Muhammad Asad Alvi
01-134181-083
Muhammad Awais Khan
01-134181-084

Phortable

Bachelor of Science in Computer Science

Supervisor:
Dr. Kashif Naseer

Department of Computer Science
Bahria University,Islamabad

January, 2022

Acknowledgement

Abstract

Technology is playing a vital role in our daily life. Almost every field is adapting technology and shifting their system to online platforms. Like People and Organizations had created their profiles on LinkedIn to show their work and progress in industry. Phortable is a social media platform for our photographers community to create their portfolios. It is a mobile application which serves the needs of photographers and end-users by providing a platform for communication and notifications features in a user-friendly manner by utilizing the best UI/UX practices in the industry. Phortable is like Instagram but it is specially designed for photographers community, to give them a chance to show people their work and startup their career. The goal also is to make the usability and interactional capabilities of this application up to the mark and in competition with today's advanced software. To achieve this goal, we have spent days in research and gathered all the necessary data to make this project a success. We have also adopted the latest techniques to design and implement this application for maximum performance and optimization.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Technology is playing a vital role in our daily life. Almost every field is adapting technology and shifting their system to online platforms. Like People and Organizations had created their profiles on LinkedIn to show their work and progress in industry. Phortable is a social media platform for photographers in which they will share their content either publicly or privately through application which includes pictures, short videos etc. The basic idea is to provide Freelance photographers a platform to build up their portfolio by showcasing their content in front of big studios to get a chance to be a part of their ventures and attract end users for small paid events like wedding, birthday parties, casual shoots etc.

1.2 Problem Statement

In market, there are so many social media platforms like Facebook, Instagram, Twitter, Snap chat etc. Where people engage with each other to see what's going on in their friends life. Businesses use it to promote their brands by sharing their content to engage more users. But there is no such specific platform where photographer's community especially newcomers who don't have their names in market to showcase their talent and get opportunities to be working with big studios or getting small events. This platform provides opportunities for all Parties.

1. Newcomers who want to grab opportunities by working with big names or getting small events by building up their portfolios.

2. Big Studios who are looking for talented freelance photographers.

3. End users who are looking for good photographers for their events

1.3   Objective

To create a mobile application which serves the needs of photographers and end-users by providing a platform for communication and notifications features in a user-friendly manner by utilizing the best UI/UX practices in the industry.

- Provide a unified platform which should cater all problems stated in the problem statement:
  - Unified platform for all activities
  - Notification system so that photographers do not any order, and they stay informed regarding any upcoming event.
  - Earning management so that photographer can keep track of their earning.
  - Messaging system so that photographer and client can have an official communication platform

- Platform where photographers and clients can interact with each other

- Provide a user-friendly mobile application which focuses on UI and UX to maintain quality.

- Design and develop an application for photographers community to build up their portfolios.

1.4   Project Scope

1.4.1   Description

The project is to provide people with a platform where communication will be made possible with sharing their professional work in form of a mobile application. The theme of this application is to allow communication among:

- Studios

- Photographers

- Clients

The project aims to deliver a multi-platform mobile app with a single code-base able to run on both iOS and Android machines. Social media platforms for Photographers community to share their content and building up their portfolio.

1.4.2   Inclusion

- A system for clients and photographers to get hired or hire for any event

- A real time messaging service

- An API connected with a database to enable real time content sharing. Available on android from version 6 on wards

- A schedule management system for the photographers.

- An automated system to cater content theft.

- A server available to the company will be used to manage the application. Application will be deployed in Pakistan only.

- The application is bi modal.

- Profile rating system for all users

1.4.3 Exclusion

- English will be the primarily and only language for the application.

- The application does not have a web platform.

- The application is not personalized.

- There are no filters or categorizing options to choose a photographer.

- There is no calling feature enabled on this application

1.5 Solution Areas

This App can be used by Freelance Photographers and Studios to connect with each other. It will help Studios to hire Freelance photographers for their ventures on project base contracts, end user to hire Studios or Freelance photographers for their events and Freelance Studio to build up their portfolios.

Chapter 2

Literature Review

2.1 Existing Research

It is significant to understand and recognize the information about existing systems to provide a better system compared to already available systems. This is quite helpful to trace a gap between already available systems and the proposed system. For this project, we have thoroughly analyzed multiple existing solutions and have concluded that there is no application on the market that successfully satisfies the scope of the proposed system.

2.2 Existing Systems

2.2.1 Facebook and Messenger
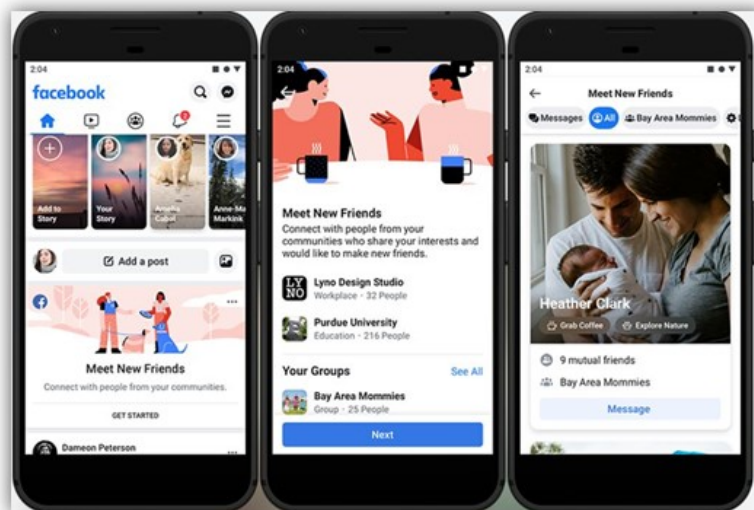
Figure 2.1: Facebook Interface

Facebook is a highly used social media platform since last 8 years which facilitate people to communicate with each other and can see what others are doing in their life. Messenger has been used alongside with Facebook. It is a platform where people can chat with each other and share their pictures and videos with their friends in the chat box. The Facebook and Messenger front-end is built in PHP, Erlang is used for chat, JAVA and C++ are used

in several places (like other languages etc.). Both Facebook and messenger are designed for all kind of people to communicate and get everyone notified about what's going in one's life. But it does not provide a platform specifically for photographers where they can provide their work and get more opportunities.
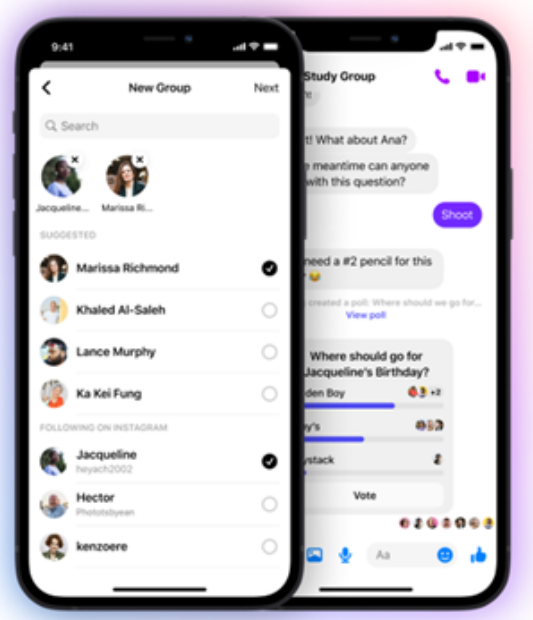
Figure 2.2: Messenger

2.2.2 Instagram

Instagram is a social media platform where people share their photos and videos. Also, it allows people to communicate to each other through Direct Message (Chat box).Instagram is generally used by amateur content creators to highlight and showcase their work.
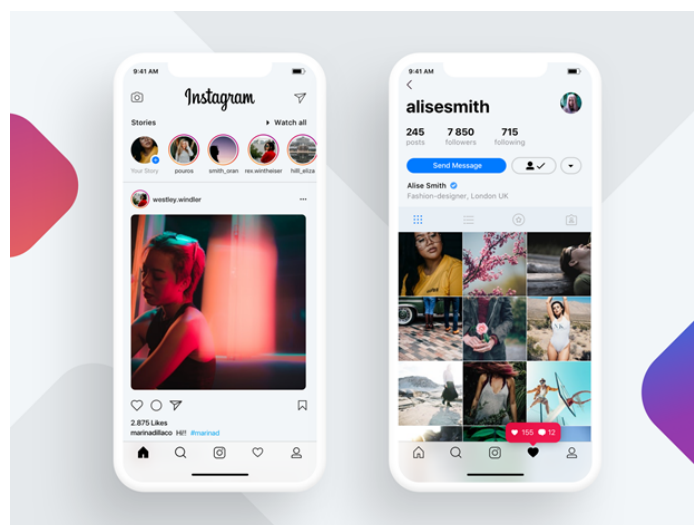
Figure 2.3: Instagram Interface

2.3   Importance of Mobile-First Design

During the last decade, mobile phones have become an integral part of our lives. More
people browse the web on mobile phones than on desktop devices and this trend will keep
on increasing therefore it is crucial that apps and websites are designed for the mobile user.
Providing good mobile experience will result in more user engagement, more traffic and in
the end more profit. Web Apps that are not designed with the mobile user in mind tend to
fail often.One of the major elements in mobile first design is simplicity. Keeping the appli-
cation as simple as possible gives the best user experience. This does not mean the extra
functionality is not necessary, but it is where we must consider how the user is searching and
what their purpose may be. For example, a desktop user may be looking for more in-depth
information and expecting full-size imagery and more white space whereas the mobile user
just wants to be able to find the answer quickly.

2.4   Value of UI / UX in Mobile Applications

Mobile users prefer quick and easy to navigate applications that can solve their problems
with fewer interactions and touch instead of tacky applications that provide a lot of useless
features therefore it is crucial that the UI is designed in such a way that it considers the
users' needs first and foremost. This will enhance the UX of the application as all the users'
needs are being met with minimal lag or disturbance. Applications that fail to keep UI/UX
principles in mind fail to attract users and keep them long-term.

Chapter 3

Requirements Specifications

3.1 Existing System

For requirements specification, we used the existing system of Instagram. Instagram is designed for all kind of people. In Instagram, there is same interface for either organization, common people, or entrepreneurs.

3.2 Proposed System

The proposed system comes with all the features within one application while keeping privacy intact. The mobile application is proposed to be designed with the latest technologies and focuses on the UI/UX aspect which will improve the user experience and provide a user-friendly interface for its users. The system will keep the photographers and clients in communication, and it will focus on features like portfolio update, earning track, order track and notification handling.

3.3 Functional Requirements

3.3.1 Sign up and Login

Table 3.1: Signup & Login Functional Requirements

FRR01	User will first signup if using the application for the first time.
FRR02	Signup for both photographers and clients are separate.
FRL01	User should be able to log in using a valid username and Password
FRL02	User should be able to see an error message alert when wrong username or password is entered.
FRL03	User should see a forget password button on the login screen to reset password by entering their username.

3.3.2   Earnings

Table 3.2: Earning Functional Requirement

FRA01	Photographer can keep record of his/her earning

3.3.3   Events

Table 3.3: Event Functional Requirements

FRC01	Client can keep track of their event and which photographer they hired for that specific day and time.
FRC02	Photographer can keep track of upcoming events.
FRC03	Photographer has a record for being booked for an event.

3.3.4   View Profile

Table 3.4: View Profile Functional Requirements

FRC01	Clients and photographer can view, edit, and update their profile.
FRC02	Clients can view profile of photographers.
FRC03	Photographer can view other photographers and studio's portfolio.

3.3.5   Chat

Table 3.5: Chat Functional Requirements

FRM01	All users should be able to send message to other users
FRM02	User should be able to receive and view a message sent to them

3.3.6   Notifications

Table 3.6: Notification Functional Requirements

FRN01	When Client has booked a photographer or studio for an event, a notification should be sent to both client and photographer.
FRN02	Client should get notification about new and high rated photographers.

3.3.7 Ratings

Table 3.7: Ratings Functional Requirements

FRCSP01	Client should have privilege to write a review and rate work of photographer.

3.4 Non-Functional Requirements

3.4.1 Performance

The performance of the application will be evaluated by its response time for the given task. For example, whenever a user launches the application, it should show the initial screen within 3 seconds.

3.4.2 Responsiveness

Application should be responsive to any activity by user or other external interruption, and it should return to the same state after the interruption. For example, if a user gets a phone call while using the application, the application should save state and resume from there once the call ends.

3.4.3 Security

All the application data should be encrypted and secure. The data should be protected from internal and external attacks. For example, the authentication token should be stored on local device for comparison purpose, and it should be accessible with user's permission.

3.4.4 Usability

The system's user interface should be user-friendly, and the system should be easy-to-use. The user should be able to understand the flow of the application without any help or guideline.

