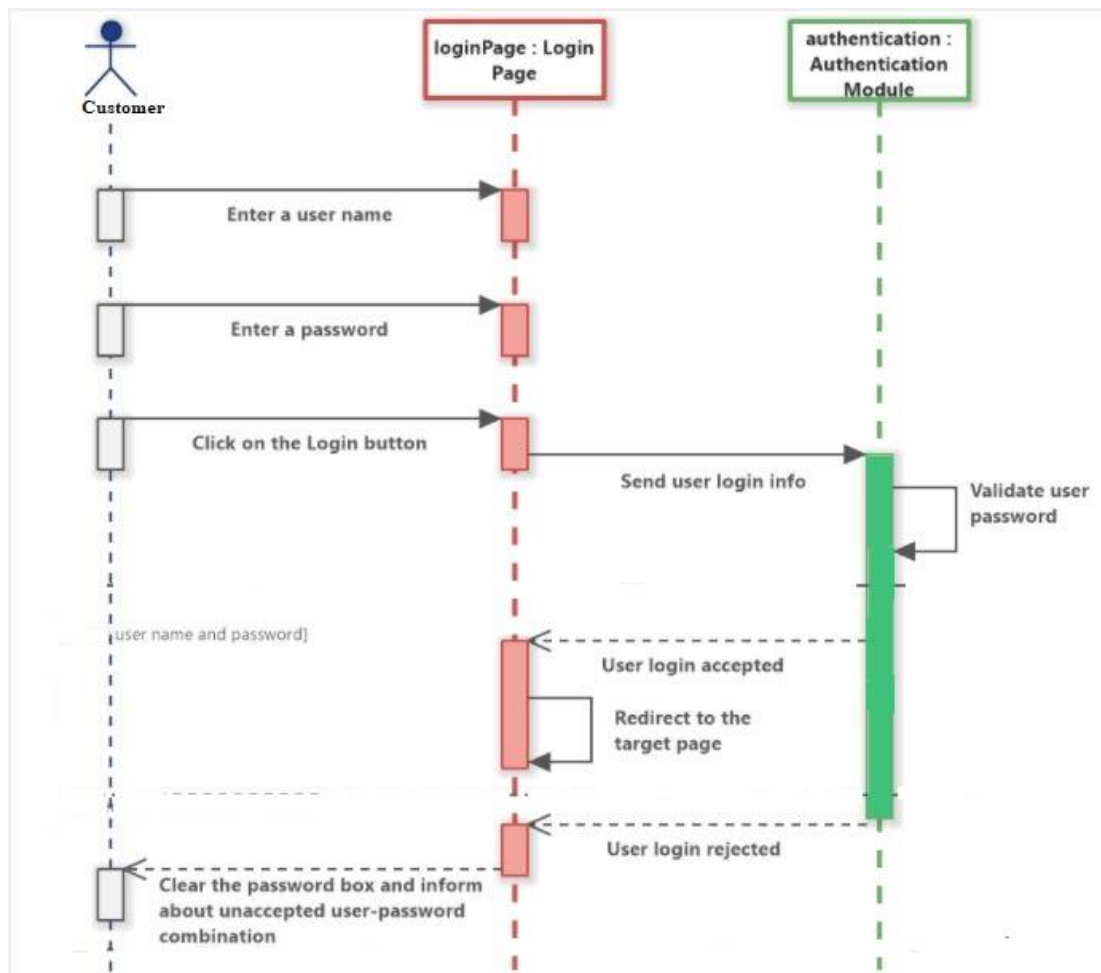


Figure 3-4 Sequence Diagram 2

3.5.3 Registration (Architect)

Figure 3-5 shows the Sequence Diagram of Registration of Architect.

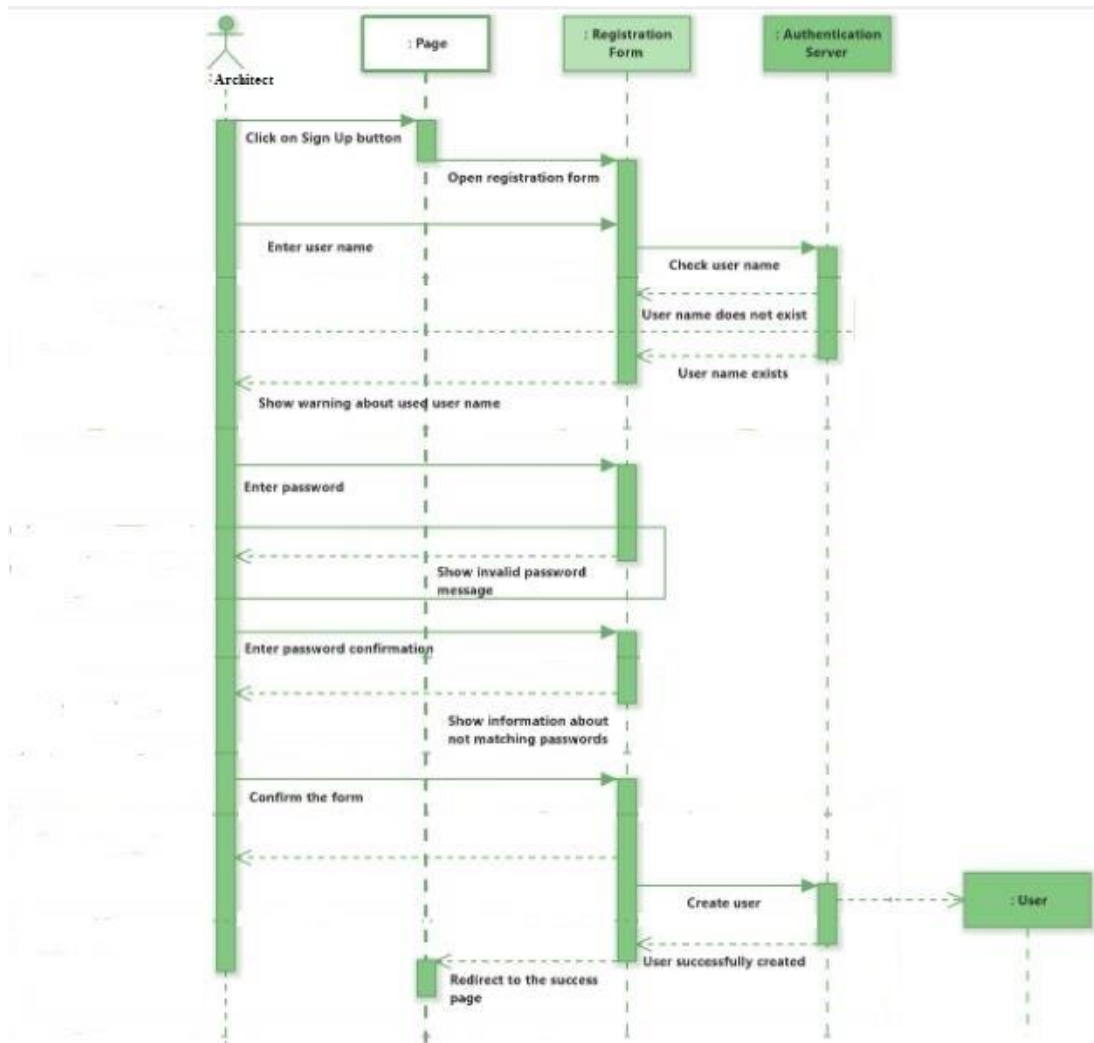


Figure 3-5 Sequence Diagram 3

3.5.4 Login (Architect)

Figure 3-6 shows the Sequence Diagram of Login of Architect.