3.4.5 Scalability

The system should be able to handle increased amount of usage and should be able to handle large data as the time progresses. For example, when the user data is increased (cache, stored data, etc.), the application should handle it without any delay and the storage of the data should be optimized.

3.5 Use Cases

The system should be able to handle increased amount of usage and should be able to handle large data as the time progresses. For example, when the user data is increased (cache, stored data, etc.), the application should handle it without any delay and the storage of the data should be optimized.
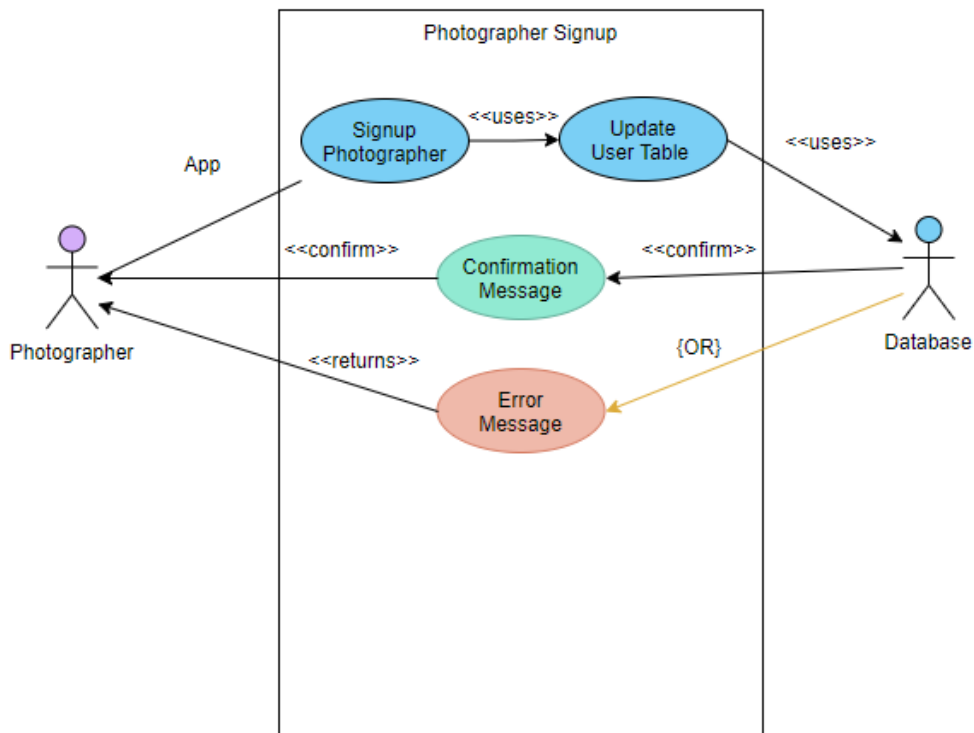
Figure 3.1: Photographer's Sign up Use Case Diagram

In Figure 3.1, there are two actors photographer and Database. First, photographer will signup then table in database will be updated. After that, a confirmation message will be sent to photographer.
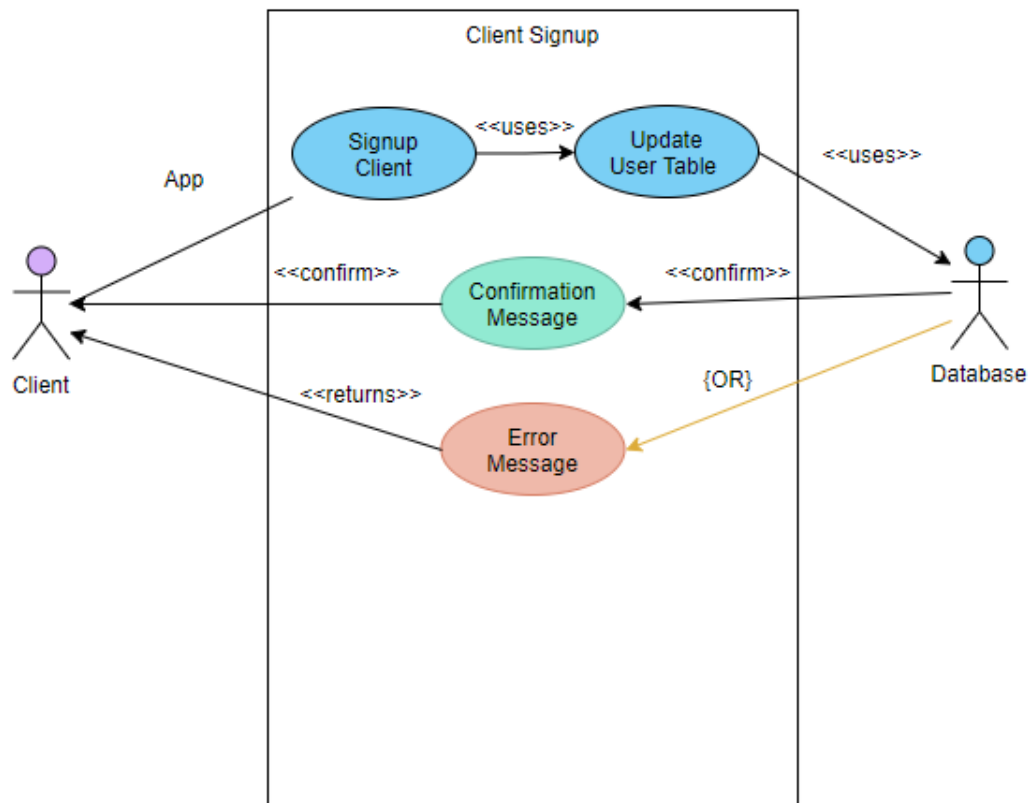
Figure 3.2: Client's Sign up Use Case Diagram

In Figure 3.2, there are two actors Client and Database. First, client will signup then table in database will be updated. After that, a confirmation message will be sent to client.
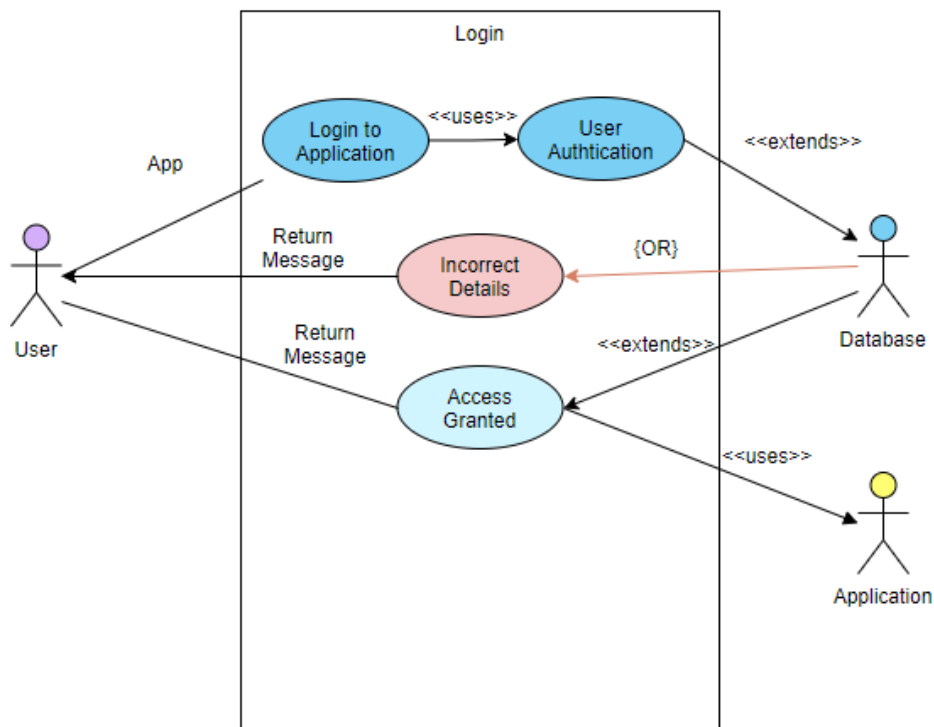
Figure 3.3: Login Use Case Diagram

In Figure 3.3, User will fill login form and enter on login button. Then database will check that either the user is authorized or not. If yes, then full access of application will be granted to user. If no, then an error message will be sent back to user.
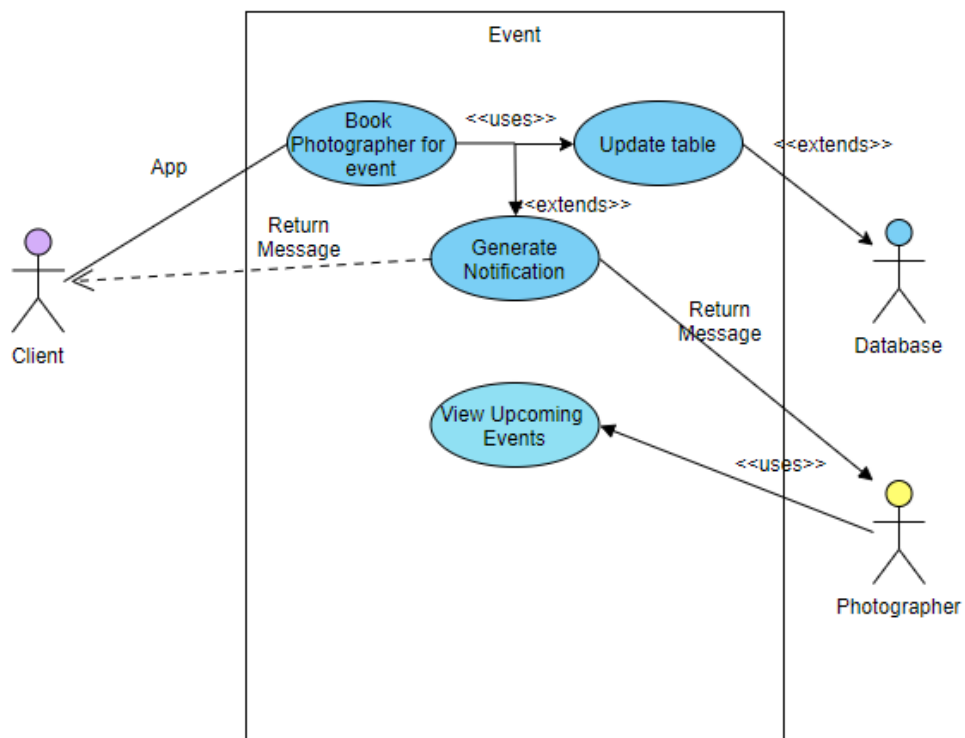
Figure 3.4: Event Use Case Diagram

In Figure 3.4, Client will book a photographer for a specific time and database at the same time will update the table. Also, application notify that photographer about his/her booking.
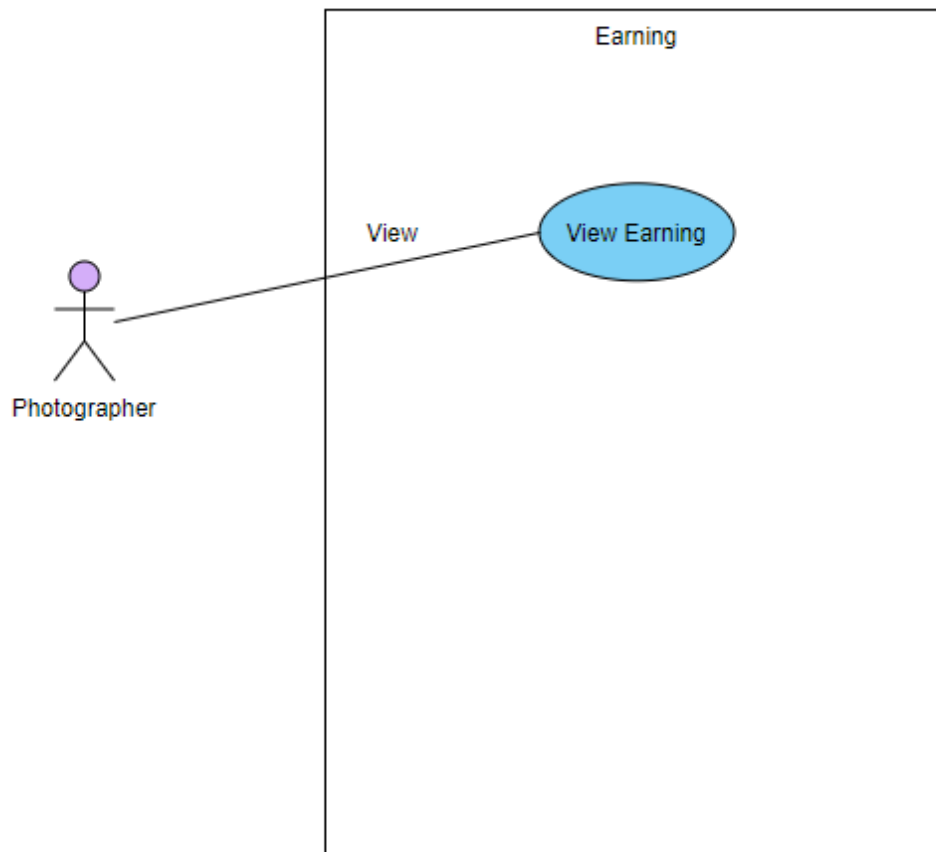
Figure 3.5: Earning Use Case Diagram

In Figure 3.5, Photographer has full access to check his/her earning.
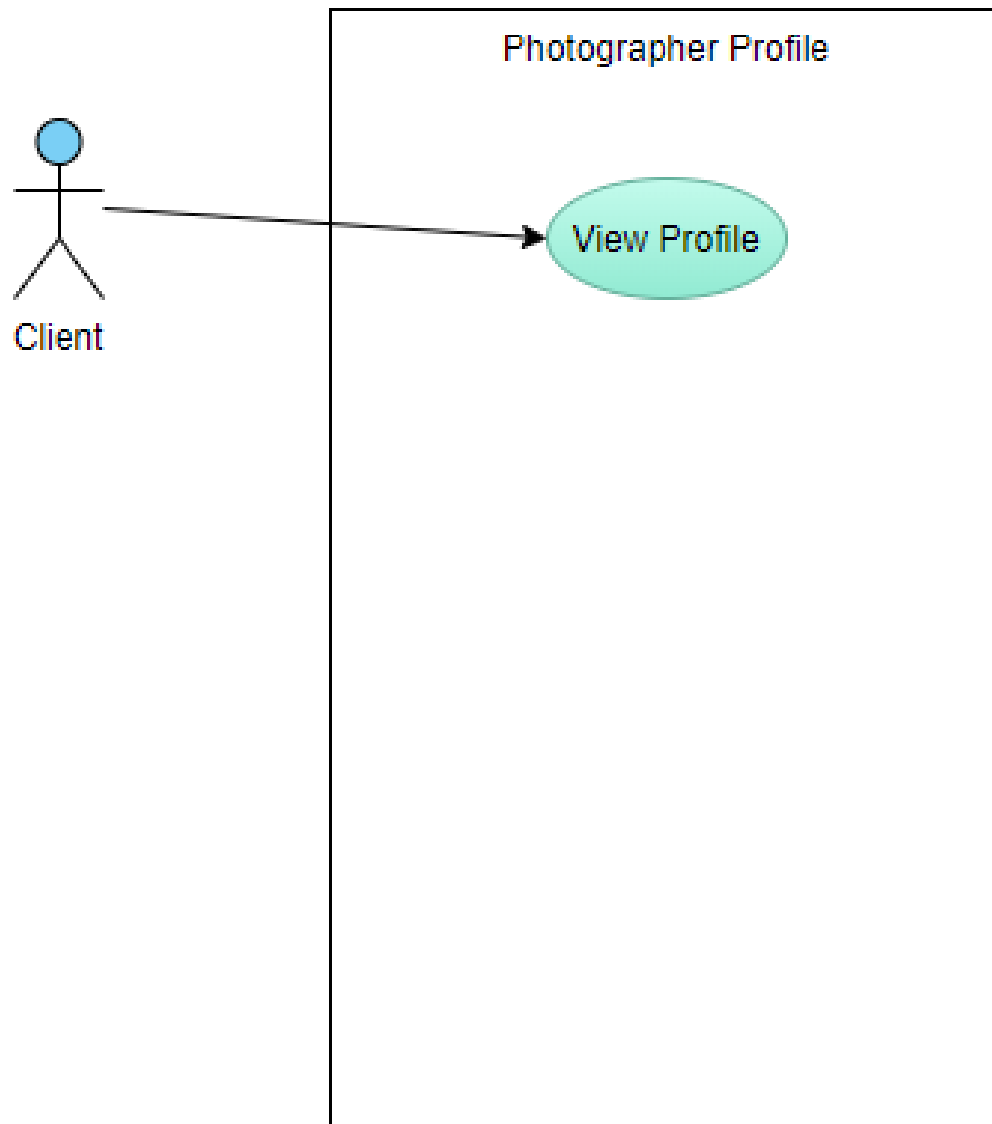
Figure 3.6: View Photographer's Profile Use Case Diagram

In Figure 3.6, Client can also view any photographer's profile.

3.5.1   Use Case Prioritization

The use cases for the project have been prioritized keeping in mind the following metrics:

- Functional Requirements

- Risk

- Development Effort

- Application Stability

Table 3.8: Use Case Prioritization

Priority	Description
High	Part of the functional requirements
	Affects the entire application flow
	Flow for other functional requirements
	will break if use case fails
Medium	Part of the functional requirements
	Flow for any other functional requirement
	will not break if use case fails
Low	Not part of the functional requirements
	Flow for any functional requirement will
	not break if use case fails

3.5.2   Use Case Description

Table 3.9: Login Use Case Description

Name	Login
ID	UCL01
Priority	High
Primary Actor	Photographer / Client
Description	This use case describes the login flow. Every user should be able to log in to the system with valid credentials.
Basic Flow	User will go to the login screen User will enter credentials User will be logged in
Alternate Flow	Invalid credentials entered User will be prompted with error message
Pre-condition	Login screen should be open Internet should be connected
Steps	Actor Actions: Step 1: User will enter username Step 2: User will enter password Step 3: User will tap on the Log in button System Response: Step 4: System will check if the credentials are valid Step 5: User will be directed to home screen
Post-condition	Home screen will be displayed

Table 3.10: Logout Use Case Description

Name	Log out
ID	UCL02
Priority	Low
Primary Actor	Photographer / Client
Description	This use case describes the logout flow. Every user should be able to log out from the system.
Basic Flow	User will open the profile menu User will select the log out option User will be logged out
Steps	Actor Actions: Step 1: User will open the profile menu Step 2: User will tap on the log out button Step 3: User will tap on confirm button System Response: Step 4: System will Prompt user with a confirmation popup Step 5: User will be directed to login screen
Post-condition	User will be logged out from the system

Table 3.11: Change password Use Case Description

Name	Change Password
ID	UCP01
Priority	Low
Primary Actor	Photographer / Client
Description	This use case describes the change password flow. Every user should be able to change password from profile settings.
Basic Flow	User will go to the profile settings screen User will select change password option User will enter old password User will enter new password User will confirm new password Password will be changed successfully
Alternate Flow	Old Password is incorrect OR New passwords do not match System will show error message and password will not be changed
Pre-condition	User should be logged in Internet should be connected
Steps	Actor Actions: Step 1: User will open the profile menu Step 2: User will tap on the profile settings button Step 3: User will tap on the change password button Step 4: User will enter old password Step 5: User will enter new password Step 6: User will confirm new password Step 7: User will tap on the confirmation button System Response: Step 8: System will check if the credentials are valid Step 9: System will update the password in the system Step 10: System will prompt the user with a confirmation message
Post-condition	Password will be changed successfully

Table 3.12: User Signup Use Case Description

Name	User Signup
ID	UCR01
Priority	High
Primary Actor	Photographer / Client
Description	This use case describes the user signup flow. Photographer will be able to create his/ her portfolio. Client will be able to see profiles of different photographers.
Basic Flow	Actor will go to the signup. Actor will enter the required information New user will be registered successfully
Alternate Flow	User already exists System will prompt the Actor with error message New user will not be created
Pre-condition	Photographer or client should be logged in Internet should be connected
Steps	Actor Actions: Step 1: User will choose the Create User option Step 2: User will enter the required information Step 3: User will choose the confirmation option System Response: Step 4: System will check if the credentials are valid Step 5: New user will be registered successfully into the system Step 6: System will prompt the user with a confirmation message
Post-condition	New user is signed up successfully

Table 3.13: Update Photographer Profile

Name	Update Photographer Profile
ID	UCUP01
Priority	Medium
Primary Actor	Photographer
Description	This use case describes the update profile flow. Photographer should be able to update their own profile password and portfolio.
Basic Flow	User will go to the update profile screen User will update the information System will update information successfully
Alternate Flow	Password does not match the required format System will show error message
Pre-condition	User should be logged in Internet should be connected
Steps	Actor Actions: Step 1: User will select the update profile option Step 2: User will enter updated profile information Step 3: User will choose the confirmation option System Response: Step 4: System will check if the information is valid Step 5: User information will be updated in the system
Post-condition	Information will be updated successfully

Table 3.14: Book Photographer Use Case Description

Name	Book Photographer
ID	UCR02
Priority	High
Primary Actor	Client
Description	This use case describes the booking of photographer by client for an event.
Basic Flow	User will go to the photographer profile User will click on booking button Select data and time User will click on Confirm booking button.
Alternate Flow	User is unable to book photographer. System will prompt with error message
Pre-condition	Client should be logged in Internet should be connected
Steps	Actor Actions: Step 1: Actor will view the Photographer Profile Step 2: Actor will click in Book Button Step 3: Actor will enter data and time Step 4: Actor will click on confirm button System Response: Step 5: System will notify both client and photographer about booking Step 6: Booking will be added to photographer Events.
Post-condition	Information will be updated successfully

Table 3.15: Chat Use Case Description

Name	Chat
ID	UCM01
Priority	Medium
Primary Actor	Photographer / Client
Description	This use case describes the messaging flow. Users should be able to send and receive messages through the system.
Basic Flow	User1 will go to the messaging option User1 will send message to User2 Message will be sent and received successfully
Pre-condition	User1 should be logged in Internet should be connected
Steps	Actor Actions: Step 1: User1 will select the message option Step 2: User1 will select User2 Step 3: User1 will type message Step 4: User1 will send message Step 6: User2 will log in Step 7: User2 will select the message option Step 8: User2 will view the received message System Response: Step 5: System will send message to User2
Post-condition	Message will be sent and received successfully

Table 3.16: View Review and Ratings Use Case Description

Name	View Reviews and Ratings
ID	UCR01
Priority	Medium
Primary Actor	Client
Description	This use case describes the view Reviews and ratings of Photographer's workflow. Client should be able to view Reviews and Ratings in Photographer's profile.
Basic Flow	User will go to the Photographer profile User will view reviews and ratings.
Pre-condition	Client user should be logged in Internet should be connected
Steps	Actor Actions: Step 1: User will go to Photographer profile screen System Response: Step 2: Review and ratings will be displayed
Post-condition	Review and Ratings will be displayed successfully

Chapter 4

System Design

4.1 Introduction

A mobile app for photographers in general where they can create their portfolio and show their work to people. Also, it can be a great opportunity for new photographers that are seeking to work with well reputed studios or use their photography skills in an event to enhance their experience and improve their hard and soft skills both.

4.2 System Architecture

The provided environment communicates the proposes system architecture as shown below in fig 4.1.
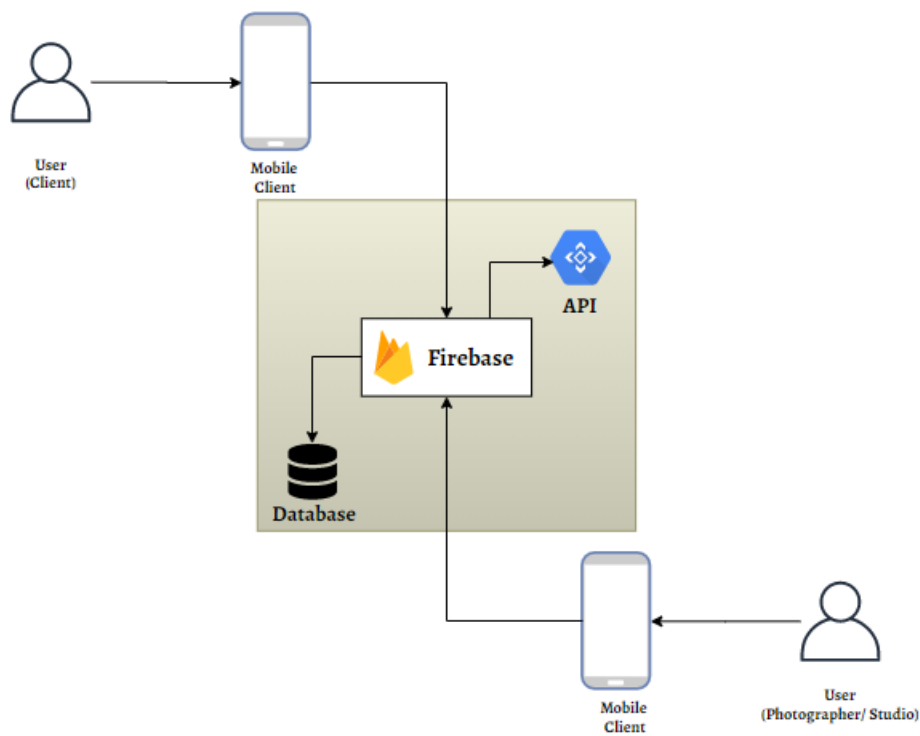
Figure 4.1: System Architecture