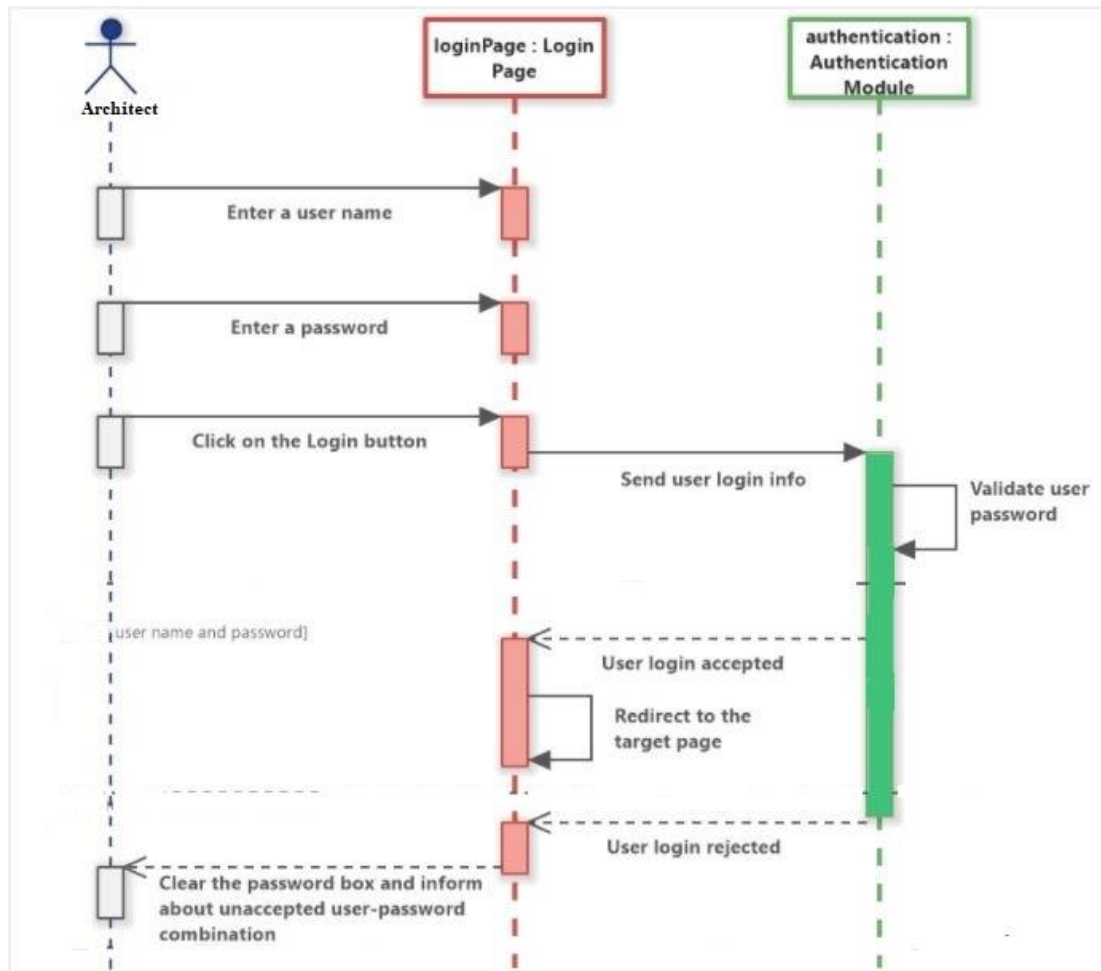


Figure 3-6 Sequence Diagram 4

3.5.5 Details for Architect

Figure 3-7 shows the Sequence Diagram of Details for Architect.

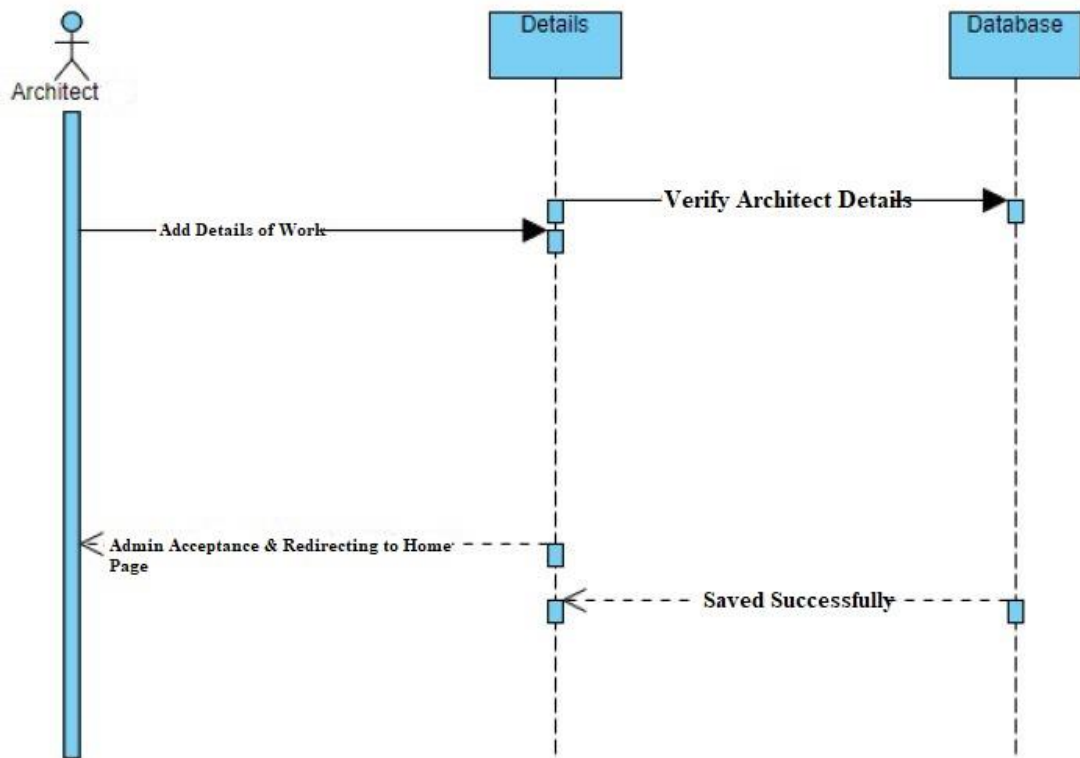


Figure 3-7 Sequence Diagram 5

3.5.6 Services for Customer/Users

Figure 3-8 shows the Sequence Diagram of Services for Customer/Users.