4.3 Sequence Diagram

4.3.1 Login Sequence
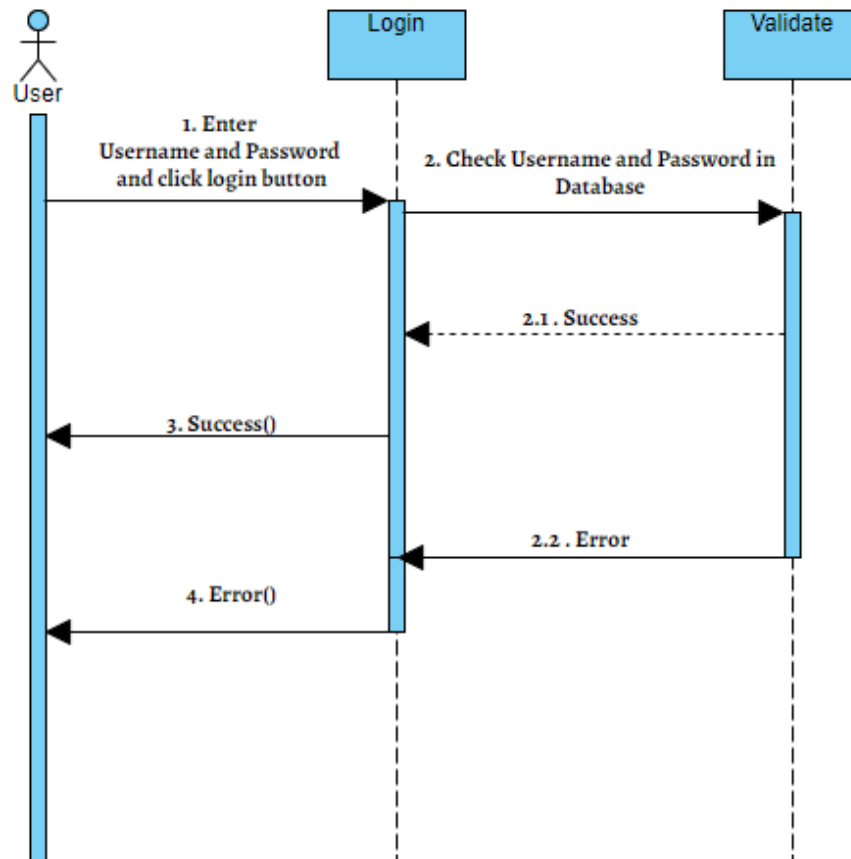
Figure 4.2: Login Sequence Diagram

If the entered credentials are correct, the login will be successful, and the user will proceed to the home screen. Otherwise, error message will be shown, and login will be unsuccessful.

4.3.2   Signup Sequence
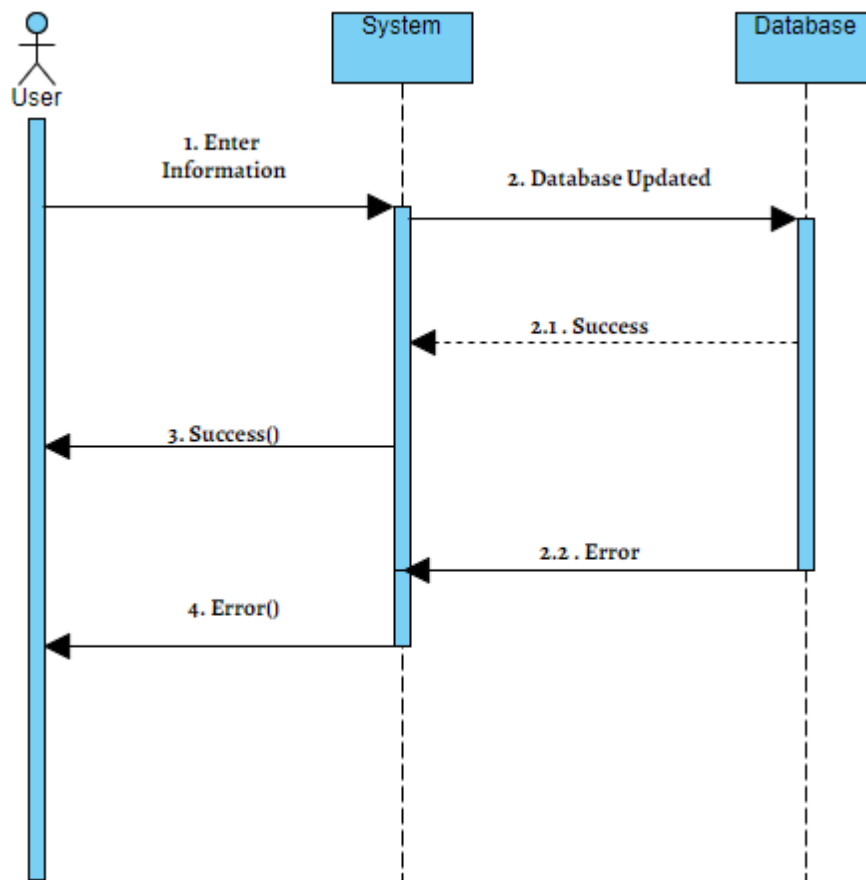
Figure 4.3: Signup Sequence Diagram

User will sign up by providing information, unique username, and password. System checks if the user does not exist and email is unique, user is signup successfully, and confirmation message is shown. Otherwise, error message is shown.

4.3.3   Chat Sequence

Users can share messages with each other. If the User2 is found in the system, message is sent to the User2 and success message is shown to User. Message is also stored in the database.
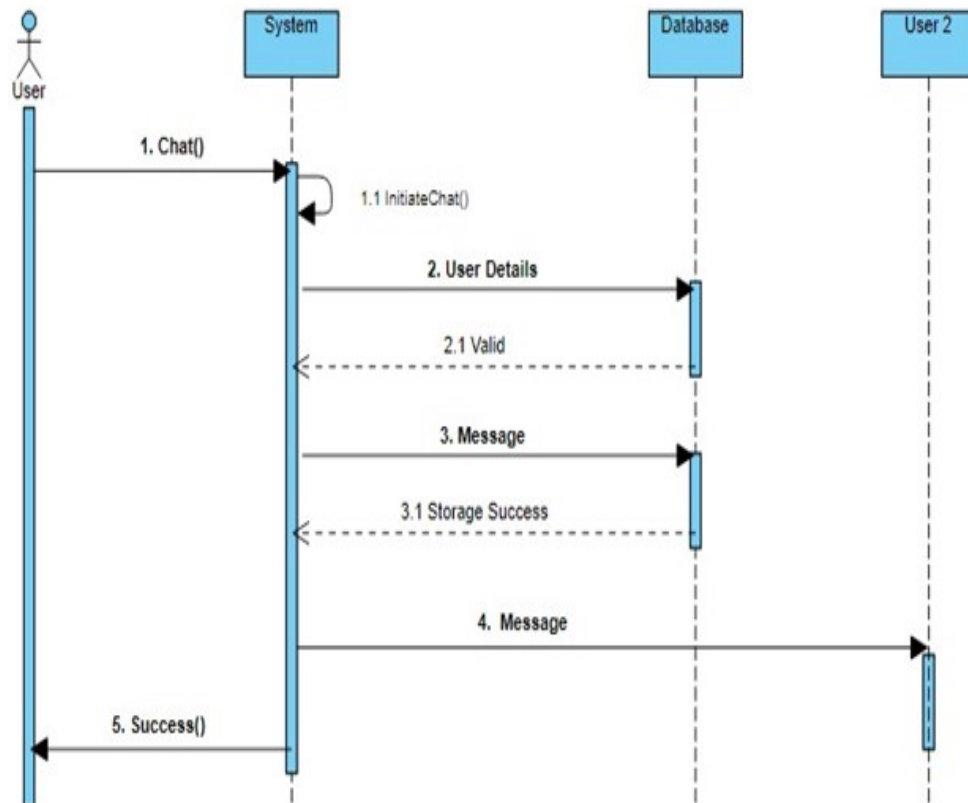
Figure 4.4: Chat Sequence Diagram

4.3.4   Photographer Booking for Event Sequence
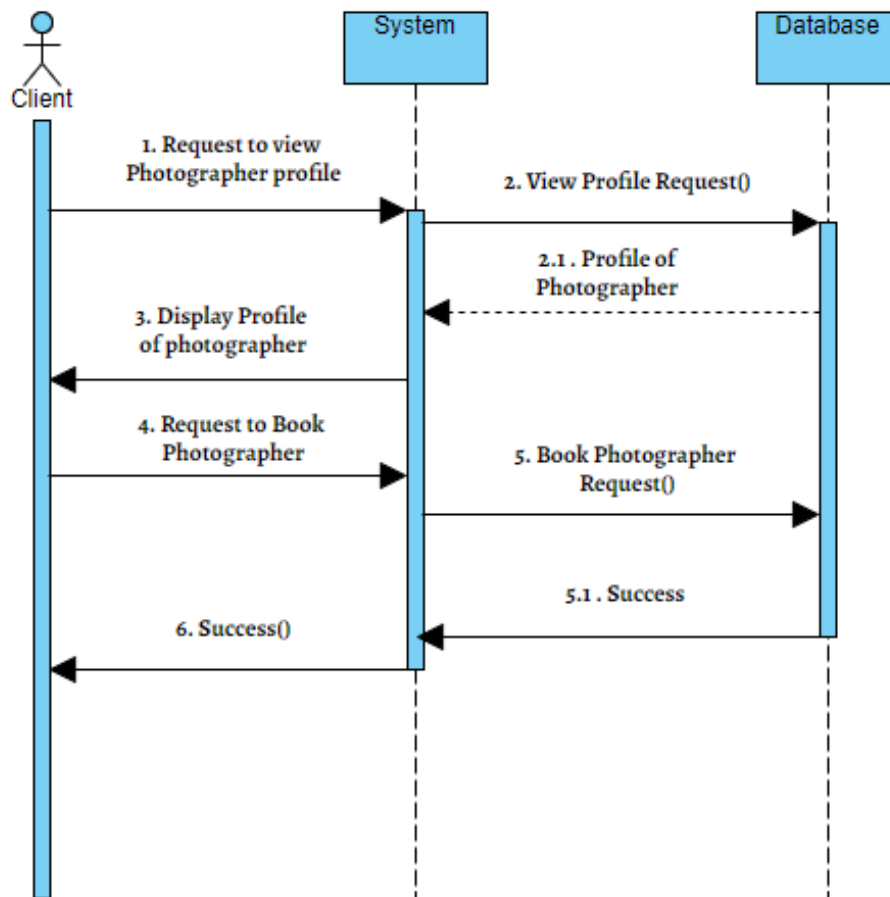
Figure 4.5: Photographer Booking for Event Sequence Diagram

Client users can book photographer in the system. The view profile request is sent to the system and profile of photographer fetched from database and given to system; system will display photographer's profile on screen. Client then book the photographer and sends post request. Booking record is stored in database and success message is shown.

4.3.5 View Event Sequence
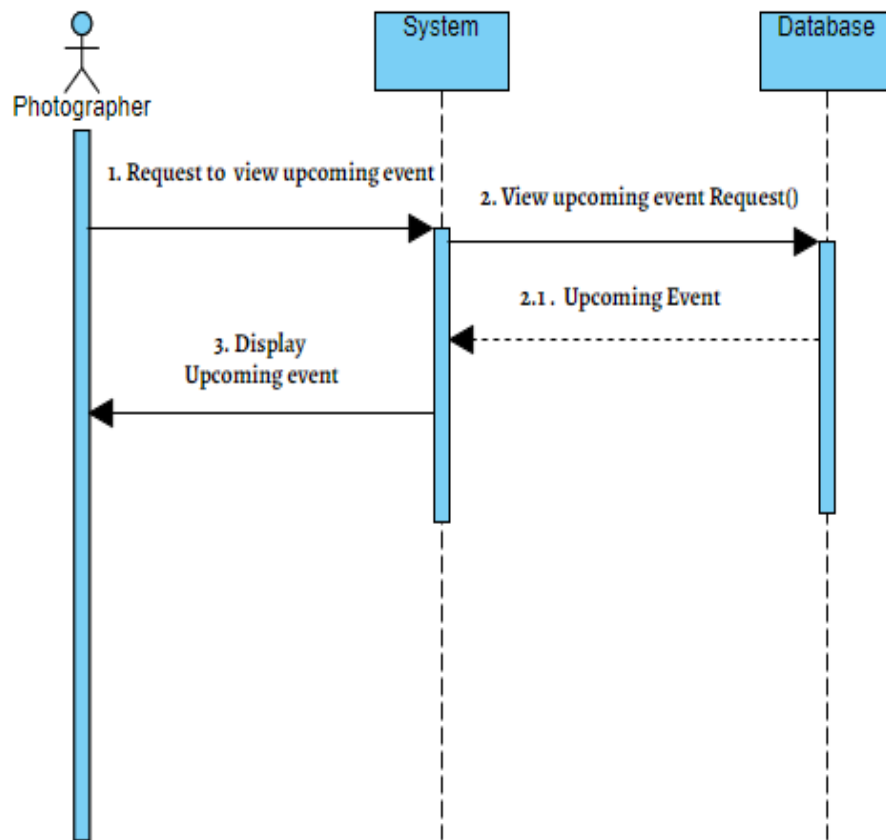
Figure 4.6: View Event Sequence Diagram

Photographer can view event. Event list is fetched from database and shown to user.
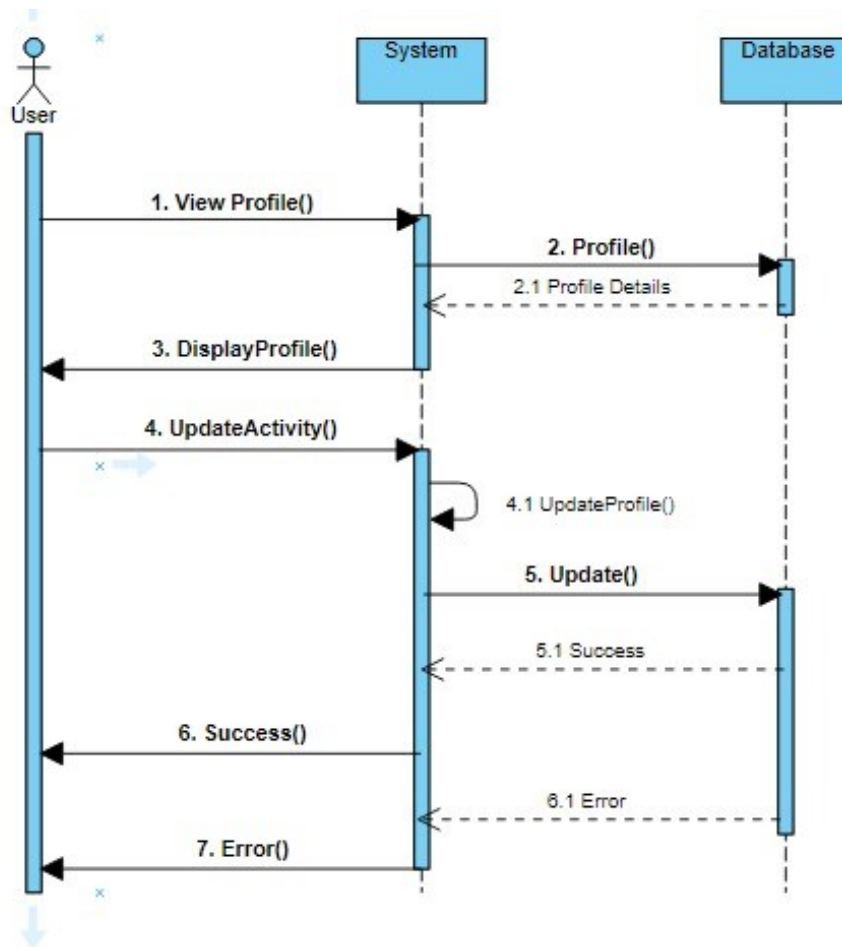
4.3.6   Update Profile Sequence

Figure 4.7: Update Profile Sequence Diagram

All photographers can update their profile in system. Profile is displayed and then the user updates profile. After update is done, success message is shown to user.

4.4   Activity Diagram

Activity diagram is another important diagram in UML to describe the dynamic aspects of
the system. Activity diagram is basically a flowchart to represent the flow from one activity
to another activity. The activity diagram of our project is given below:
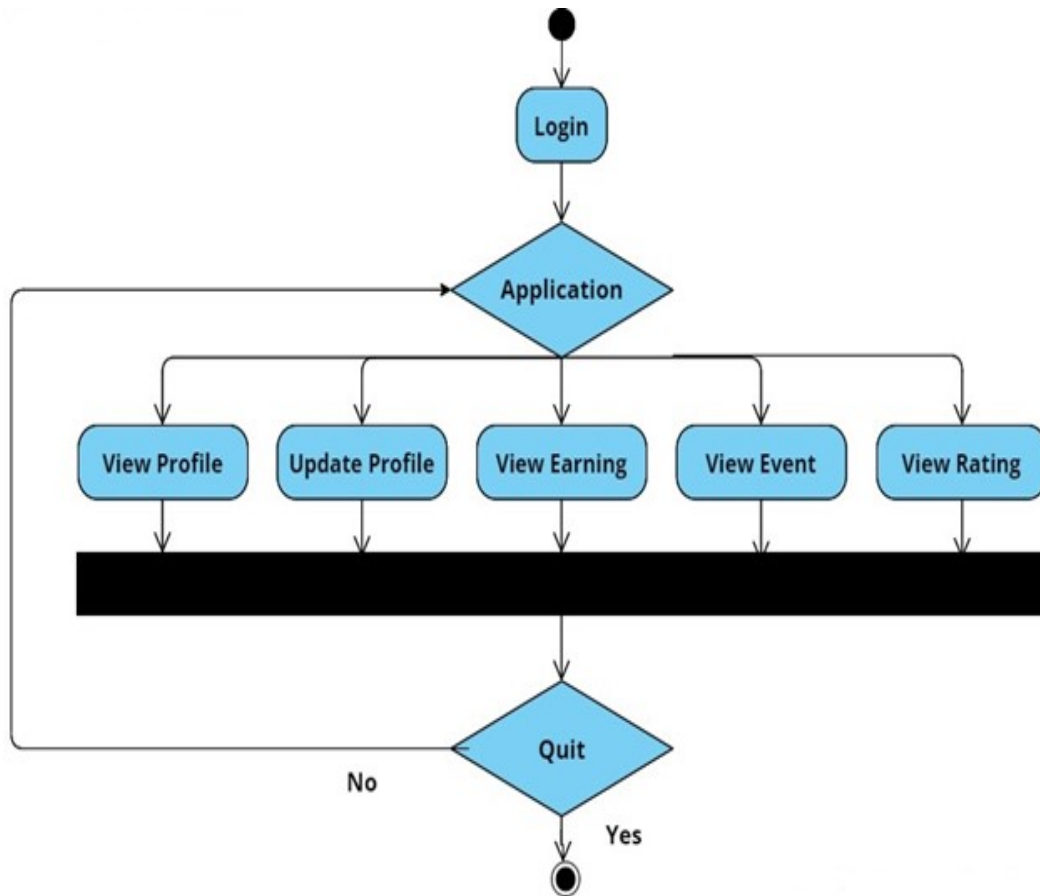
Figure 4.8: Activity Diagram

Chapter 5

System Implementation

System implementation is a process which explains how the system should be built, make sure that the system is meeting all requirements and the quality standards.

5.1 System Architecture

The system based on the 3-tier architecture.

- Data Layer
- Business Layer
- Presentation Layer

5.2 Advantages of the 3-tier Architecture

- System layers can be updated and modified independently.
- Different teams can work on different aspects of the application.
- More secure in nature, as the end user is interacting with presentation layer, which then interacts with the business layer, keeping the data layer secure.

5.3 Tools and Technologies

5.3.1 Android Studio

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems or as a subscription-based service in 2020. It is a replacement for the Eclipse Android Development Tools (E-ADT) as the primary IDE for native Android application development. We have deveoped the whole application on it.

5.3.2 Firebase

Firebase is a platform developed by Google for creating mobile and web applications. It was originally an independent company founded in 2011. In 2014, Google acquired the platform and it is now their flagship offering for app development.Firebase's first product was the Firebase Realtime Database, an API that synchronizes application data across iOS, Android, and Web devices, and stores it on Firebase's cloud. The product assists software developers in building real-time, collaborative applications. We have used their database to store information of both client an photographer. Also, all the information related to both side communication is saved in it.

5.3.3 Java

Java is the number 1 programming language and development platform. It reduces costs, shortens development timeframes, drives innovation, and improves application services. With millions of developers running more than 51 billion Java Virtual Machines worldwide, Java continues to be the development platform of choice for enterprises and developers. We have used this language for authentication at different points in our application.

5.3.4 Python

For machine learning based applications Python is a very powerful platform because there is a vast set of libraries that makes the development process much easier and efficient thats why Python was used to train and test the recommender system.

5.4 Recommender System

An artificially intelligent recommender system trained using the unsupervised approach is also used to give studio suggestions to the user based on the overall explicit feedback collected from the users on all the studios registered in the system.

5.4.1 Collaborative Filtering

Collaborative methods for recommender systems gives recommendations solely on the past interactions(**ratings**) recorded between users and items(**studios**) to produce new recommendations. The ratings from users are stored in an user interaction matrix. The user interaction matrix from this web application would be

Figure 5.1: User interaction matrix

	Mahad Studio	Hamza Studio	Waseem Studio	Asad Studio	Adil Studio
User 1	3	5	2	1	0
User 2	5	1	2	3	4
User 3	4	2	3	1	5
User 4	3	2	5	4	1

5.4.2   Collaborative Filtering Approach

Model based collaborative filtering approach was used to create the recommender system with K-Nearest Neighbours (KNN) being the primary model trained on Python. Similarities between user's choices and ratings on studios are calculated based on an unsupervised learning model. In this approach, we also can limit the number of similar users as k, which makes system more scale-able in case the number of users are increased in the future.
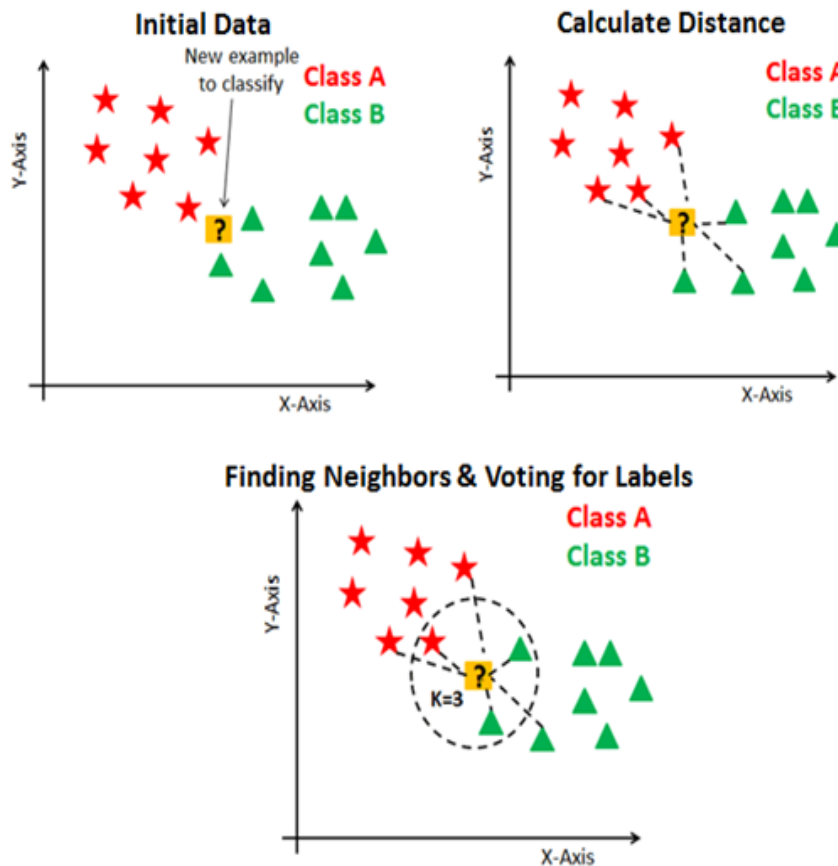

Figure 5.2: KNN Algorithmic Working

5.4.3   Dataset

For 124 Photography studios 953,296 Ratings from different customers were used to train the recommender system.

5.4.4   Mode Training

To train the model **NearestNeighbors** Python's from sklearn.neighbors was used which basically allows the user to choose between the three different nearest neighbors algorithms BallTree, KDTree, and a brute-force algorithm. For the development of this web application the brute force approach was selected. Brute Force Algorithm involves the brute-force computation of distances between all pairs of points in the data set which is feasible for the data set used for model training but not feasible if a very large data set is used for model training.

5.4.5 Mode Testing

To test the model following steps were taken

1. A test data set was created by using the following approach

   (a) Rating of a given studio was stored in a variable for comparison.

   (b) 10 nearest Recommendations were generated from the model and their ratings were also stored for comparison.

   (c) The difference between the current studios rating and the recommendations rating is also stored for comparison.