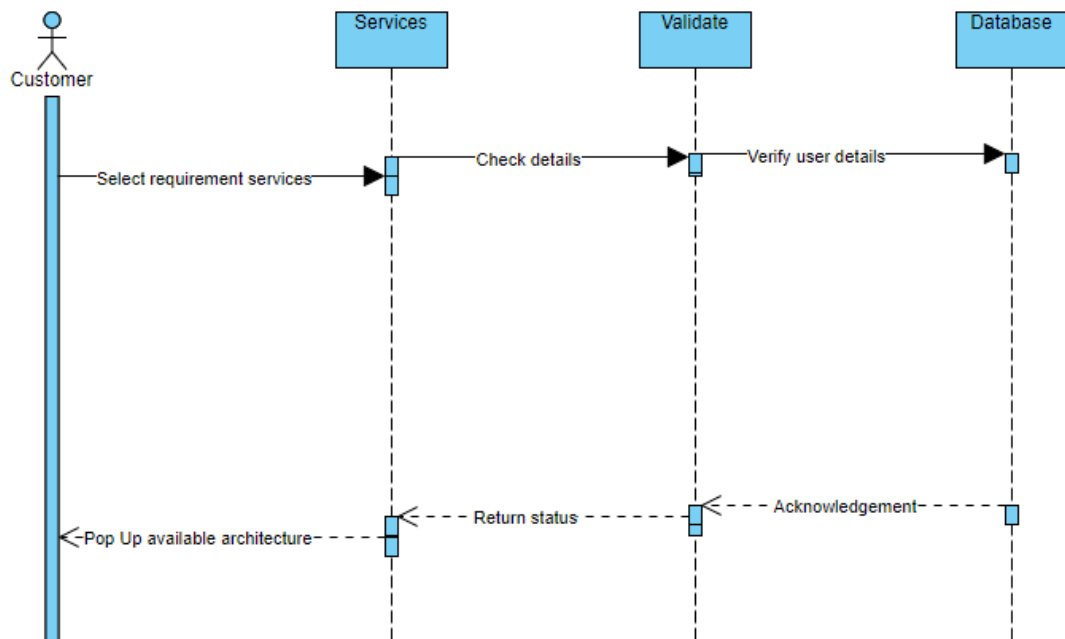


Figure 3-8 Sequence Diagram 6

3.5.7 Update Details

Figure 3-9 shows the Sequence Diagram of Update Details

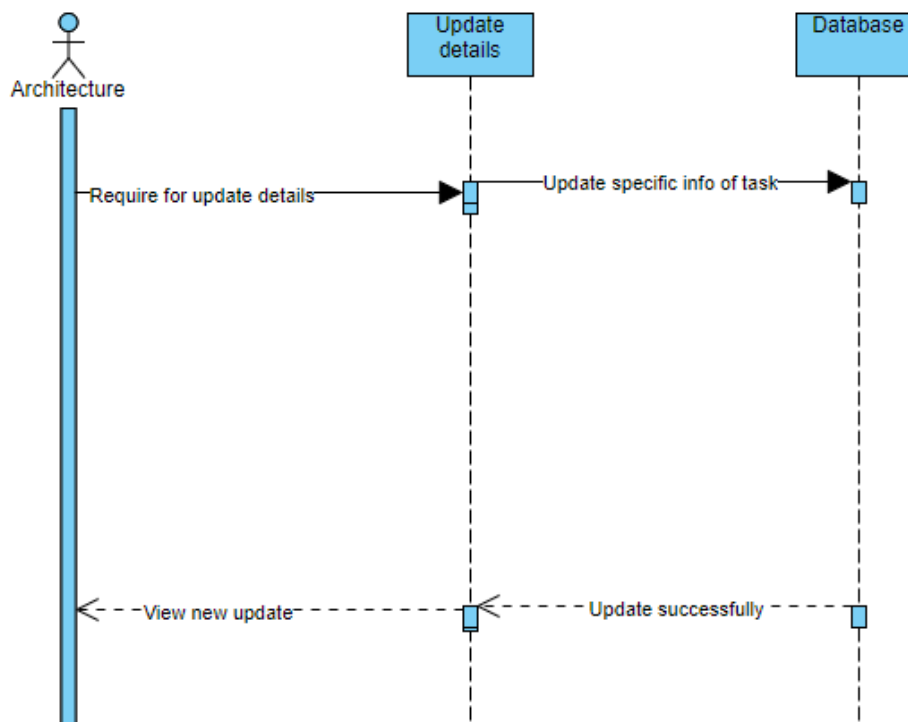


Figure 3-9 Sequence Diagram 7

3.6 Operation Contracts

Operation contracts describe preconditions and post conditions changes to the object in the Domain Model because of system operation. They are text-based documentation. Operation contracts are following:

3.6.1 Registration

Table 3-7 Registration

Name	Registration
Responsibilities	User will register themselves to the application.

Cross Reference	U1
Expectation	No
Precondition	Customers have the source to register like internet connection and have the access to application.
Post condition	User can get the access to view our available services.

3.6.2 Admin Verify User Account

Table 3-8 Admin Verify User Account

Name	Admin accept request of user.
Responsibilities	Admin can check the details of Architect and Customer and should accept or reject. Without real identification papers admin can reject the request of user.
Cross Reference	U5
Expectations	No
Precondition	User must have added the details.
Post condition	After selection of architect by customer, both will meet through admin.

3.6.3 Login

Table 3-9 Login

Name	Login
Responsibilities	User will login to our Application.

Cross Reference	U2
Expectation	No
Precondition	User must sign up to App by providing required credentials.
Post condition	Users have access to use any available services

3.6.4 Architects Add Details

Table 3-10 Architects Add Details

Name	Architects add details
Responsibilities	Architect should add his details related to his work with professionals' certificates. The architect will accept or reject the request of the customer.
Cross Reference	U4
Expectation	No
Precondition	Customer must select any service.
Post condition	Architect will perform the service.

3.6.5 Customers Select Service

Table 3-11 Customer Select Service

Name	Customers select service
------	--------------------------

Responsibilities	The Customer will select the service or architect according to their needs and desire.
Cross Reference	U3
Expectation	No
Precondition	Customer should log in to use the service.
Post condition	Customer can request the service, which is required.

3.6.6 Logout

Table 3-12 Logout

Name	Logout
Responsibilities	Users can logout from our App.
Cross Reference	U8
Expectations	No
Precondition	User must login to App
Post condition	Successfully logged out after using the App.

3.7 Design Class Diagram

Class diagram describes the attributes and operations of a class and the constraints imposed on the system. The class diagrams are widely used in the modelling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages. Class diagram shows a collection of classes,

interfaces, associations, collaborations, and constraints. It is also known as a structural diagram. Figure 3-11 shows the Main System Design Class Diagram.

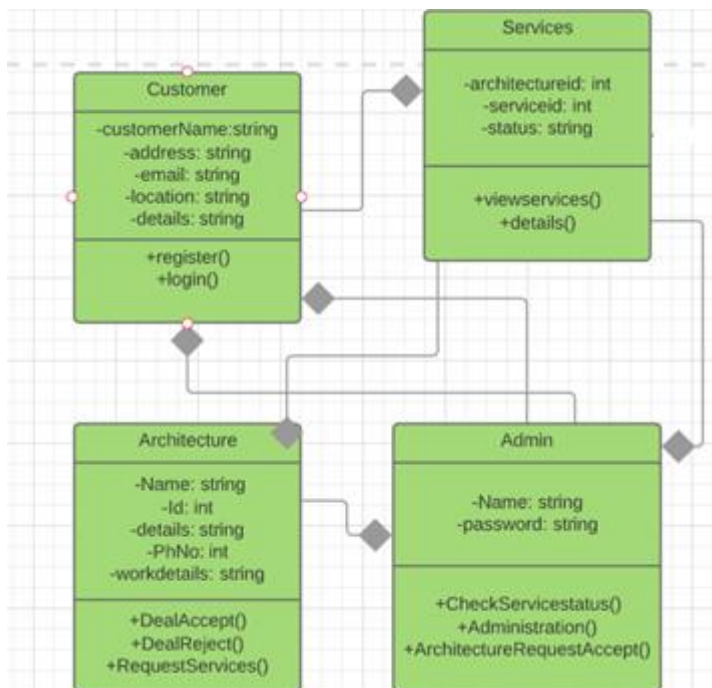


Figure 3-10 Main System Design Class Diagram

3.8 Data Model

The Data Model is defined as an abstract model that organizes data description, data semantics, and consistency constraints of data. The data model emphasizes on what data is needed and how it should be organized instead of what operations will be performed on data. Data Model is like an architect's building plan, which helps to build conceptual models and set a relationship between data items.

3.8.1 Entity Relationship Diagram

An entity–relationship model describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types and specifies relationships that can exist between entities.