2. If the differences are near to the current studio's rating then they are given a class **1** in the testing data set else **0** based on the difference acceptance value set by the programmer.

Figure 5.3: Model Evaluation

Range	>=1	>=1.7
Recommendations that are closer to the current studio's prediction **(1)**	13%	80%
Recommendations that are not closer to the current studio's prediction **(0)**	86%	19%

5.5 Mobile Application Modules

The application has been divided into two modules, one for each user:

1. Studio / Photographer Module

2. Client Module

5.6   System Internal Components

5.6.1   Splash Screen

When user will open the application, a welcome splash screen is displayed which makes the user aware about the system status hence insuring feedback aspect of usability engineering.
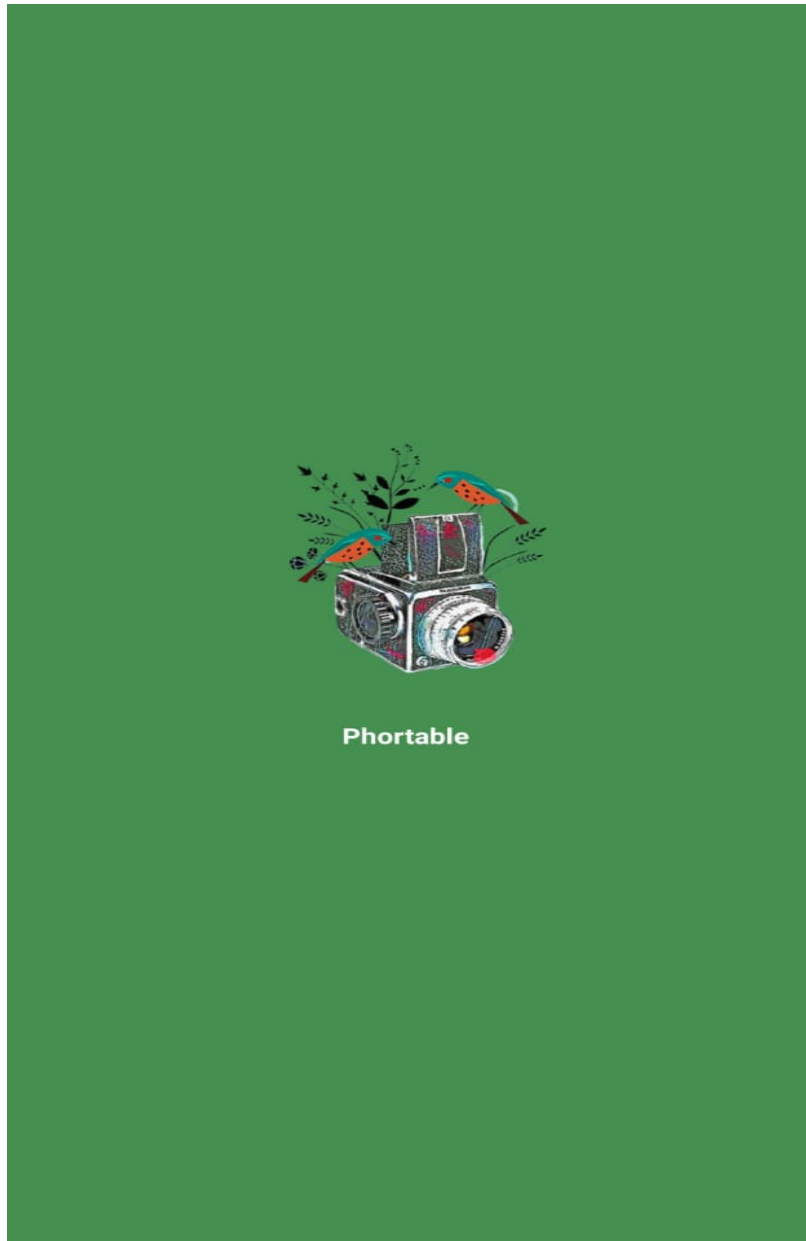
Figure 5.4: Welcome Splash Screen

5.6.2 Login/Sign Up Screen

When a user is already registered in the application they can login in to the app using the email and the password they used for the sign up process. The email and the password are stored in the database. If the user is not signed up already they can sign up in the application by providing their name, number, address, email, password and select their category from either a studio or a customer.

Figure 5.5: Login Screen

Figure 5.6: Sign Up Screen

5.6.3 Available Studios Screen

Studios that have already been registered in the application are now visible for the general user for an event booking. They can view all options here and choose the studio of their choice.
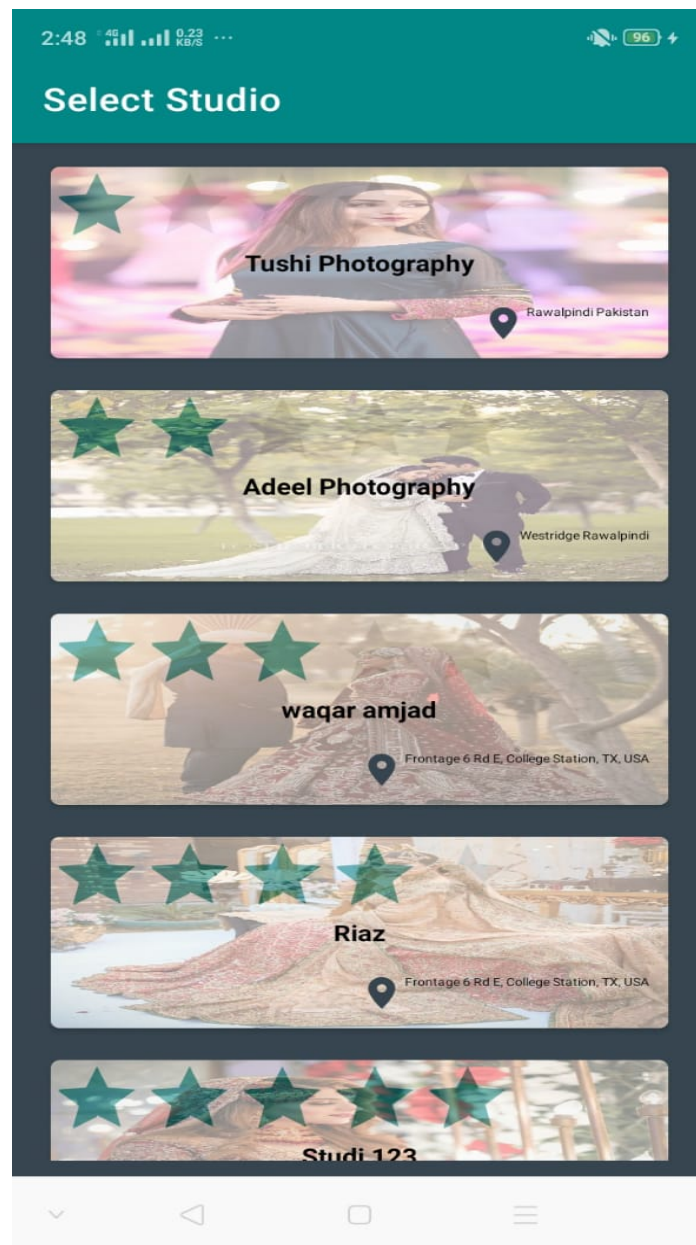

Figure 5.7: Available Studios Screen

5.6.4   Studio Portfolio Screen

The user can now view the portfolio or work by the studio of their choice if they like the work they can choose the book now option from the same page or choose to chat first.
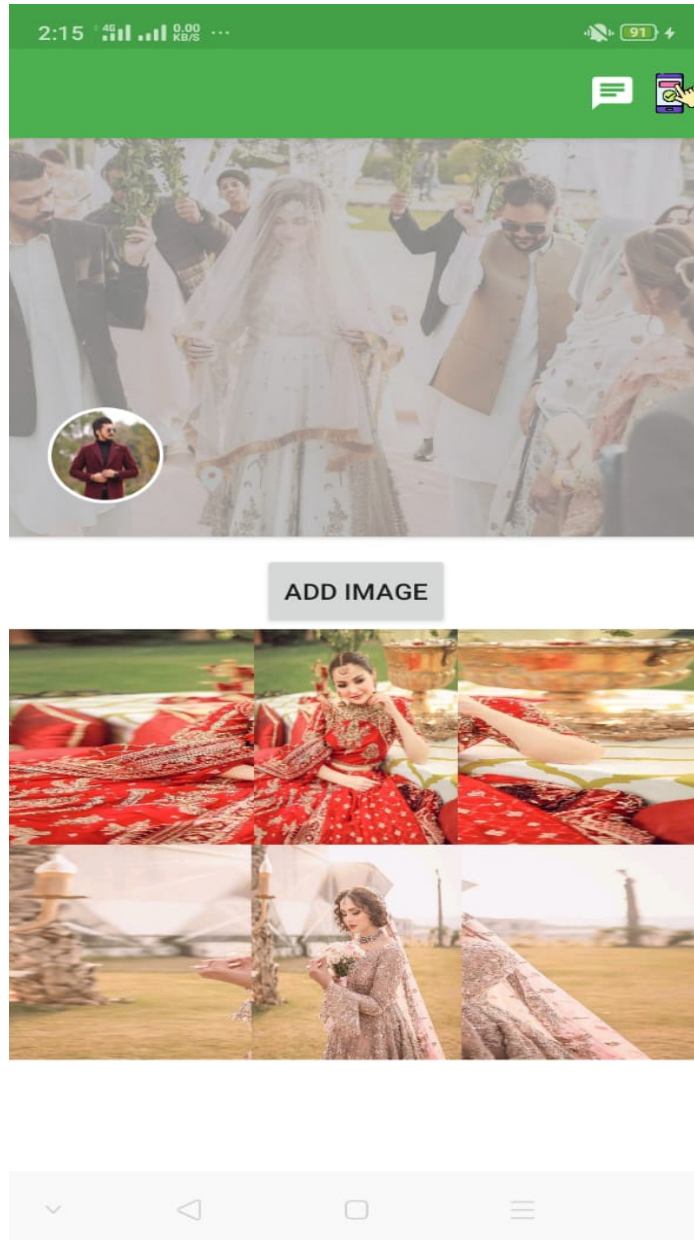

Figure 5.8: Portfolio Screen

5.6.5 Chat

An API is used to send an OTP to the users phone whenever they login to the chat first time this was done for information security purposes. Once logged in studio and the user can chat to discuss an event before or after the booking an event.
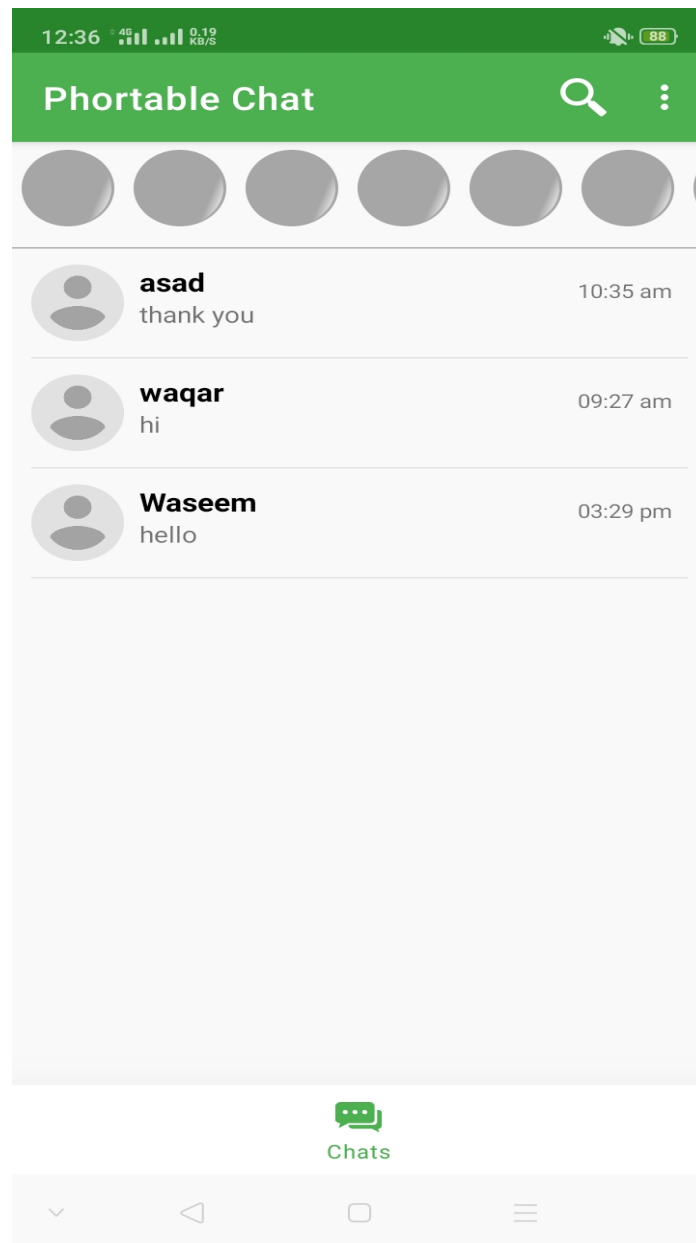

Figure 5.9: Chat Screen

5.6.6   Booking Screen

The user can now view the available booking options of the selected studio and the options are Single Photo, Single Video, Drone and Album. The total bill from their choices will be calculated by the app and the booked is saved in the database and can be viewed on the admin dashboard of the system.
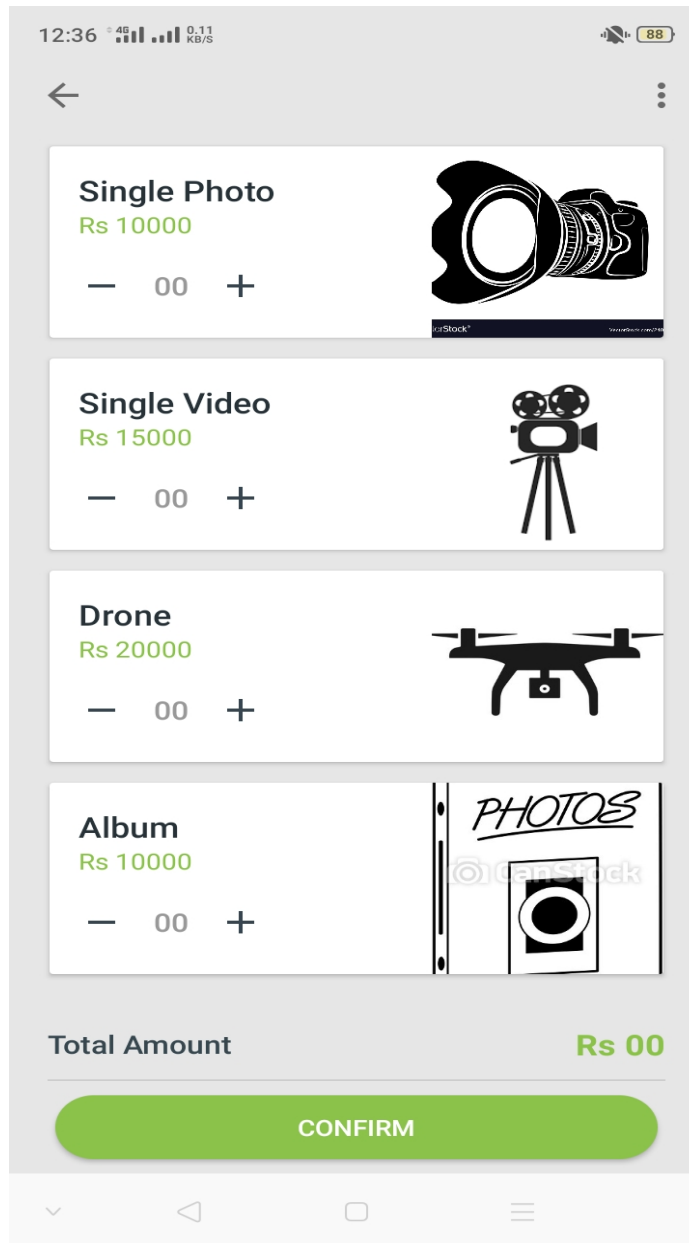

Figure 5.10: Booking Screen

5.6.7 Rating Screen

To gather and collect explicit feedback from the users about the work of studios they can choose the rating option. Their ratings are stored in the database and that is used by the recommender system to generate studio suggestions.
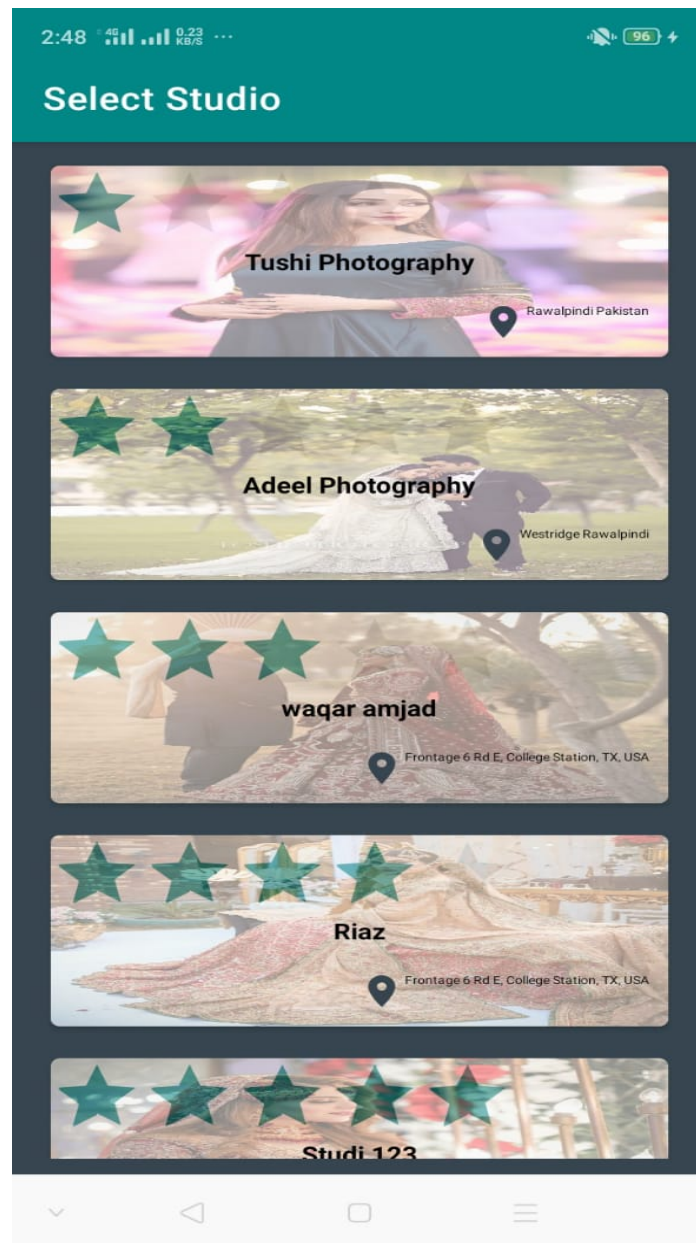

Figure 5.11: Rating Screen

5.6.8   Suggestion Screen

Based on the recommendations from the recommnder systems a list of studio suggestions are generated for the users on the explicit feedback.
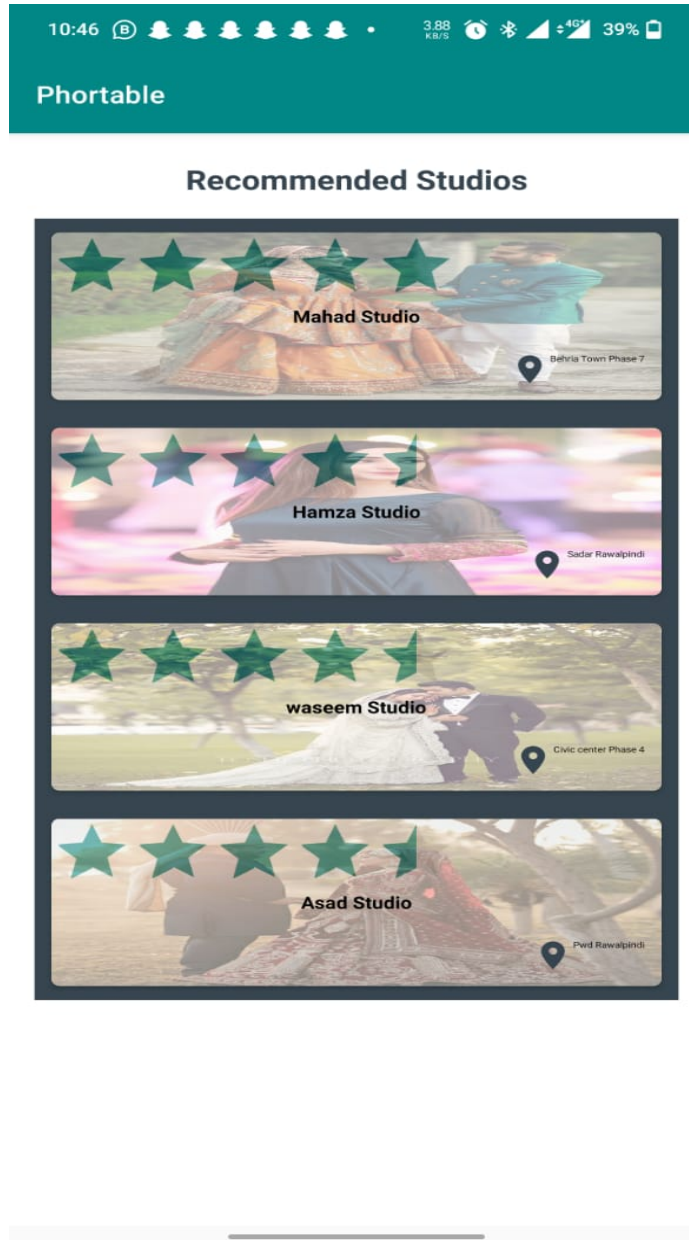