Figure 3-12 shows the Main System Entity Relationship Diagram

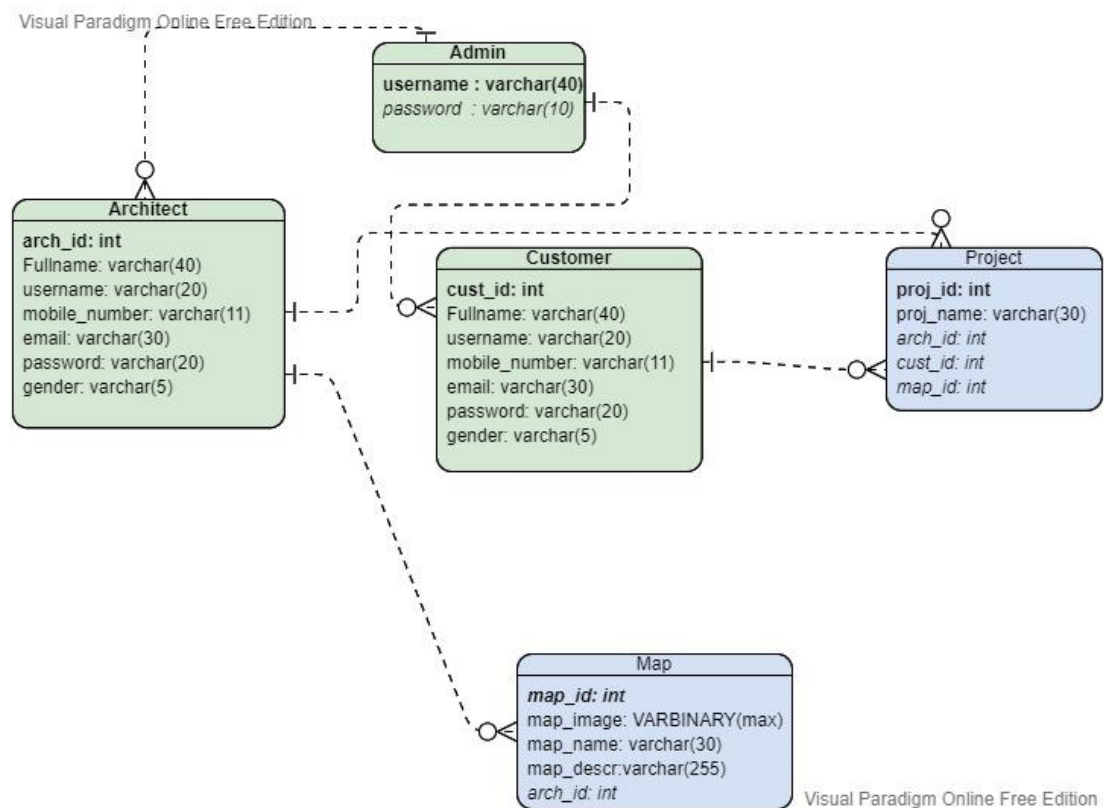


Figure 3-11 Main System Entity Relationship Diagram

CHAPTER 4

IMPLEMENTATION

Tool

Android Studio

Language

JAVA and XML

4.1 Splash Screen



Figure 4-1 Splash Screen

4.2 Admin Panel

This is Admin Panel for Admin Sign-in. After Sign-in, admin can view Architects and Customers activity.

4.2.1 Admin Sign-in

This is Admin Sign-in Page.



Architect Zone
FROM IMAGINATION TO REAL

Admin

Log In

Email:

Password:



[Forget Password](#)

Log In

[Create New Account](#)

Figure 4-2 Admin Sign-in

4.2.2 Admin Dashboard

This is Admin Dashboard Page.

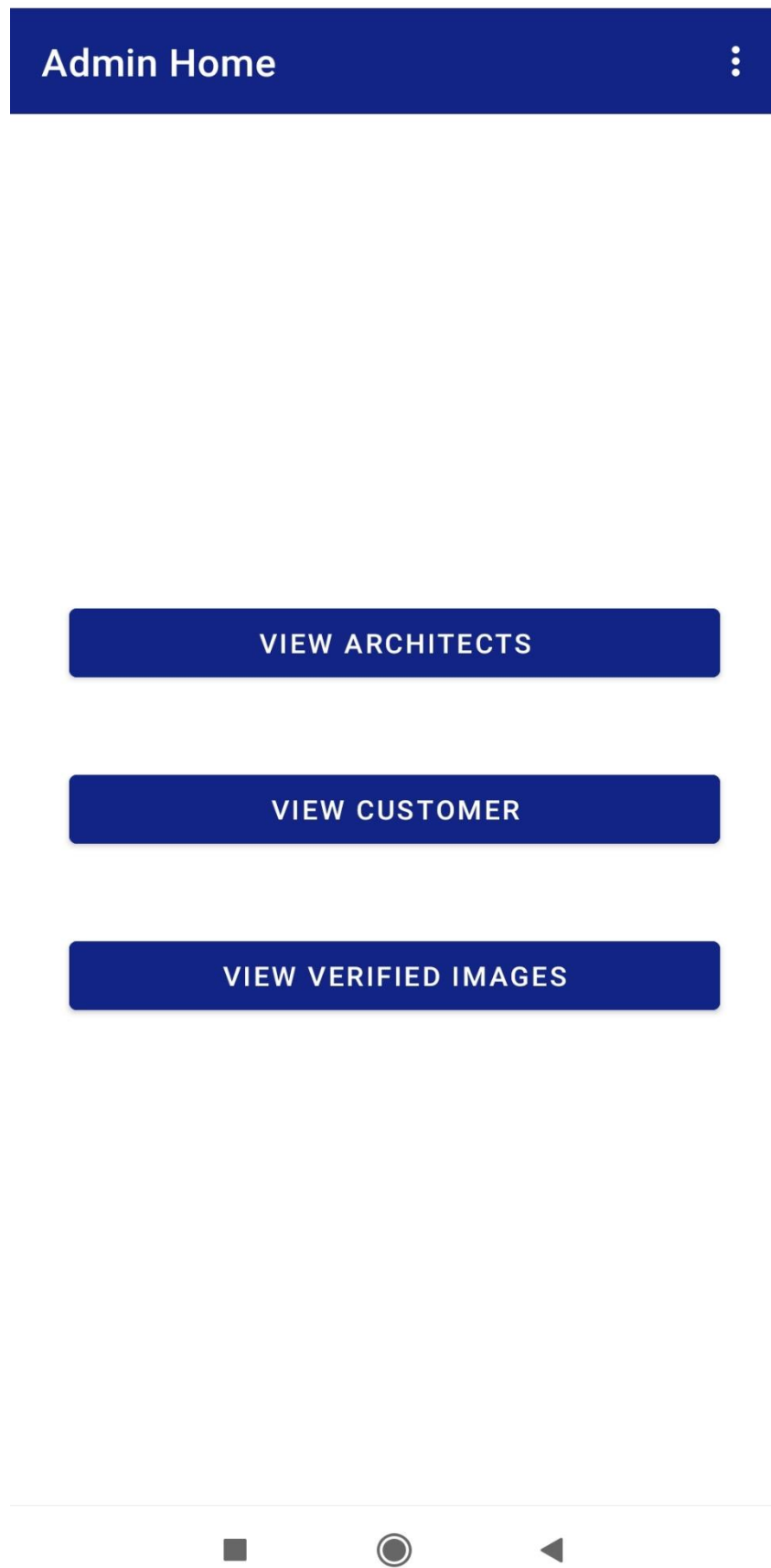


Figure 4-3 Admin Dashboard

4.2.3 Admin View Customer

Admin View Customer Activity.