Figure 5.12: Rating Screen

5.6.9 Admin Module Login Screen

This the web dashboard that is developed to manage and control the records in Fire base database for the users and their respective bookings first the admin would log in the dashboard.
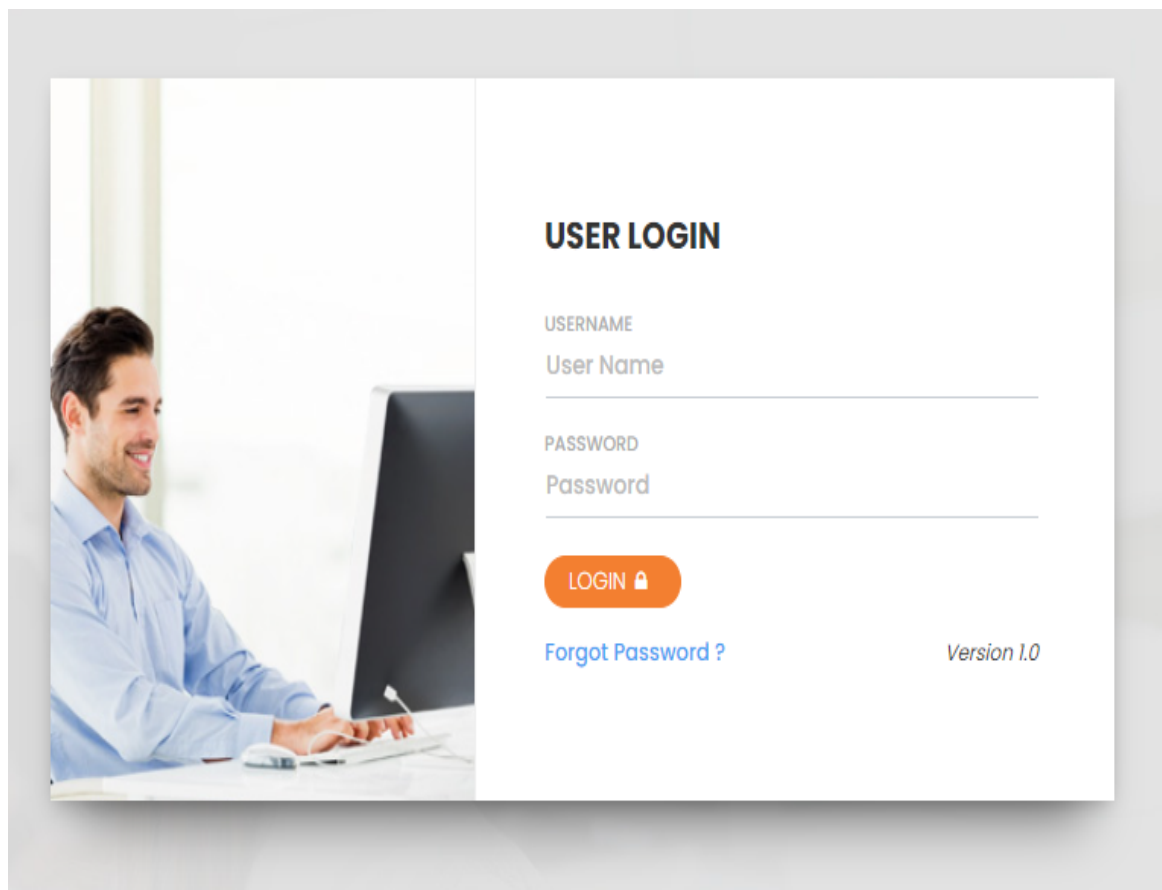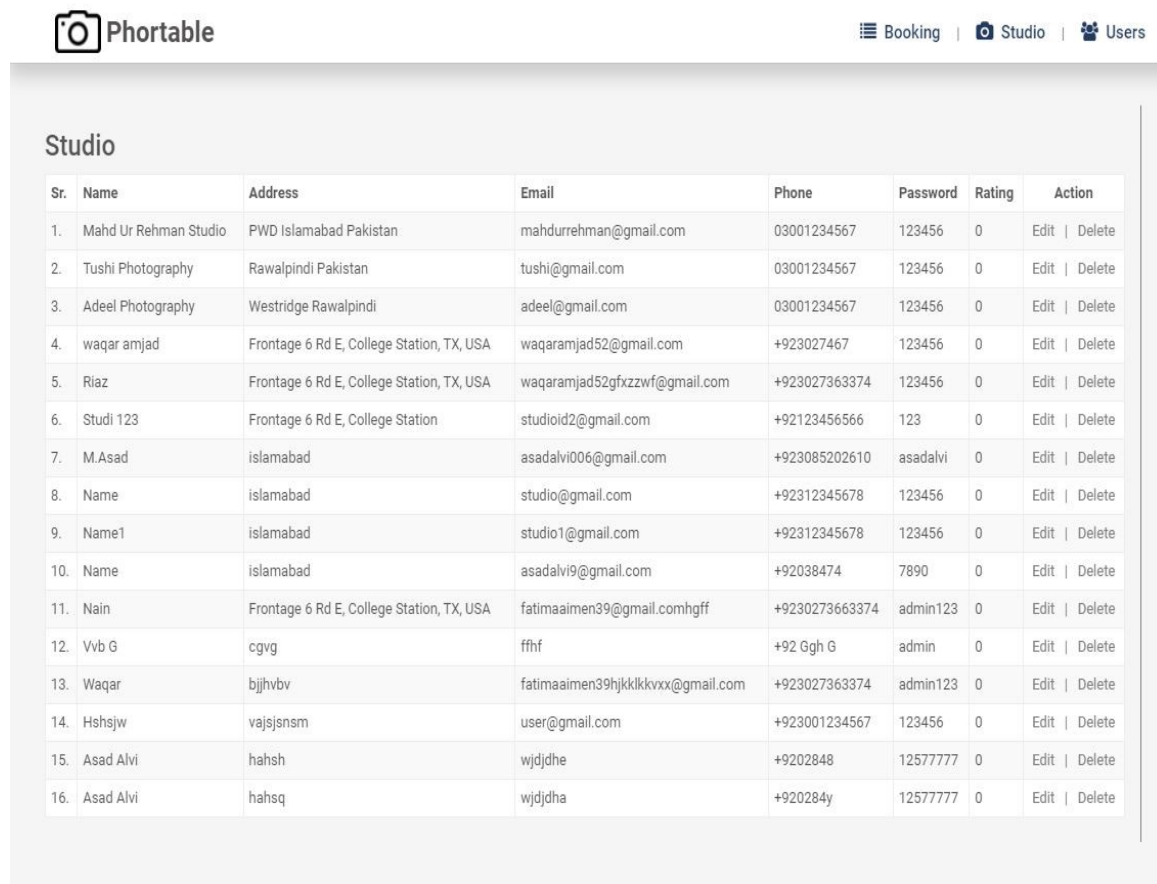

Figure 5.13: Admin Module Login Screen

5.6.10 Admin Module Data View

The admin can view and work with the data of bookings, users and studios.

Figure 5.14: Studio Data Screen

Figure 5.15: Booking Data Screen

Figure 5.16: Users Data Screen

Chapter 6

System Testing and Evaluation

6.1 Introduction

The safety elements for application are quite exiting to discover and want thorough testing cycles to be performed. The first-class way to keep away from any system vulnerabilities is to adopt the testing method inside the early implementation cycle of the application. Testing phase can be defined because as the running and evaluation of system with the cause of finding vulnerabilities and errors.Testing guarantees the excellent of an application and complements its integrity by using mentioning the flaws and errors inside the machine.

6.2 Testing Approaches

To test the application, different approaches can be followed, and the test cases were designed accordingly. The following approaches will be described further in this chapter:

- Black box Testing

- GUI Testing

- Performance Testing

- Usability Testing

- Acceptance Testing

6.2.1 Black box Testing

Black box testing is a testing technique in which functionalities of application are tested without considering internal code structure of the application.
We have tested the application and validated inputs as per system requirements. The output displayed by system was compared to expected output to ensure quality standards.

6.2.2 GUI Testing

Graphical User Interface (GUI) testing is a testing technique in which application working is checked to ensure whether the application is working fine or not. This is done by checking whether the application performing all stated tasks or not and the output is according to expected output or not. We have tested the application and interacted with GUI. We tested the components of application's interface and passed all use cases by running multiple test cases.

6.2.3 Performance Testing

The overall response time of the system is noted in this step.
We checked that the system response time was on time, and the application's performance was meeting the quality standards.

6.2.4 Usability Testing

This approach helps in identifying the UI / UX related issues and design and user interface drawbacks in the system.
In this approach we performed the following:

- Checked if users were able to complete all tasks successfully

- Checked the completion time of a task

- Checked the satisfaction rate of users

- Evaluated application performance and matched it with our system requirements

6.2.5 Acceptance Testing

Finally in this phase, the main requirements of the system specified in the acceptance criteria were tested and after a complete evaluation, the application passed the testing phase.

6.3 Test Cases

Figure 6.1: Login Test Case

Name	Login
ID	01
Priority	High
Primary Actor	Photographer / Client
Description	This test case describes the login flow. Every user should be able to log in to the system with valid credentials.
Basic Flow	Tester will go to the login screen Tester will enter credentials Tester will be logged in
Alternate Flow	Invalid credentials entered Tester will be prompted with error message
Pre-condition	Login screen should be open Internet should be connected
Steps	**Actor Actions:** Step 1: Tester will enter username Step 2: Tester will enter password Step 3: Tester will tap on the Log in button **System Response:** Step 4: System will check if the credentials are valid Step 5: Tester will be directed to home screen
Post-condition	Home screen will be displayed

Figure 6.2: Logout Test Case

Name	Log out
ID	02
Priority	Low
Primary Actor	Photographer / Client
Description	This test case describes the logout flow. Every tester should be able to log out from the system.
Basic Flow	Tester will open the profile menu Tester will select the log out option Tester will be logged out
Pre-condition	Tester should be logged in
Steps	**Actor Actions:** Step 1: Tester will open the profile menu Step 2: Tester will tap on the log out button Step 4: Tester will tap on confirm button **System Response:** Step 3: System will Prompt teste with a confirmation popup Step 5: Tester will be directed to login screen
Post-condition	Tester will be logged out from the system

Figure 6.3: Change Password

Name	Change Password
ID	03
Priority	Low
Primary Actor	Photographer / Client
Description	This test case describes the change password flow. Every tester should be able to change password from profile settings.
Basic Flow	Tester will go to the profile settings screen Tester will select change password option Tester will enter old password Tester will enter new password Tester will confirm new password Password will be changed successfully
Alternate Flow 2	Old Password is incorrect OR New passwords do not match System will show error message and password will not be changed
Pre-condition	Tester should be logged in Internet should be connected
Steps	**Actor Actions:** Step 1: Tester will open the profile menu Step 2: Tester will tap on the profile settings button Step 3: Tester will tap on the change password button Step 4: Tester will enter old password Step 5: Tester will enter new password Step 6: Tester will confirm new password Step 7: Tester will tap on the confirmation button **System Response:** Step 8: System will check if the credentials are valid Step 9: System will update the password in the system Step 10: System will prompt the teste with a confirmation message
Post-condition	Password will be changed successfully

Figure 6.4: Tester Sign up

Name	Tester Signup
ID	04
Priority	High
Primary Actor	Photographer / Client
Description	This test case describes the tester signup flow. Photographer will be able to create his/ her portfolio. Client will be able to see profiles of different photographers.
Basic Flow	Actor will go to the signup. Actor will enter the required information New tester will be registered successfully
Alternate Flow	Tester already exists System will prompt the Actor with error message New tester will not be created
Pre-condition	Photographer or client should be logged in Internet should be connected
Steps	**Actor Actions:** Step 1: Tester will choose the Create User option Step 2: Tester will enter the required information Step 3: Tester will choose the confirmation option

System Response: Step 4: System will check if the credentials are valid Step 5: New tester will be registered successfully into the system Step 6: System will prompt the tester with a confirmation message |
| Post-condition | New teste is signed up successfully |

Figure 6.5: Update Photographer Profile

Name	Update Photographer Profile
ID	05
Priority	Medium
Primary Actor	Photographer
Description	This test case describes the update profile flow. Photographer should be able to update their own profile password and portfolio.
Basic Flow	Tester will go to the update profile screen Tester will update the information System will update information successfully
Alternate Flow	Password does not match the required format System will show error message
Pre-condition	Tester should be logged in Internet should be connected
Steps	**Actor Actions:** Step 1: Tester will select the update profile option Step 2: Tester will enter updated profile information Step 3: Tester will choose the confirmation option **System Response:** Step 4: System will check if the information is valid Step 5: Tester information will be updated in the system
Post-condition	Information will be updated successfully

Figure 6.6: Book Photographer

Name	Book Photographer
ID	06
Priority	High
Primary Actor	Client
Description	This test case describes the booking of photographer by client for an event.
Basic Flow	Tester will go to the photographer profile Tester will click on booking button Tester will Select data and time Tester will click on Confirm booking button.
Alternate Flow	Tester is unable to book photographer. System will prompt with error message
Pre-condition	Tester should be logged in Internet should be connected
Steps	**Actor Actions:** Step 1: Actor will view the Photographer Profile Step 2: Actor will click in Book Button Step 3: Actor will enter data and time Step 4: Actor will click on confirm button **System Response:** Step 5: System will notify both client and photographer about booking Step 6: Booking will be added to photographer Events.
Post-condition	Information will be updated successfully

Figure 6.7: Chat

Name	Chat
ID	07
Priority	Medium
Primary Actor	Photographer / Client
Description	This test case describes the messaging flow. Users should be able to send and receive messages through the system.
Basic Flow	Tester will go to the messaging option Tester will send message to dummy user Message will be sent and received successfully
Pre-condition	Tester should be logged in Internet should be connected
Steps	**Actor Actions:** Step 1: Tester will select the message option Step 2: Tester will select dummy user Step 3: Tester will type message Step 4: Tester will send message Step 6: dummy user will log in Step 7: dummy user will select the message option Step 8: dummy user will view the received message **System Response:** Step 5: System will send message to dummy user
Post-condition	Message will be sent and received successfully

Figure 6.8: View Reviews and Ratings

Name	View Reviews and Ratings
ID	08
Priority	Medium
Primary Actor	Client
Description	This test case describes the view Reviews and ratings of Photographer's workflow. Client should be able to view Reviews and Ratings in Photographer's profile.
Basic Flow	Tester will go to the Photographer profile Tester will view reviews and ratings.
Pre-condition	Tester should be logged in Internet should be connected
Steps	**Actor Actions:** Step 1: Tester will go to Photographer profile screen **System Response:** Step 2: Review and ratings will be displayed
Post-condition	Review and Ratings will be displayed successfully

6.4 Result

All test cases were run successfully on the system.

Chapter 7

Conclusion

7.1 Conclusions

The application brings all the vital information together in one platform and focuses on the UI / UX which has resulted in an exceptional satisfaction rate and motivated us to add more to this application in future.

7.2 Recommendations

There are more features which can be added to this application and it can cover not just used in Pakistan but also in other countries in the world.

7.3 Learning Outcomes

From this final year project, the learning outcomes can be listed down as follows:

- Project planning needs to be as detailed and comprehensive as possible

- Deadlines should be set with a realistic approach

- Technology for development should be chosen while keeping cost-effectiveness and project scope in consideration

- Project deadlines should be met at any cost

    The overall learning experience can be used to initiate and complete projects in IT firms.

7.4 Future Work

We will add more functionalities like facial recognition for login, to the application and publish the application to the Google Play Store and App Store. Our work on these applications will continue and we will also add more functionalities to cater the further needs of photographers community and clients so that the application can be used according to user expectations as well.

References

[1] Nilanjan Chatterjee, Souvik Chakraborty, Aakash Decosta, and Asoke Nath. Real-time communication application based on android using google firebase. *Int. J. Adv. Res. Comput. Sci. Manag. Stud*, 6(4), 2018.

[2] NWOKDSP Nanayakkara and DMR Dissanayake. Application of social media for personal branding: A conceptual review. 2020.

[3] Jenni Vilander. Personal branding on social media and social media based entrepreneurship. 2017.