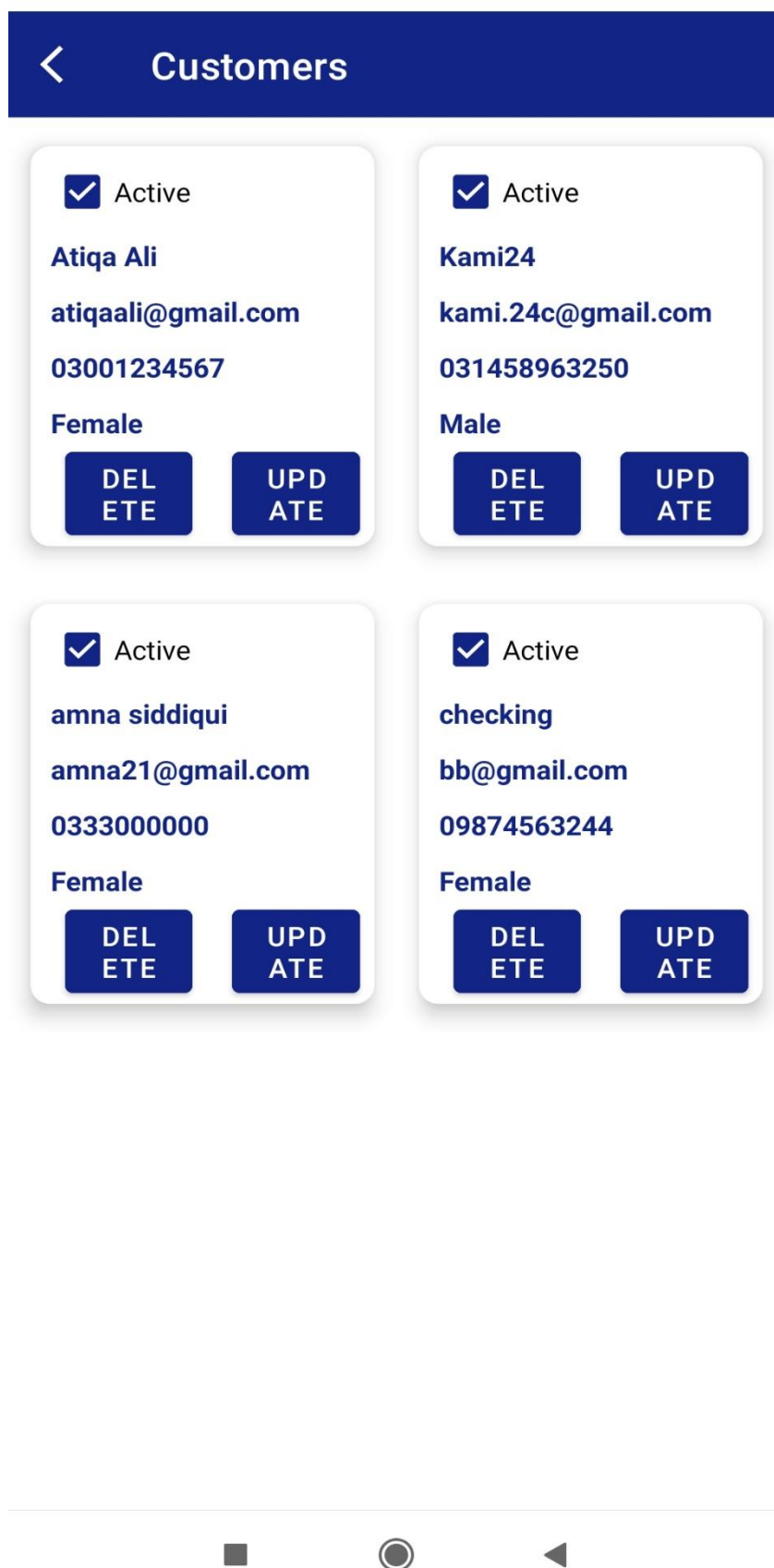


Figure 4-4 Admin View Customer

4.2.4 Admin Verified Images

Admin view Verified images activity.

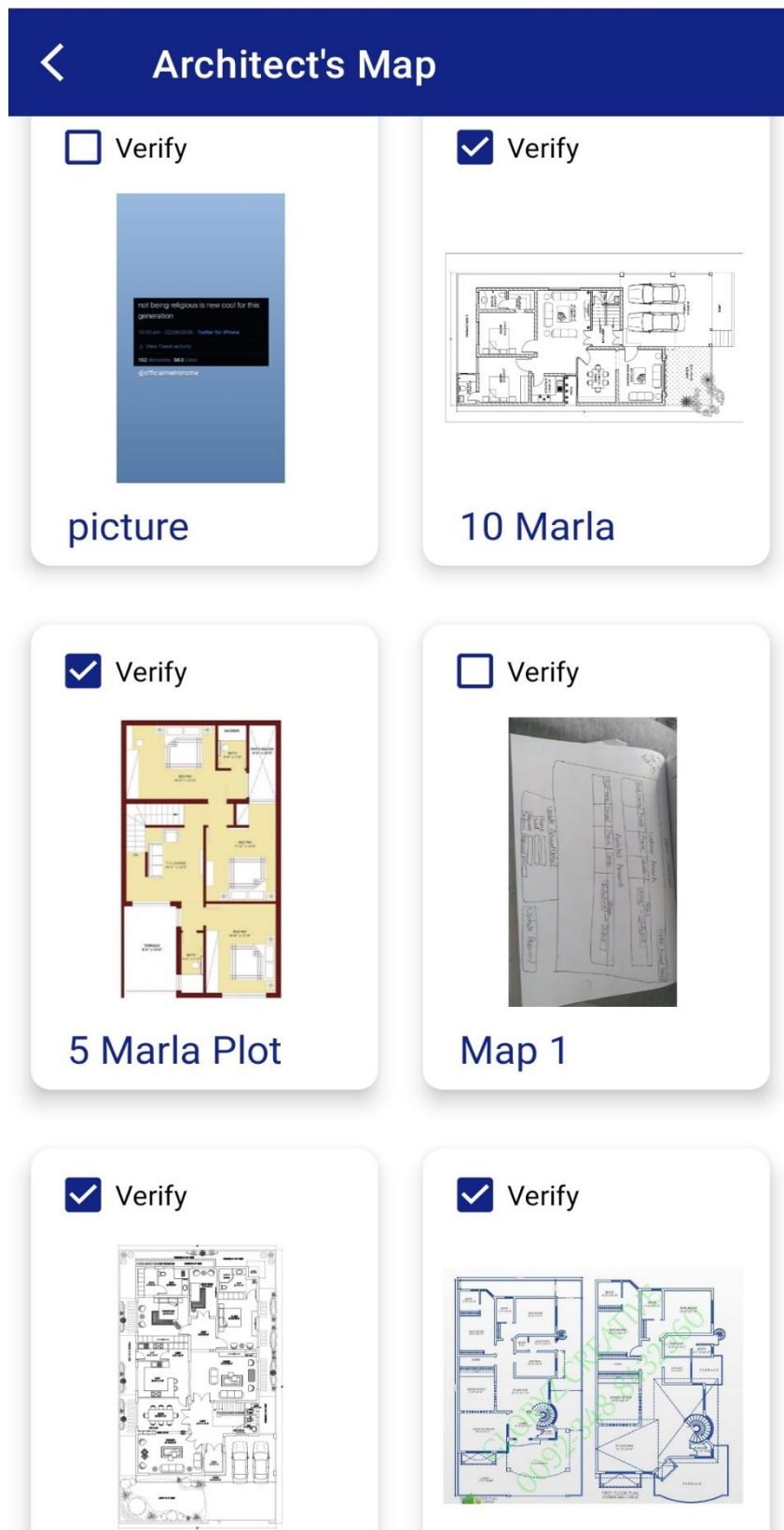


Figure 4-5 Verified Images

4.2.5 Admin Logout

Admin Logout Activity.

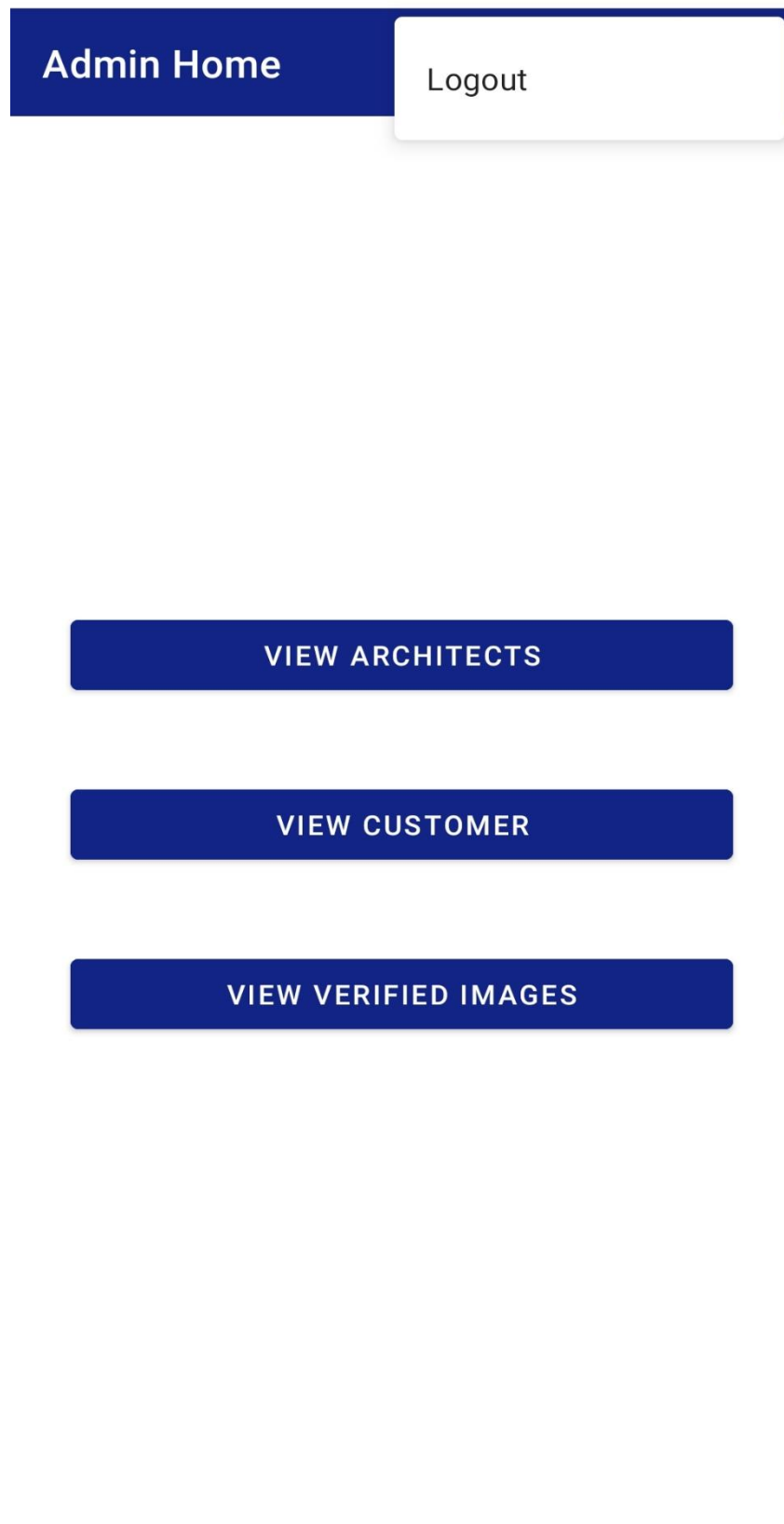


Figure 4-6 Admin Logout

4.3 Architect Panel

Architect Panel for architect create new account, login and other activities.

4.3.1 Admin Sign-up

< Sign Up


Architect


Full Name:

User Name:

Mobile Number:

Email:

Password: 

Confirm Password: 

Gender

Male Female

Sign Up




Figure 4-7 Architect Sign-up

4.3.2 Architect Login



Architect

Log In

Email:
rubab4601@gmail.com

Password:
..... 

[Forget Password](#)

Log In

[Create New Account](#)



Figure 4-8 Architect Login

4.3.3 Architect Dashboard

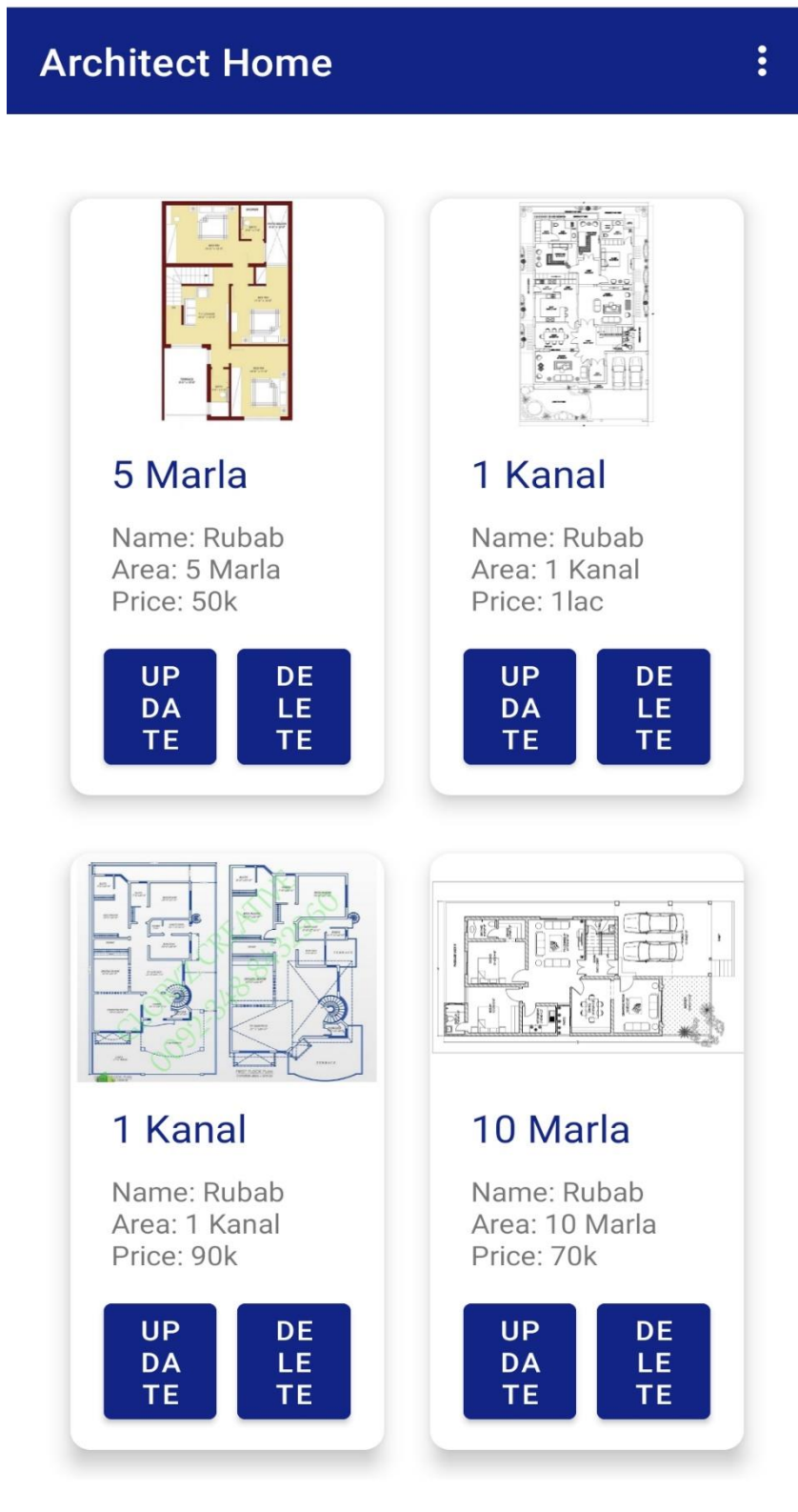


Figure 4-9 Architect Dashboard

4.3.4 Architect Add Data

< Add Map

Map Name:

Map Image: **CHOOSE FILE**

Map Image: **CHOOSE FILE**

Map Description:

INSERT

■ ● ◀

Figure 4-10 Add Data

4.3.5 Architect Update Account

The screenshot shows a mobile application interface for updating an architect's account. At the top, there is a dark blue header bar with a white back arrow on the left and the text "Architect Profile" in white. Below the header, there are four white input fields with rounded corners and dark blue borders. The first field is labeled "Mobile Number:" and contains the text "03319547851". The second field is labeled "Email:" and contains "rubab4601@gmail.com". The third field is labeled "Password:" and contains seven dots, with a blue eye icon on the right side. The fourth field is labeled "Confirm Password:" and also contains seven dots, with a blue eye icon on the right side. Below these fields is a dark blue button with the text "Update Account" in white. At the bottom of the screen, there is a white bar with three dark blue icons: a square, a circle, and a triangle pointing left.

Figure 4-11 Update Account

4.3.6 Architect Logout

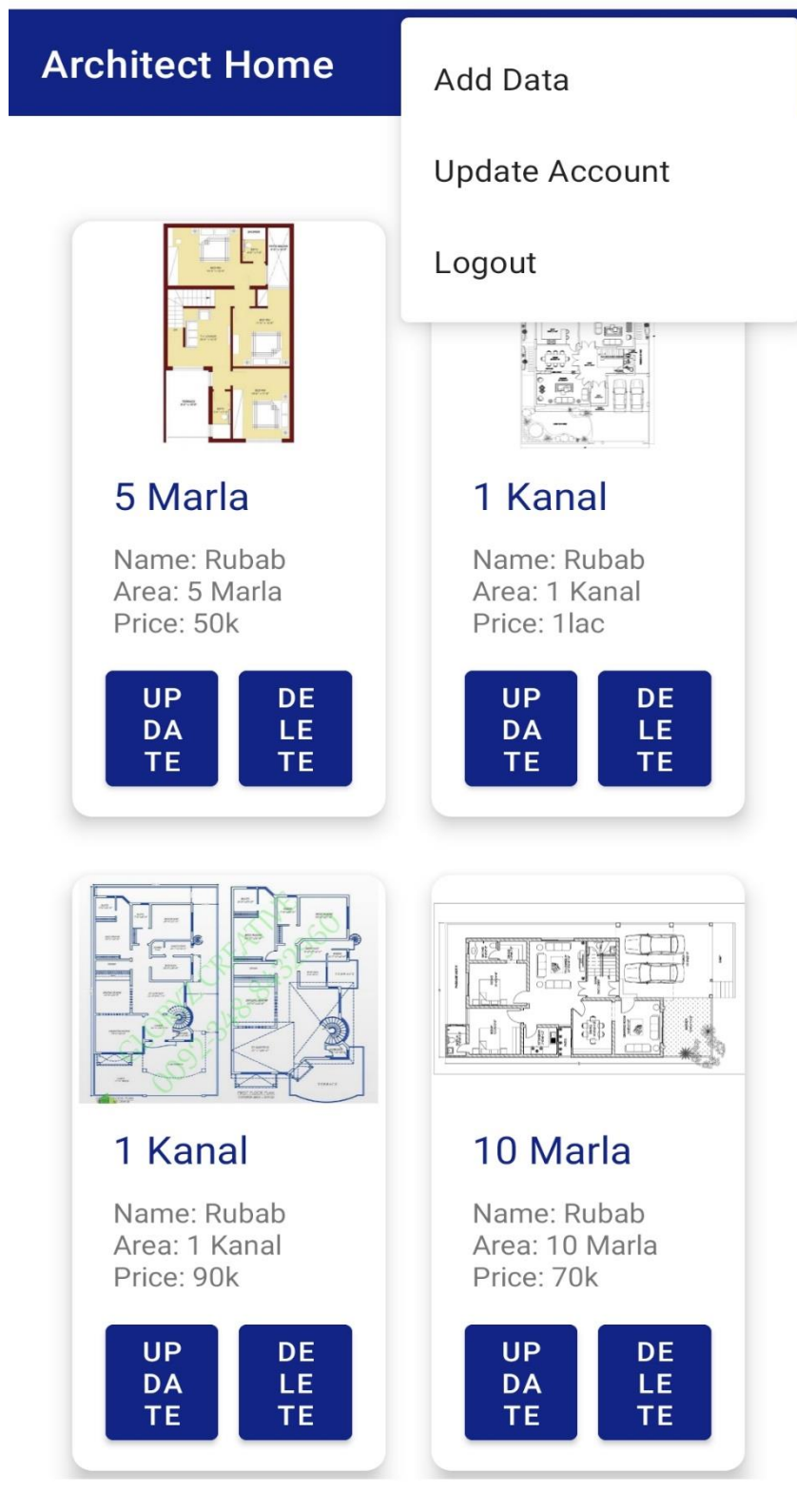
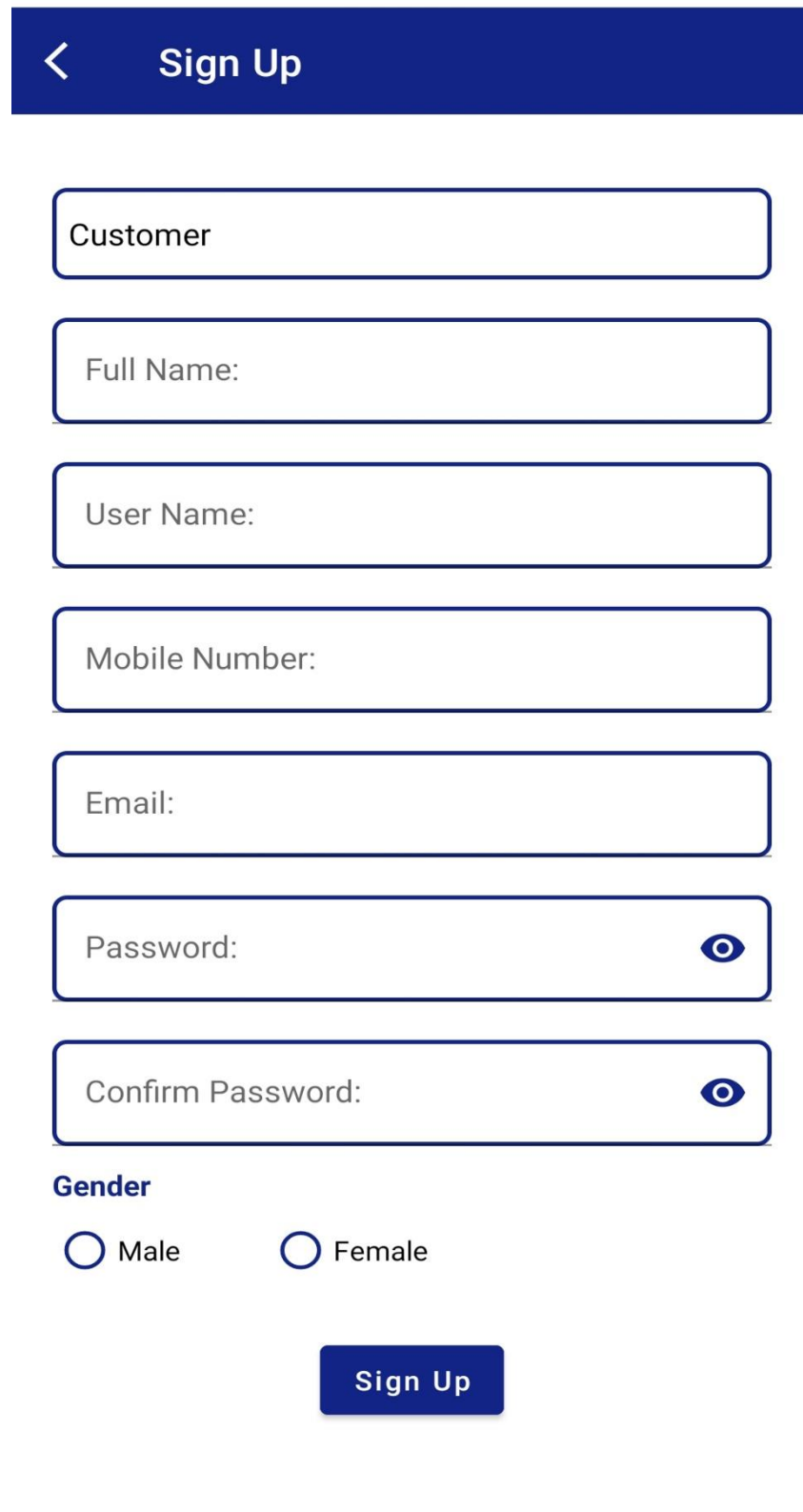


Figure 4-12 Architect Logout

4.4 Customer Panel

Customer Panel for customer create new account, login, view images after admin verify them.

4.4.1 Customer Sign-up



The image shows a mobile application interface for customer sign-up. At the top, there is a dark blue header bar with a white back arrow on the left and the text "Sign Up" in white. Below the header, there are seven input fields, each with a dark blue border and rounded corners. The first field contains the text "Customer". The second field is labeled "Full Name:". The third field is labeled "User Name:". The fourth field is labeled "Mobile Number:". The fifth field is labeled "Email:". The sixth field is labeled "Password:" and has a dark blue eye icon on the right side. The seventh field is labeled "Confirm Password:" and also has a dark blue eye icon on the right side. Below the input fields, there is a section titled "Gender" in bold. Underneath, there are two radio button options: "Male" and "Female". At the bottom of the form, there is a dark blue button with the text "Sign Up" in white. At the very bottom of the screen, there is a white bar with three dark grey icons: a square, a circle, and a triangle pointing left.

Figure 4-13 Customer Sign-up

4.4.2 Customer Login



Customer

Log In

Email:
amna21@gmail.com

Password:
.....



[Forget Password](#)

Log In

[Create New Account](#)



Figure 4-14 Customer Login

4.4.3 Customer View Images

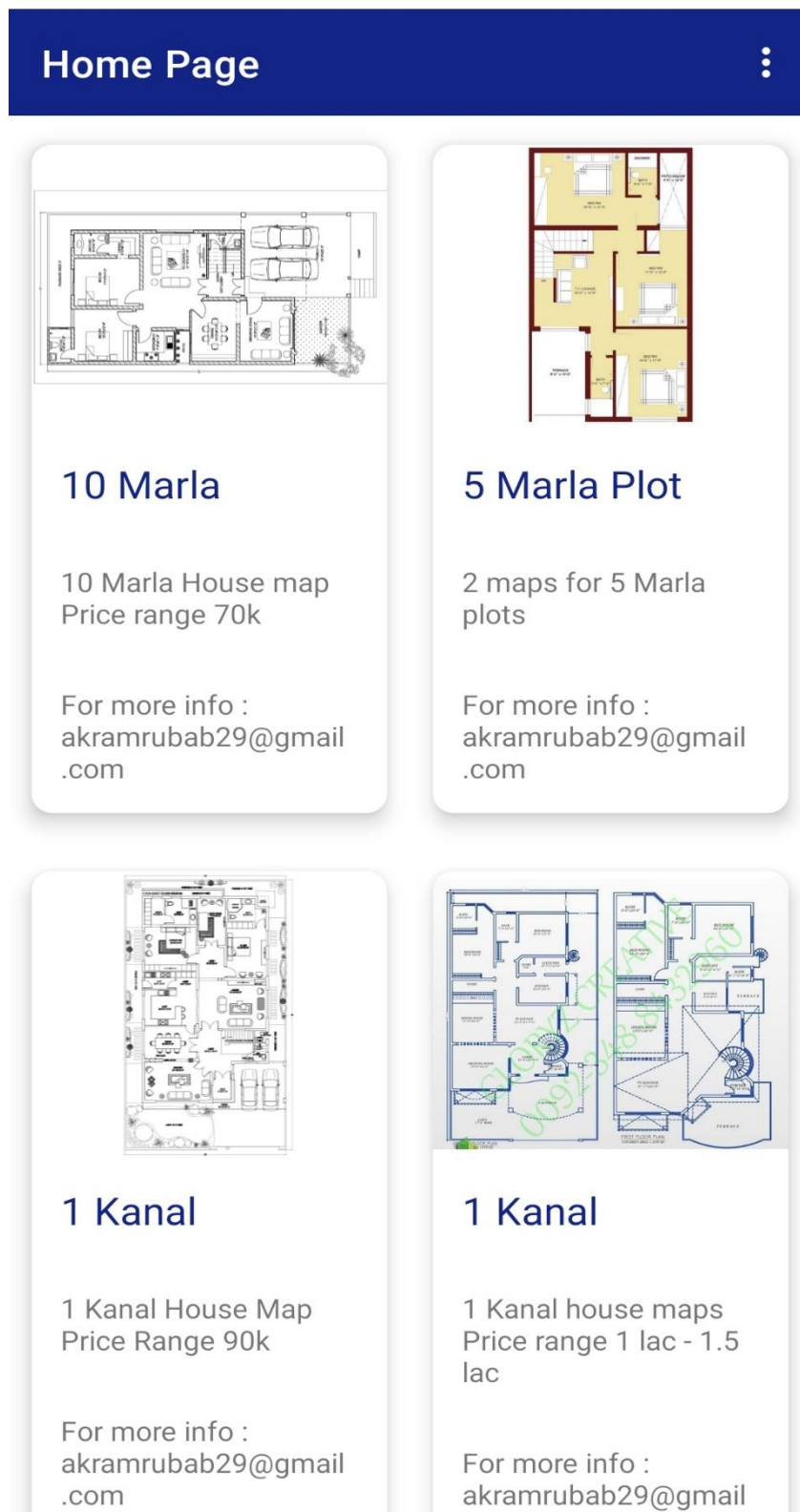


Figure 4-15 View Images

4.4.4 Customer Logout

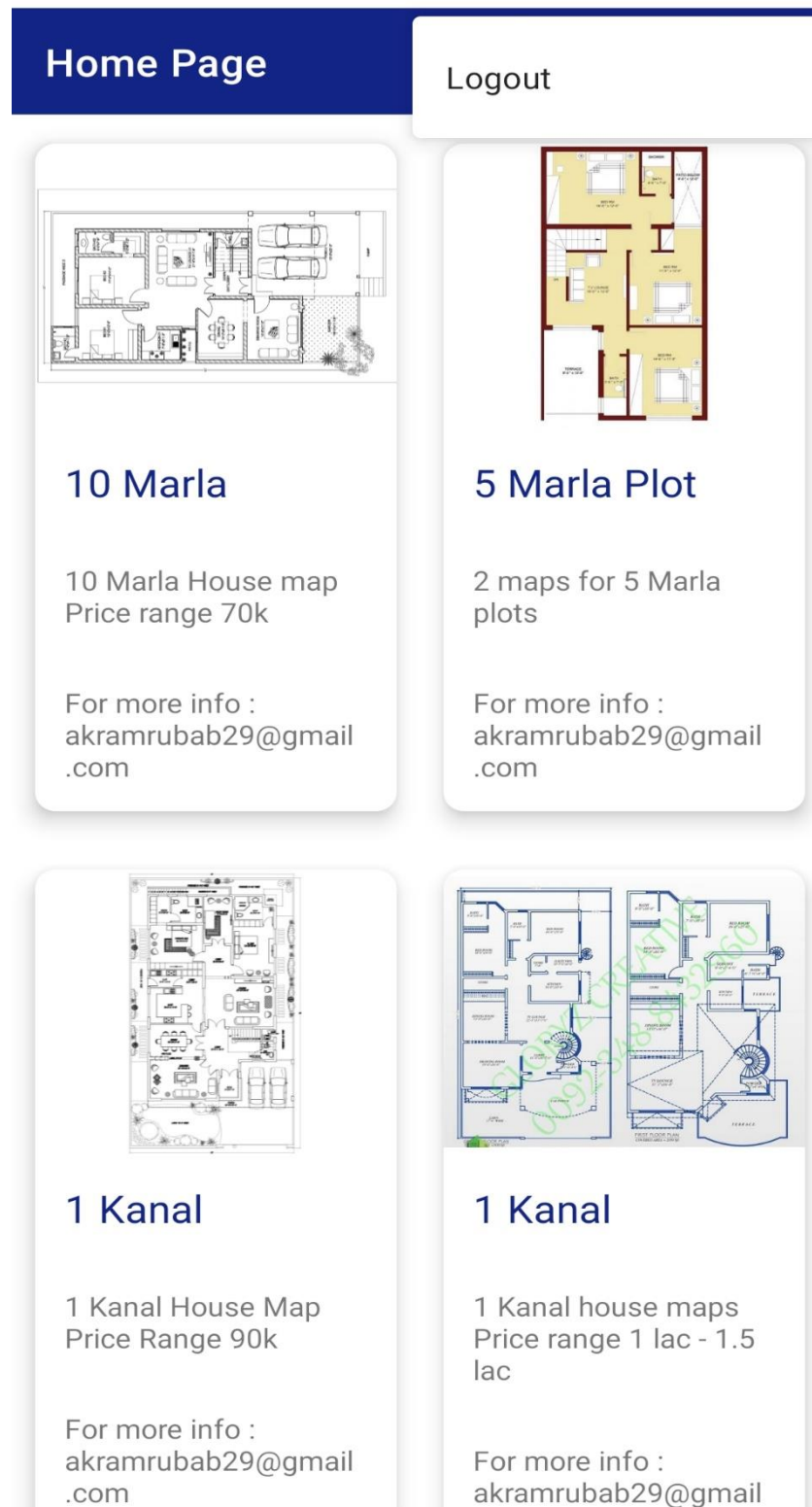


Figure 4-16 Customer Logout

CHAPTER 5

RESULTS AND DISCUSSIONS

We Architect Zone providing a third party services for our domain holders, which are Architects and Customers. They both can register themselves. Providing them a better solutions, better discussions and better designs in an efficient way and in a flexible way so that they can communicate easily in a better environment. To make the application interactive, different controls have been used and designed using the layout file. Although a less amount of related work has been done related to our concept, but there have been many flaws in those previous works.

CHAPTER 6

CONCLUSION

The conclusion of Architect Zone is to provide the compatible services for the domain holder's i.e Architect and Customer to interact with the friendly environment designs and providing the better functionalities that they easily cooperate their work and share their designs, ideas and as for the better progressive statements. We have learned a lot from this project on how to develop Android Application and publishing it in real time. As there is no such application already in market that provide an easy and hustle free interaction. Now, in post COVID-19 world, almost everything is getting online and people avoid meeting in public so if people use Architect Zone in future they have several advantages.